

FPGA alapú rendszerek fejlesztése

Házifeladat

2025.

Tartalom

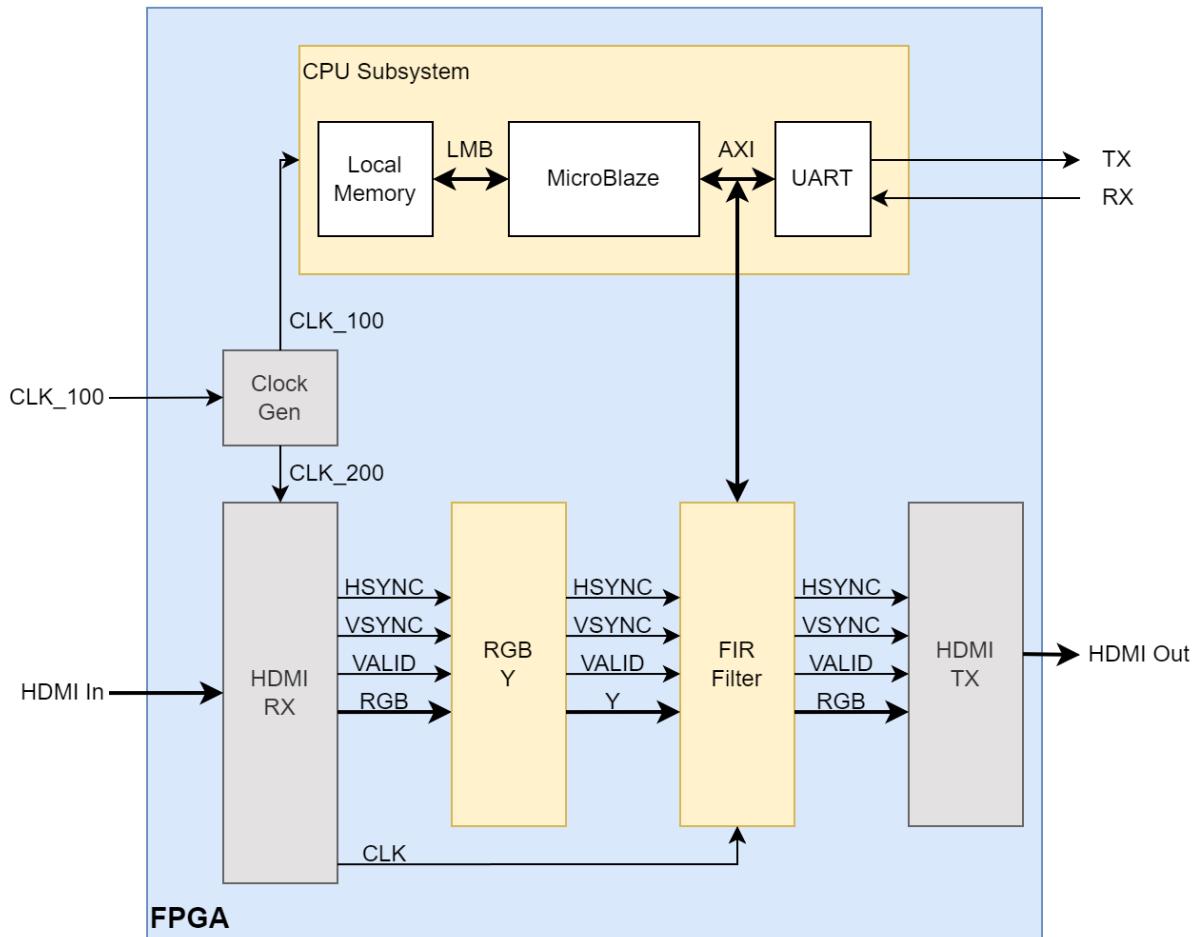
1.	Feladat: 2D FIR szűrő	2
1.1.	Videó formátum	3
2.	Feladatok	4
2.1.	FIR szűrő megvalósítása Verilog nyelven (12 pont).....	4
2.2.	Processzoros alrendszer UART perifériával (3 pont).....	5
2.3.	Regiszter interfész (7 pont)	5
2.4.	Integráció (3 pont).....	5
2.5.	Extra feladat: hisztogram számítás (+10 pont).....	5
3.	Beadandó.....	6
4.	Értékelés	7

1. Feladat: 2D FIR szűrő

A házifeladat egy valós idejű 2D FIR szűrő megvalósítása Logsys Kintex-7 kártyán:

1. A szűrő bemenete az IP blokként kapott HDMI vevő (HDMI RX) kimenete.
2. A szűrő kimenetét az ugyancsak IP blokként kapott HDMI adó (HDMI TX) felé kell továbbítani.
 - 2.1. A szűrőablak mérete 5x5 pixel, az együtthatók formátuma: 16 bites előjeles fix pontos szám 8 bitnyi törtrésszel. Amennyiben előfordulhat, úgy a kimeneti túlcsordulást szaturációval kell kezelní.
3. A szürést szürkeárnyalatos képen kell elvégezni, melyet a 3 színkomponenst tartalmazó bemeneti képből kell előállítani (RGB → Y konverzió, lásd 4. gyakorlat). A kimenetre szürkeárnyalatos képet kell kiadni, azaz minden komponensre az Y kerül.
4. A szűrőnek képesnek kell lennie tetszőleges, legfeljebb 1600x900 pixeles videó folyam feldolgozására. A videó felbontás és frissítési frekvencia működés közben változhat.
5. A szűrő együtthatókészlete működési időben konfigurálható.
 - 5.1. A konfigurációt egy MicroBlaze processzort tartalmazó processzoros alrendszer végzi, amely a felhasználó PC-jével soros porton (UART) keresztül kommunikál.
 - 5.2. Új szűrő konfiguráció alkalmazásához pontosan 25 darab együttható értéket kell az FPGA felé küldeni, amit egy (kocsi vissza + új sor) karakter-pár követ. Amennyiben a fogadott együtthatók száma kevesebb vagy több, úgy a beérkező érték-sorozatot érvénytelennek kell tekinteni. Az együtthatók formátuma:
 - előjel: + vagy – karakter
 - egész rész karakterek
 - tizedespont: .
 - tört rész karakterek

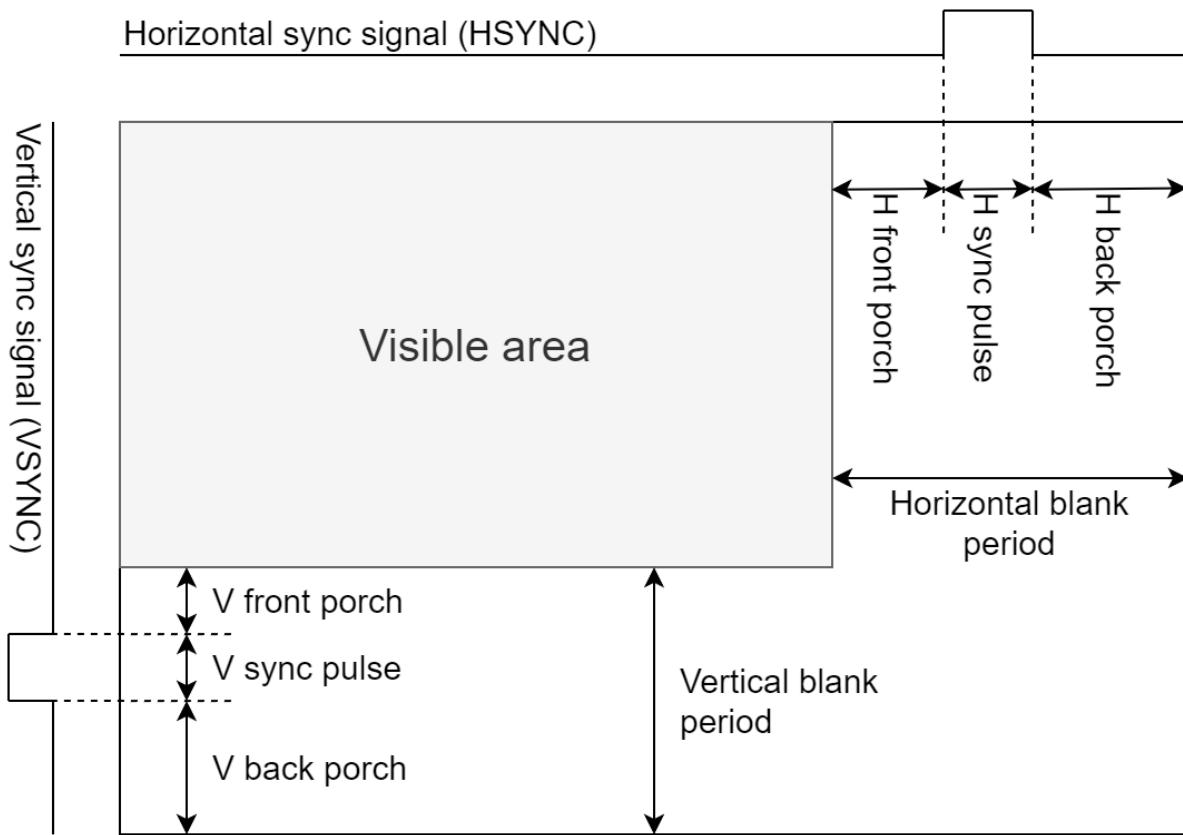
A teljes rendszer blokkvázlatát az alábbi ábra mutatja:



A szürkével jelölt blokkok rendelkezésre állnak a kiadott HF projektben, az RGB→Y átalakító pedig a 4. gyakorlat anyaga.

1.1. Videó formátum

A HDMI vevő minden órajelben egy pixel értékét, valamint a 3 vezérlőjelet szolgáltat, melyek megegyeznek a VGA interfész jeleivel. A teljes továbbított kép mind horizontális, mind pedig vertikális irányban látható tartományból és kioltási (blank) intervallumokból áll. A horizontális kioltási idő alatt (azaz minden egyes sorban) található a horizontális szinkron pulzus (HSYNC), míg a vertikális képkioltási idő alatt (tehát képenként egyszer) a vertikális szinkron pulzus. A pulzusok polaritása felbontástól függően lehet ponált vagy negált, az alábbi ábra ponált esetet mutat.



A HDMI vevő által szolgáltatott jelek:

6. rx_red, rx_green, rx_blue: a 3 színkomponens 8-8 biten
7. rx_hsync: horizontális szinkronjel
8. rx_vsync: vertikális szinkronjel
9. rx_dv: a látható pixelek alatt 1, a blank periódusok alatt 0

A maximális 1600x900 @ 60 Hz felbontás időzítési adatait a mellékelt VESA szabvány tartalmazza. A megvalósított tervnek képesnek kell lennie a HDMI bemeneten érkező bármilyen felbontás kezelésére.

2. Feladatok

2.1. FIR szűrő megvalósítása Verilog nyelven (12 pont)

Valósítsa meg Verilog nyelven a 2D FIR szűrőt. Ennek lépései:

1. A szükséges bemeneti mintákat tároló sorbuffer egység megvalósítása, majd szimulációja.
2. Az együttható számábrázolásának meghatározása. Általános FIR szűrőről lévén szó, a gyakran használt eseteket (pl. átlagolás, GAUSS szűrés, Laplace szűrés) támogatnia kell. A szükséges formátum ezen szűrők paramétereitől alapján határozhatók meg.
3. A tényleges konvolúciót végző MAC egységek minél hatékonyabb megvalósítása, majd szimulációja.
4. A sorbuffer és a MAC egységek integrációja, az integrált rendszer szimulációja.
5. A szűrő tesztelése FPGA kártyán, a processzoros alrendszer nélkül, Verilog-ban megadott együtthatókkal.

2.2. Processzoros alrendszer UART perifériával (3 pont)

Készítse el Vivado Block Design-ban a MicroBlaze alapú processzoros alrendszer AXI UART Lite perifériával. Implementáljon egy teszt szoftvert, amely a PC felől UART-on érkező karaktereket visszaküldi UART-on a PC-nek.

2.3. Regiszter interfész (7 pont)

Egészítse ki a processzoros alrendszer a szűrő együtthatók programozását lehetővé tevő perifériával (25 db írható/olvasható regiszter). A periféria csatlakoztatható közvetlenül az AXI buszra AXI4 Lite interfészen keresztül, de felhasználható a Xilinx AXI to APB Bridge IP is, melynek segítségével elegendő egy egyszerűbb APB slave periféria implementálása.

Módosítsa a 2.2. feladat teszt szoftverét a következőknek megfelelően: a specifikációnak megfelelő formátumban érkező együtthatókat a teljes együtthatókészlet fogadása után írja be a megvalósított regiszterekbe, majd ezen regiszterekből visszaolvasott értékeket küldje el UART-on az alábbinak megfelelő formátumban (EH „együttható sorszám”: „együttható értéke fix pontos decimális formátumban”):

EH 0: +0.0011

EH 1: -0.0012

....

EH 24: +0.0011

2.4. Integráció (3 pont)

Integrálja a processzoros alrendszeret és a FIR szűrőt. Először tesztelje le a rendszert szoftverben meghatározott fix együtthatókkal, majd adja hozzá az UART-on keresztül történő beállítás lehetőségét.

2.5. Extra feladat: hisztogram számítás (+10 pont)

A megvalósítandó egység a MicroBlaze felől érkező engedélyező jel hatására a következő teljes szürkeárnyalatos bemeneti kép alapján hisztogramot számol, amelyet a MicroBlaze processzor soros porton a PC felé továbbít. A hisztogram számítás kérése PC felől: H karakter majd (kocsi vissza + új sor) karakterek elküldésével lehetséges. A PC felé továbbítandó az egyes hisztogram bin-ek értéke vesszővel elválasztva.

3. Beadandó

A házi feladatot a hf.mit.bme.hu címen található MIT HF portálra kell feltölteni, egy ZIP fájl formájában.

A ZIP fájl tartalma:

1. Dokumentáció: **VEZETÉKNÉV.pdf**
 - 1.1. Formátuma a diplomaterv sablon: <https://github.com/FTSRG/thesis-template-word>
2. Egy **VEZETÉKNÉV** könyvtárban a teljes Vivado és Vitis project cleanup után, hogy beleférjen a feltöltési limitbe.

A dokumentáció tartalma:

1. A feladat specifikációja.
2. Hardver leírás
 - 2.1. A teljes terv modul szintű blokkvázlata (saját rajz, pl. Draw.io).
 - 2.2. A HDL-ben megvalósított modulok ismertetése.
 - 2.2.1. A modul leírása.
 - 2.2.2. Részletes, regiszter szintű blokkvázlat.
 - 2.2.3. Kommentezett, színezett HDL kód szövegként beillesztve.
 - 2.2.3.1. Másolható színezett kód online: <https://pinetools.com/syntax-highlighter>
 - 2.2.3.2. A Notepad++ is tud RTF-t másolni plugin segítségével.
 - 2.2.3.3. A Visual Studio Code-ban is van Verilog plugin és tud RTF-t másolni.
 - 2.2.4. Szimulációs eredmény, magyarázattal.
 - 2.3. A processzoros alrendszer ismertetése.
 - 2.3.1. Blokkvázlat.
 - 2.3.2. Címkirosztás.
 - 2.4. Implementációs eredmények: erőforrás felhasználás és időzítés.
 3. Szoftver leírás.
 - 3.1. A MicroBlaze szoftverének ismertetése.
 - 3.1.1. Folyamatábra.
 - 3.1.2. A megvalósított függvények ismertetése.
 - 3.1.3. Kommentezett, színezett C kód.

A házi feladatot élőben is be kell mutatni. A bemutatón belekerdezünk a megvalósítás részleteibe, a kérdésekre gondolkodási idő nélkül kell tudni válaszolni, tehát aki korábban készítette el a HF egyes részeit, az is nézze át, hogy mit-miért-hogyan csinált!

A bemutató pontos időpontját Teams-en hirdetjük meg, értelemszerűen a beadási határidő után lesz, a vizsgaidőszak első hetében.

4. Értékelés

A HF pontszáma hozzáadódik a vizsga pontszámához vagy megfelelő minőség esetén megajánlott jegy kapható.

Megajánlott jegy:

- A megajánlott Jó (4) osztályzat feltétele egy tökéletesen működő FIR szűrő, amelynek együtthatókészletét a processzoros alrendszer szoftvere határozza meg, tehát a szoftver újrafordításával változtatható (nem tartalmazza az UART kommunikációt).
- A megajánlott Jeles (5) osztályzat feltétele az alábbiak közül az egyik:
 - Tökéletesen működő FIR szűrő, amelynek együtthatókészlete a specifikációnak megfelelően UART-on keresztül módosítható.
 - Tökéletesen működő FIR szűrő és hisztogram számító egység. A szűrő együtthatókészletét a processzoros alrendszer szoftvere határozza meg, tehát a szoftver újrafordításával változtatható (UART-on keresztsüli beállítás nincs), a hisztogram értékeit pedig 2.5-nek megfelelően UART-on kell elküldeni a PC-nek.