SNAKE

Feladat felosztás:

    SW: Limbay Bence

    HW: Cseh Péter

Perifériák:

    LCD(27mmx38mm)

    Navigációs gomb(regiszterbe beleírja az utolsó gombnyomás értékét, nem generál interruptot, timer interruptnál kérdezzük le)

    LED: szint kijelzése

    7szeg: pontszám (0-99)

    Gombok: előző szint(BTN2), szint újrakezdése(BTN1), következő szint(BTN0), (regiszterbe beleírja az utolsó gombnyomás értékét, nem generál interruptot, a timer interruptnál kérdezzük le)

    Timer nehézségi szintek beállítása

    UART(debugolásra)

Buszrendszer:

    AXI4-Lite

Szabályok:

Pálya szélének megy a kígyó, akkor meghal.

Ha megeszi a bogyót, akkor 1-el nő.

Ha saját magába belemegy, akkor is meghal

Objektumok:

    Kígyó fej: 3x3 teli

    Kaja: 3x3 -> (0,1; 1,0; 1,2; 2,1) -> lásd ábra a kiadott lapon

    Keret -> úgy vágjuk le, hogy belül 3-al osztható legyen



**Budapesti Műszaki és Gazdaságtudományi Egyetem**

**Villamosmérnöki és Informatikai Kar**

**Programozható logikai eszközök alkalmazástechnikája mellékspecializáció** **(MIT)**

**Mikrorendszerek tervezése**

**Snake játék LOGSYS Spartan 6 FPGA kártáyra**

**Konzulens: Wacha Gábor**

**Készítették: Limbay Bence (E2JT1E)**

**Cseh Péter (DM5HMB)**

**2017.10.06.**

# 1. Specifikáció

## 1.1. Játékleírás, szabályok

A Snake játék célja, hogy egy irányított kígyóval meg kell enni a játéktéren megjelenő ételeket.

A kígyó 4 irányban haladhat a pályán: fölfelé, lefelé, balra és jobbra. Az aktuális haladási irányhoz képest a kígyó egy lépésben nem fordulhat az átellenes irányba, tehát például ha fölfelé megy a kígyó, akkor csak balra vagy jobbra fordulhat, lefelé nem.

A kígyó blokkokból áll. Haladáskor a blokk felveszi az előtte lévő blokk pozícióját, a fej új pozícióba kerül.

A játék a kígyó számára generál ételeket. Ha a kígyó áthalad az ételen (megeszi), akkor a kígyó mérete eggyel növekszik, és a játékos pontja is eggyel nő. Amikor a kígyó megeszi az ételt, a játék új ételt generál.

A játékban 8 szint van, az 1. szint a legkönnyebb, a 8. szint a legnehezebb. A nehézség a kígyó gyorsaságában mutatkozik meg, nehezebb pályákon a kígyó gyorsabban halad.

Ahhoz, hogy a játékos teljesítse a szintet, ahhoz el kell érnie 99 pontot az adott szinten.

Ha a kígyó belemegy a saját farkába, vagy ha neki megy a falnak, akkor meghal, a szintet újra kell kezdeni.

# 2. Hardver

## 2.1. Specifikáció

A Snake játékot a Logsys Spartan-6-os board-ján kell implementálni. A játék a 102x64 pixeles LCD kijelzőn jelenik meg. A játékos pontszámát a hétszegmenses kijelző mutatja, a pályaszintet a LED-ek. A kígyó irányítását a navigációs kapcsolóval lehet megtenni. A három gombbal (BTN0, BTN1, BTN2) lehet pályát váltani, illetve újraindítani a szintet.

## 2.2. Tervezés

### 2.2.1. LCD

SPI buszon keresztül lehetséges

Az LCD-t 33MHz-el lehet legfeljebb frissíteni. FPGA clk freq: 50MHz -> az LCD-t 25MHz-el frissítsük

Regiszterek -> csak írható???. lásd visio

## 2.2.2. Egyszerű soros interfész

#### 2.2.2.1. LED

8 szint kijelzése (one hot kódolás?)

Regiszterek -> lásd visio

#### 2.2.3.2. Hétszegmenses kijelző

Pontok kijelzése: 0-99

Regiszterek -> lásd visio

#### 2.2.4.3. Navigációs gombok

Kígyó irányítása

Regiszterbe beleírja az utolsó irány értékét, nem generál interruptot

A regiszter kiolvasásakor törlődik a regiszterbe írt érték; csak olvasható

### 2.2.3. Nyomógombok

3 gomb: előző pálya betöltése; pálya újrakezdése; következő pálya betöltése

Regiszterbe beleírja az utolsó lenyomott gomb értékét, nem generál interruptot

A regiszter kiolvasásakor törlődik a regiszterbe írt érték; csak olvasható

Szinkronizáció:  
<https://web.stanford.edu/class/ee183/handouts_spr2003/synchronization_pres.pdf>  
<http://www.sunburst-design.com/papers/CummingsSNUG2008Boston_CDC.pdf>

Pergésmentesítés:  
Lassabb frekvenciára mintavételezünk  
Órajel: 50Mhz; Nyomógomb mintavételezése legyen ~ 200Hz;  
50M / 200 = 250000; log2(250k) = 17,93bites számláló és komparáló; ->  
-> 18 bites számláló, nullára komparálunk -> 50M / 2^18 = 190,73Hz-es mintavételi frekvencia

### 2.2.4. Timer

### 2.2.5. Interrupt

### 2.2.6. UartLite

## 2.3. Implementáció

## 2.4. Tesztelés

# 3. Szoftver

## 3.1. Tervezés

## 3.2. Implementáció

## 3.3. Tesztelés

# Irodalomjegyzék