

Név: ~~XXXXXXXXXX~~Neptun: ~~XXXXXXXXXX~~E-mail: ~~XXXXXXXXXX~~

A vizsgadolgozat feladatainak megoldásával szereshető maximális pontszám 75 pont. A házi feladat maximum 25 pontjával az összesen 100 pont alapján a szokásos 40% - 55% - 70% - 85% határokat alkalmazzuk a 2, 3, 4, és 5 jegyek megállapításánál.

1. feladat: teszt kérdések (30 pont)

A kérdéseknél jelölje be az igaz állításokat, a hamisakat pedig ne! Lehetséges olyan kérdés is, ahol mindegyik állítás igaz és olyan is, ahol egyik sem. Állításonként a jó választás +0,5 pontot, a rossz választás -0,5 pontot, csak a „nem tudom” válasz bejelölése pedig 0 pontot ér. A feladatra kapható teljes pontszám minimum 0 pont.

1) Melyik állítás igaz a processzor adatstruktúrákra?

- A MicroBlaze CPU két regisztercímes adatstruktúrával rendelkezik.
- A RISC CPU-k akkumulátor alapú adatstruktúrával rendelkeznek.
- A verem alapú adatstruktúra esetén az eredmény mindig a verem tetejére kerül.
- A két regisztercímes architektúra jó kompromisszum a programban szükséges utasítások száma és az utasításszavak mérete között.
- Nem tudom.

3) A Xilinx Vivado környezetben...

- ...az IP alapú fejlesztést az IP Integrator biztosítja a Block Design-ok által.
- ...a megkötések az XDC fájlokban adhatók meg, amelyek TCL parancsokat tartalmaznak.
- ...a logikai analízátor felhasználói felülete nem a Hardware Manager-ből érhető el.
- ...az IP-k IP-XACT leírása az IP Integrator segítségével módosítható.
- Nem tudom.

5) Melyik állítás igaz az AXI4 buszra?

- Az írási művelet során a cím és az adat átvitele egymástól független.
- A VALID jel magas szintje jelzi, hogy a cél tud adatot fogadni.
- A master egység nem adhatja ki a BREADY jelzést, mielőtt a slave egység beolvassa az adatot.
- Az írási és az olvasási adatátvitel egymástól függetlenül végrehajtható.
- Nem tudom.

7) Melyik állítás igaz a megszakításra (interrupt)?

- Kiszolgálását a MicroBlaze státusz regiszterének IE bitje engedélyezi.
- A MicroBlaze processzor esetén a megszakítás a kivételnél nagyobb prioritású esemény.
- A megszakítás egy jelzés a hardver egységtől a processzor felé kiszolgálási igényre.
- A megszakítás használatához mindenképpen szükséges megszakítás vezérlő periféria.
- Nem tudom.

9) Melyik állítás igaz a DDR3 SDRAM-ra?

- A késleltetése miatt csak burst-ös és átlapolat adatátvitel esetén biztosít megfelelő hatékonyságot.
- Nem kell inicializálni, a bekapcsolás után egyből használható.
- Szóhossz növelése esetén az összes memória IC adatvonalai és DQS vonalai azonos jelterjedési idővel kell, hogy rendelkezzenek.
- Írási vagy olvasási hozzáférés előtt az adott bankban aktiválni kell a megfelelő sort az ACT parancs kiadásával.
- Nem tudom.

11) Melyik állítás igaz a DMA adatátvitelre?

- Az AXI DMA periféria memóriába leképzett perifériák között valósít meg adatátvitelt.
- Használata esetén az adatátvitelt hatékonyan, hardveresen valósítja meg a DMA vezérlő vagy a bus-master képességű I/O periféria.
- Alatta a processzor egyéb hasznos tevékenységet is végezhet.
- Virtuális memóriakezelés esetén csak a scatter-gather DMA mód használható, mert az adatpuffer nem biztos, hogy fizikailag folytonos.
- Nem tudom.

2) Melyik állítás igaz a beágyazott rendszerekre?

- Hardver és szoftver együttese.
- Figyelik a környezetüket és reagálnak a bekövetkező változásokra.
- Ezeket tudományos számítások támogatására tervezik.
- Az FPGA eszközzel megvalósított beágyazott rendszerek biztosítják a hardver módosíthatóságát is, ami hasznos a hibák javítása vagy új funkciók hozzáadása szempontjából.
- Nem tudom.

4) Melyik állítás igaz a Xilinx MicroBlaze processzorra?

- Egyetlen utasítással tud memóriából memóriába adatot mozgatni.
- A pipeline felépítés miatt a végrehajtott ugrás esetén bekövetkező veszteségek mérsékléséhez támogatott a delay slot és az ugrásbecslés.
- A periféria regiszterek eléréséhez AXI4-Lite adat interfészt használ.
- A memória elérést biztosító utasításoknál csak abszolút és regiszter indirekt címzési módok állnak rendelkezésre.
- Nem tudom.

6) Az AXI4-Stream interfész esetén...

- ...a TKEEP=1 és TSTRB=0 kombináció a pozíció bájtot jelzi, amely nem távolítható el a folyamból, de az értéke megváltoztatható.
- ...nem biztosítható az eltérő órajel tartományban lévő egységek közötti adatátvitel.
- ...a TREADY jel jelzi az adatcsomag utolsó szavát.
- ...a slave egység is tud adatot küldeni a master egység felé.
- Nem tudom.

8) Az AXI Interrupt Controller periféria...

- ...adja a megszakításkezelő rutin címét a processzornak, ha a gyors megszakítás mód használata engedélyezve van.
- ...csak érzékeny megszakítás bemenetekkel rendelkezik.
- ...Interrupt Status Register-ében (ISR) a megszakítás bemenetekhez nem tartozó bitek írhatók, így szoftverrel is kiváltható megszakítás.
- ...használata esetén 32-nél több megszakítás forrás is kezelhető, mert lehetőség van a megszakítás vezérlők kaszkádosítására.
- Nem tudom.

10) Melyik állítás igaz a BSP alacsony szintű eszközmeghajtóira?

- Több eszközpéldány támogatását a báziscím megadásával biztosítják.
- Az adott hardver eszköz minden funkcióját elérhetővé teszik az alkalmazás számára.
- Kisebb memóriaigénnyel rendelkeznek a magas szintű eszközmeghajtókhoz képest.
- Az itt lévő adatátvitellel kapcsolatos függvények nem blokkoló hívásokat valósítanak meg.
- Nem tudom.

12) Melyik funkcionális analízisre vonatkozó állítás igaz?

- SW→HW irányú cross-trigger esetén a trigger feltétel helyét az ablak legelejére kell megválasztani a logikai analízátorban.
- A Xilinx SDK-ban assembly utasítások szintjén elhelyezhető töréspont.
- A Vivado logikai analízátor trigger komparátora nem képes felfutó, illetve lefutó élt detektálni a vizsgált jelen.
- A szoftver debuggolást a MicroBlaze CPU esetén az MDM biztosítja.
- Nem tudom.

13) Melyik állítás igaz a Linux eszközfára (device tree)?

- A Xilinx SDK Windows operációs rendszer alatt is képes ezt generálni a Device Tree Generator kiegészítő segítségével.
- A nem felderíthető buszra kapcsolódó perifériák adatait tárolja hierarchikus formában.
- Nem tartalmazza a perifériára vonatkozó cím információit.
- Az itt lévő **compatible** azonosítók alapján keresi meg az operációs rendszer a perifériákhoz az eszközmeghajtót.
- Nem tudom.

15) Melyik állítás igaz a Linux eszközmeghajtókra?

- Itt nem fontos a hibakezelés, mert a kernel ezt elvégzi.
- Nem futnak folyamatosan, hanem „eseménykezelőket” tartalmaznak, amelyeket a kernel hív meg.
- A **platform device**-okhoz tartozó meghajtók az eszközfából (device tree) kérdezik le a kezelt eszköz paramétereit.
- A kezelt periféria regisztereinek eléréséhez először a periféria fizikai címét le kell képezniük a virtuális címtartományba.
- Nem tudom.

14) Melyik állítás igaz a Linuxot futtatni képes rendszerekre?

- A processzornak kell tartalmaznia gyorsítótárat.
- A rendszerben mindenképpen kell, hogy legyen DMA vezérlő periféria a hatékony adatátvitel megvalósítása érdekében.
- A rendszerben kell, hogy legyen időzítő periféria és nagyobb kapacitású memória is.
- A rendszernek nem szükséges tartalmaznia megszakítás vezérlőt, mert a Linux kernel lekérdezősen is kezeli a perifériákat.
- Nem tudom.

2. feladat: adatátvitel AXI4 buszon (10 pont)

Adott egy processzoros rendszer, amelyben 32 bites AXI4 buszon keresztül kapcsolódik egymáshoz a külső memória vezérlő, a Xilinx AXI Central DMA periféria és egy memóriába leképzett, burst-ös adatátvitelt is támogató 32 bites FIFO (mélysége 512 szó). A FIFO adatregiszter címe 0x40000000. A külső memória báziscíme 0xC0000000, mérete 1 GB. A DMA átvitel indításánál a DMA vezérlő forráscím regiszterébe (SA) 0x40000000, a célcím regiszterébe (DA) 0xC0000010 és a bájtyszám regiszterébe (BTT) pedig 0x00000020 érték kerül beírásra. A DMA vezérlő úgy van hardveresen konfigurálva, hogy egy burst legfeljebb 16 ütemből állhat. Válaszoljon az alábbi kérdésekre!

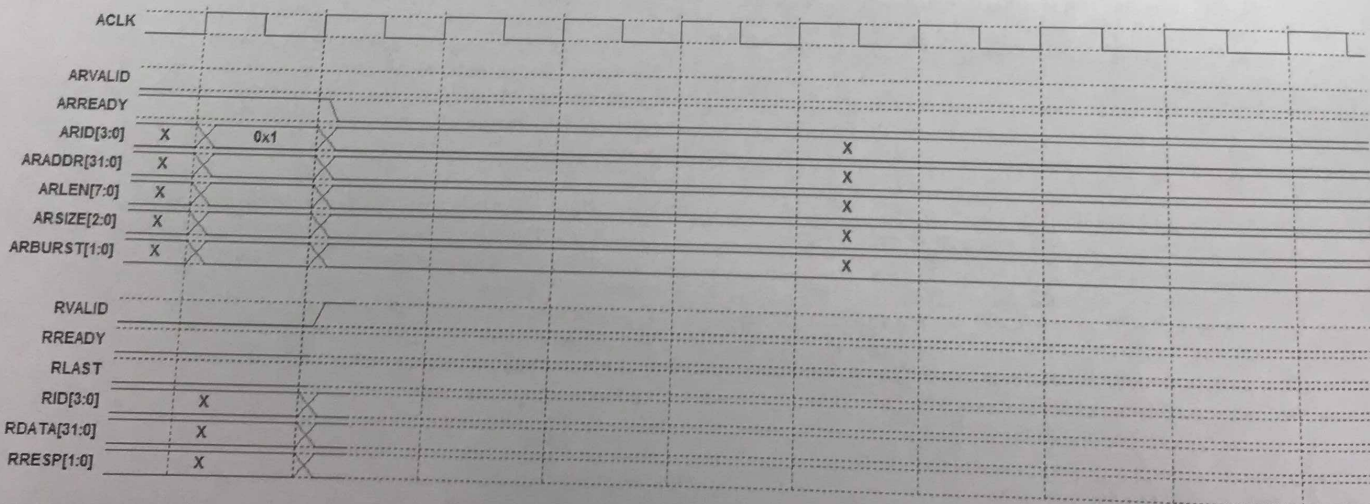
a) Hány bájt kerül átvitelre egy ütemben és ezt melyik vonal jelzi az AXI4 buszon? (Segítség: a bájtok számát a megfelelő jel értékének kettő hatványa adja.)

b) Hány ütemből áll az adatátvitel és ezt melyik vonal jelzi az AXI4 buszon? (Segítség: az ütemszámot a megfelelő jel eggyel megnövelt értéke adja.)

c) A FIFO-hoz tartozó adatátvitel esetén milyen típusú burst-re van szükség? Húzza alá a jó választ és röviden írja le ennek jellemzőit!
INCR (01) WRAP (10) FIXED (00)

d) Milyen címre kerül kiírásra a burst utolsó szava?

e) Rajzolja be, illetve töltsze ki a DMA olvasási adatátvitelhez tartozó idődiagram hiányzó részeit és a bitvektorok értékeit! Tegyük fel, hogy minden adatátviteli fázis a lehető leghamarabb végrehajtásra kerül. A nyugtázás minden esetben OKAY (00). **A teljes DMA átvitel után a külső memória tartalma az idődiagram alatti táblázatban látható.**



A memória tartalma	0x00	0x04	0x08	0x0C
0xC0000000	0x00000011	0x00000022	0x00000033	0x00000044
0xC0000010	0x00005500	0x00006600	0x00007700	0x00008800
0xC0000020	0x00990000	0x00AA0000	0x00BB0000	0x00CC0000
0xC0000030	0xDD000000	0xEE000000	0xFF000000	0x00000000

3. feladat: MicroBlaze assembly program vizsgálata (10 pont)

Adott egy assembly nyelvű program, amely AXI4 interfésszel rendelkező MicroBlaze processzorra készült. Elemezze a program működését és válaszoljon a feladatban szereplő kérdésekre! Segítségképpen megadjuk az egyes utasítások értelmezését és a memória műveletekhez az adatmemória tartalmát.

- Mi történik, ha írjuk a MicroBlaze processzor r0 regiszterét? Milyen érték olvasható ki ebből a regiszterből?
- A MicroBlaze processzor utasításkészletében nincs olyan assembly mnemonik (pl. `mov rD, rA`), amely közvetlenül regiszterből regiszterbe mozgat adatot. Milyen assembly utasítással lehetne ezt a műveletet mégis megvalósítani (segítségként lásd az utasítások értelmezését tartalmazó táblázatot)?
- Mit jelent a big-endian bájtsorrend?
- A feladatban szereplő MicroBlaze processzor esetén mi a natív bájtsorrend? Húzza alá a jó választ! **little-endian** **big-endian**
- Töltse ki a programot tartalmazó táblázatban az utasítások végrehajtása után a cél regiszter új értékét, ahol ez értelmezett, illetve adatmemória írás esetén az új tartalmat írja be bájtonként a megfelelő címekre!

A program utasításai		A cél regiszter új értéke (hexa)	Az adatmemória tartalma				
			0x00	0x01	0x02	0x03	
<code>imm</code>	<code>0xA000</code>		<code>0xA0000000</code>	<code>0x00</code>	<code>0xFF</code>	<code>0x78</code>	<code>0x09</code>
<code>addi</code>	<code>r21, r0, 0x0005</code>		<code>0xA0000004</code>	<code>0xAA</code>	<code>0x82</code>	<code>0xAA</code>	<code>0x80</code>
<code>addi</code>	<code>r22, r21, 0x0007</code>		<code>0xA0000008</code>	<code>0xDE</code>	<code>0xAD</code>	<code>0x37</code>	<code>0x62</code>
<code>addi</code>	<code>r23, r22, 0x0008</code>		<code>0xA000000C</code>	<code>0x11</code>	<code>0x00</code>	<code>0x01</code>	<code>0x00</code>
<code>lbu</code>	<code>r24, r21, r0</code>		<code>0xA0000010</code>	<code>0x39</code>	<code>0x40</code>	<code>0xAC</code>	<code>0xFF</code>
<code>sext8</code>	<code>r24, r24</code>		<code>0xA0000014</code>	<code>0x11</code>	<code>0x22</code>	<code>0x33</code>	<code>0x44</code>
<code>lw</code>	<code>r25, r22, r0</code>		<code>0xA0000018</code>	<code>0x99</code>	<code>0x66</code>	<code>0x17</code>	<code>0x32</code>
<code>add</code>	<code>r25, r24, r25</code>		<code>0xA000001C</code>	<code>0x84</code>	<code>0x42</code>	<code>0x99</code>	<code>0x23</code>
<code>swr</code>	<code>r25, r0, r23</code>		<code>0xA0000020</code>	<code>0x03</code>	<code>0x55</code>	<code>0x01</code>	<code>0x00</code>
			<code>0xA0000024</code>	<code>0x03</code>	<code>0x05</code>	<code>0xF0</code>	<code>0xBA</code>

A MicroBlaze assembly utasítások értelmezése		
<code>imm</code>	<code>IMM</code>	konstans felső 16 bitjének betöltése, az alsó 16 bitet a rákövetkező utasítás tartalmazza
<code>addi</code>	<code>rD, rA, IMM</code>	regiszter összeadása konstanssal: $rD = rA + IMM$
<code>add</code>	<code>rD, rA, rB</code>	regiszter összeadása regiszterrel: $rD = rA + rB$
<code>lbu</code>	<code>rD, rA, rB</code>	8 bites előjel nélküli memória olvasás: $rD[0:23] = 0x000000$ és $rD[24:31] = DMEM8[rA + rB]$
<code>lw</code>	<code>rD, rA, rB</code>	32 bites memória olvasás natív bájtsorrenddel : $rD = DMEM32[rA + rB]$
<code>swr</code>	<code>rD, rA, rB</code>	32 bites memória írás fordított bájtsorrenddel : $DMEM32[rA + rB] = rD$
<code>sext8</code>	<code>rD, rA</code>	8 bites érték előjel kiterjesztése: $rD[0:23] = \{24[rA[24]]\}$ és $rD[24:31] = rA[24:31]$

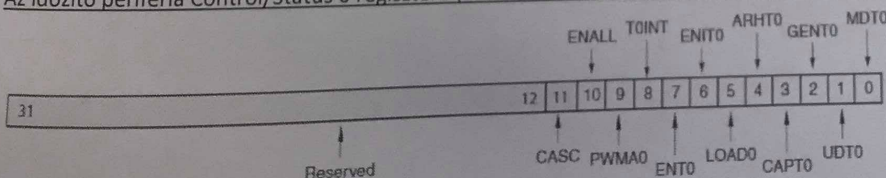
4. feladat: C program írása (5 pont)

Külön lapon készítsen MicroBlaze processzorra egy egyszerű C nyelvű programrészletet, amely az 5. feladatban szereplő periféria és egy AXI Timer periféria használatával a `btn0_pressed` változó 1-be állításával jelzi, ha a BTNO nyomógombot éppen megnyomták. A perifériák báziscímét az `xparameters.h` fájlban definiált `XPAR_SW_BTN_BASEADDR` és `XPAR_AXI_TIMER_BASEADDR` konstansok adják meg. A nyomógomb állapotváltozásának detektálásához az időzítő periféria másodpercenként 100 megszakítást generál, a hozzá tartozó megszakításkezelő rutint a `timer_int_handler` függvény valósítja meg. A megszakítás jelzés az időzítő Control/Status 0 regiszterében a 8. bit (TOINT), amely 1 beírásával törölhető. A perifériák inicializálásával nem kell foglalkozni.

```
//A BTNO nyomógomb eseményt jelző változó.
volatile unsigned char btn0_pressed;

//Az időzítő periféria megszakításkezelő rutinja (ezt kell elkészíteni).
void timer_int_handler(void *instancePtr) {
    .....
}
```

Az időzítő periféria Control/Status 0 regisztere (BASEADDR + 0x00, 32 bites, írható és olvasható):



5. feladat: perifériaillesztés AXI4-Lite buszra (20 pont)

Illeszen egyszerű regiszter típusú slave perifériát AXI4-Lite buszra, a leírást Verilog nyelven adja meg külön lapon! A perifériának a LOGSYS Kintex-7 FPGA kártyán lévő 8 darab kapcsoló és 4 darab nyomógomb állapotának lekérdezését kell biztosítania két 32 bites, csak olvasható regiszter segítségével:

- BÁZISCÍM+0x00: kapcsoló adatregiszter (a felső 24 bit mindig 0)
- BÁZISCÍM+0x04: nyomógomb adatregiszter (a felső 28 bit mindig 0)

Minden, a „külvilágból” bejövő adatbitet szinkronizáljon az AXI órajelhez két flip-flop felhasználásával! Az AXI írási műveletek kezelését nem kell megvalósítani. Az AXI busz olvasási csatornái közötti függőségeket figyelembe véve törekedjen a lehető legegyszerűbb busz interfész egység elkészítésére. **Minden jó megoldást elfogadunk, de a legegyszerűbb éri a legtöbb pontot!**

Töltse ki a periféria Verilog moduljának fejlécében a hiányzó részeket, a cím bitszáma a lehető legkisebb legyen! A továbbiakban az itt szereplő jelneveket, a `clk` órajelet és az aktív magas `rst` reset jelet használja!

```
module sw_btn(  
    //AXI órajel és reset jel.  
    ..... wire          s_axi_aclk,  
    ..... wire          s_axi_aresetn,  
  
    //AXI4-Lite olvasási cím csatorna.  
    ..... wire [.....:0] s_axi_araddr,  
    ..... wire          s_axi_arvalid,  
    ..... wire          s_axi_arready,  
  
    //AXI4-Lite olvasási adat csatorna.  
    ..... wire [31:0] s_axi_rdata,  
    ..... wire [1:0] s_axi_rresp,  
    ..... wire          s_axi_rvalid,  
    ..... wire          s_axi_rready,  
  
    //Felhasználói jelek.  
    ..... wire [7:0] sw,  
    ..... wire [3:0] bt,  
);  
  
//Belső órajel és reset jel.  
wire clk = s_axi_aclk;  
wire rst = ~s_axi_aresetn;
```