

Processzortechnika (GEVAU508B) c. tantárgy
előadásának ütemterve
Villamosmérnöki (BSc) Alapszak
G3BV1, G3BV2 tanulókörök számára

Naptári hét	Előadás
7.	CPU modell, aritmetika
8.	Aritmetika, vezérlésátadás
9.	ARM utasítások, ASM, C eszközök
10.	C projekt felépítése, CMSIS
11.	Startup modul, linker szkript
12.	IO függvények
13.	C-könyvtár, Filerendszer, ADC-DAC
14.	IO: GPIO, SPI, UART, Timer
15.	Cache kezelés I
16.	Cache kezelés II
17.	Pipeline: hazardok, out-of-order végrehajtás
18.	Pipeline: ugrások, előrejelzés.
19.	Ismétlés
20.	Ismétlés

Miskolc, 2018. február 9.

Dr. Trohák Attila
intézet igazgató, egyetemi docens

Drótos Dániel
tanszéki mérnök
előadó

Processzortechnika (GEVAU508B) c. tantárgy
gyakorlatának ütemterve
Villamosmérnöki (BSc) Alapszak
G3BV1, G3BV2 tanulókörök számára

Naptári hét	Gyakorlat
7.	ARM fejlesztőrendszer bemutatása
8.	ARM programok felépítése
9.	ARM programozás gyakorlása
10.	Egyéni feladatok kiadása
11.	Egyéni feladat megoldása
12.	Egyéni feladat megoldása
13.	Egyéni feladat megoldása
14.	Egyéni feladat megoldása
15.	Egyéni feladat megoldása
16.	Egyéni feladat megoldása
17.	Egyéni feladat megoldása
18.	Egyéni feladat megoldása
19.	Pótlás
20.	Pótlás

Miskolc, 2018. február 9.

Dr. Trohák Attila
intézet igazgató, egyetemi docens

Drótos Dániel
tanszéki mérnök
gyakorlatvezető

Processzortechnika (GEVAU508B) c. tantárgy
követelményrendszere
Villamosmérnöki (BSc) Alapszak
G3BV1, G3BV2 tanulókörök számára

Aláírás feltételei:

- Gyakorlaton a féléves feladat legalább elégséges szinten való megoldása (a félév során, illetve az aláírás pótlási időszakban pótolható).

Gyakorlat: féléves egyéni feladat, amely 1-5 osztályzattal értékelt.

Félév értékelése: Aláírás megszerzése után a kiadott kérdéssor alapján írásbeli vizsgát kell tenni. A félév teljesítéséhez legalább elégségest kell elérni a vizsgán. A féléves jegy a vizsga jegye.

Miskolc, 2018. február 9.

Dr. Trohák Attila
intézet igazgató, egyetemi docens

Drótos Dániel
tanszéki mérnök
gyakorlatvezető

ZH nincs a tárgyból.

GEVAU508B{L} Processzortechnika vizsga

Név: _____

Neptun kód: _____

1. Az ARM processzor regiszterei, az egyes regiszterek speciális funkciói. _____
2. Sorolja fel, milyen utasítások megvalósítására kell áramköröket tartalmaznia az ALU-nak! Tegyen javaslatot arra, hogy melyik utasítás melyik flag-eket módosítsa! _____
3. Hogyan kell beépíteni C nyelvű ARM programba a megszakítás kiszolgáló szubrutinokat (ISR)? _____
4. Regiszter átnevezés nélküli, pipeline processzoron futtatjuk a következő kódrészletet:
 1. MOV R1=1234
 2. MOV R2=5678
 3. SUB R3=R1-R2
 4. DEC R4
 5. INC R3Mely utasítások között kell „buborékok” alkalmazásával kiüríteni a pipeline-t és miért? _____
5. Pipeline felépítésű processzorokban milyen módszereket alkalmaznak annak előre jelzésére, hogy a feltételes vezérlésátadás be fog-e következni? _____

Értékelés: 0-2p: 1; 2.1-2.7p: 2; 2.8-3.5p: 3; 3.6-4.2p: 4 4.3-5.0p: 5

Eredmény: _____

Vizsga megoldási minta

1. Az ARM processzor regiszterei, az egyes regiszterek speciális funkciói.

R0..R15 általános célú 32 bites regiszterek. Ezek közül speciális szerepűek:

R13 stack pointerként használatos

R14 Link regiszter, szubrutinhíváskor tárolja a visszatérési címet

R15 PC, programszámláló

SR Státusz és vezérlő regiszter, a Flag biteket, az üzemmód beállító biteket, a megszakítás engedélyező biteket tárolja

2. Sorolja fel, milyen utasítások megvalósítására kell áramköröket tartalmaznia az ALU-nak! Tegyen javaslatot arra, hogy melyik utasítás melyik flag-eket módosítsa!

Aritmetikai utasítások:

- összeadás (minden flag)
- kivonás (minden flag)

Logikai utasítások:

- logikai műveletek: és, vagy, kizáró vagy, negálás (Z flag)
- eltoló, forgató műveletek (Z, C flag-ek)

3. Hogyan kell beépíteni C nyelvű ARM programba a megszakítás kiszolgáló szubrutinokat (ISR)?

- A startup modulban létrehozunk függvényneveket az ISR-ek számára, ezeket weak tulajdonsággal látjuk el.
- A weak nevekkel feltöltjük a megszakítás táblázatot.
- A weak neveket alapértelmezésben hozzárendeljük egy „default” függvény névhez.
- Az alap függvény általában egy végtelen ciklusból áll.
- A programban a weak függvénnyel azonos nevű függvényt hozunk létre ISR gyanánt.
- Az ISR kódját az általános szabályok betartásával írjuk (rövid kód, függvényhívások kerülése)

4. Regiszter átnevezés nélküli, pipeline processzoron futtatjuk a következő kódrészletet:

1. MOV R1=1234

2. MOV R2=5678

3. SUB R3=R1-R2

4. DEC R4

5. INC R3

Mely utasítások között kell „buborékok” alkalmazásával kiüríteni a pipeline-t és miért?

A 2-3 között, mert a 3-ban az R2 bemenet a 2-ben kimenetként (eredmény) szerepel.

5. Pipeline felépítésű processzorokban milyen módszereket alkalmaznak annak előre jelzésére, hogy a feltételes vezérlésátadás be fog-e következni?

Statikus előrejelzés: nem használja fel a történetet. Pl:

- Az előrejelzés mindig: igen (ugrás lesz)
- Az előrejelzés az ugrásiránytól függ: hátraugrásnál: igen, előre ugrásnál: nem
- Véletlenszerű: random bittől függő eredményt ad.

Dinamikus előrejelzés, figyelembe veszi a múltbéli ugrásokat

- Egyszintű: 4 állapot, amelyet egy 2 bites számláló reprezentál. Ugrásnál növekszik, nem ugrásnál csökken. A tipp igen, ha a számláló 2 vagy 3 értékű, 0 vagy 1 esetén a tipp nem.
- Kétszintű: az előző n ugrás eredmény egymás mellé illesztve egy n bites számot ad. Ez indexként szolgál egy 2^n elemű táblázathoz, amelyben az egyszintű módszernél ismertetett 2 bites számlálók vannak. Az előrejelzéshez felhasznált számláló az előző n ugrás eredményéből adódó mintázattól függ.