

## Ellenőrző kérdések a „Nagyteljesítményű mikrovezérlők” tárgyhoz (VIMIM342)

1. Mutassa be, a mikrovezérlő piac alakulását az elmúlt 20 évben! Mutassa be az ARM cég filozófiáját és a hagyományos ARM magok: ARM7, ARM9, ARM11 jellegzetességeit. Mutassa be az ARM Cortex mag sorozatának (M, R, A) jellegzetességeit! Melyik Cortex-es sorozat melyik hagyományos mag sorozat kiváltására készült!
2. Hasonlítsa össze a 8 bites és 32 bites mikrovezérlők lábszámának és Flash memóriájának, változását az elmúlt 10 évben, ábrázolja ezt a lábszám - flash memória koordinátarendszerben!
3. Hasonlítsa össze a 8 bites és 32 bites mikrovezérlők sleep fogyasztásának és működési feszültség tartományának, árának, valamint perifériakészletének alakulását az elmúlt 10 évben! Ismertesse a 8 bites és 32 bites mikrovezérlők választása mellett és ellen szóló legfőbb érveket!
4. Mutassa be az ARM Cortex M sorozat tagjainak (M0, M3, M4, M7) képességeit és hasonlítsa őket össze. Röviden adja meg képességeiket, valamint a mikrovezérlő piacon megélt szerepüket!
5. Ismertesse az ARM7 magok jellemzőit: utasításkészlet, architektúra működési módok, interrupt kezelés! Mutassa be a Cortex M3 processzor mag fő jellemzőit: architektúra, főbb blokkok, újdonságok az ARM7-hez képest!
6. Mutassa be röviden a Cortex M3 alapú vezérlők memória szervezésének jellemzőit. Milyen főbb cím tartományok vannak, mi az a bit banding, mi a non-aligned memória hozzáférés. Mik a főbb különbségek az ARM7 alapú vezérlőkkel szemben!
7. Mutassa be a Cortex M3 utasítás végrehajtását (pipe-line), programozói modelljét (load and store), regiszter készletét! Mutassa be a Cortex M3 működési módjait és hasonlítsa ezeket össze az ARM7 működési módjaival!
8. Mutassa be a Cortex M3 standard perifériáit (System timer, NVIC, debugg blokkok) és az ezek által nyújtott előnyöket! Mutassa be a piacon kapható legelterjedtebb Cortex M3 alapú mikrovezérlő sorozatokat!
9. Mutassa be a legelterjedtebb ARM7 magú és Cortex M3 magú mikrovezérlő sorozatok belső felépítésének fejlődését az elmúlt 10 évben! Milyen változásokat mi indokolt?
10. Mutassa be röviden a Cortex M0 mag jellegzetességeit, miben különbözik a Cortex M3 magtól és az ARM7 magtól! Mi az a WIC (Wake-up Interrupt Controller) és miért fontos az energiatakarékosság szempontjából?
11. Mutassa be a Cortex M4 mag jellegzetességeit! Mik azok a SIMD és MAC utasítások és miért lehetnek ezek hatékonyak jelfeldolgozásra? Milyen további fejlesztéseket hoz az M7 mag az M4-hez képest?
12. Mutassa be, hogy az ARM Cortex M0 és M4 magú vezérlők tipikus belső felépítését. Mik a hasonlóságok és különbségek? Mutassa be az NXP LPC4300-as sorozatának felépítését! Milyen szerepet szánunk az M0-ás és az M4-es magoknak, mutasson erre példát!
13. Mutassa be a Reset és elindulás folyamatát egy tipikus ARM Cortex magú vezérlőn! Milyen műveletekre és miért van szükség ahhoz, hogy a C program main függvénye elindulhasson?

14. Mi az a Flash gyorsító modul, miért szükséges. Mutassa be röviden a hasznát és a működését! Mutasson rá példákat a különböző ARM magú mikrovezérlő generációkból!
15. Mutassa be egy tipikus ARM Cortex M magú vezérlő órajel hálózatát. Magyarázza meg az egyes órajel osztások értelmét és szükségességét. Milyen előnyökkel és hátrányokkal járnak ezek a beállítási lehetőségek.
16. Mutassa be röviden a CMSIS (Cortex Microcontroller Software Interface Standard) bevezetésének szükségességét. Ismertesse a CMSIS core szerepét és az általa nyújtott szolgáltatásokat.
17. Mutassa be röviden a CMSIS v 4 (Cortex Microcontroller Software Interface Standard) szerkezetét és újdonságait. Mutassa be a piacon kapható fejlesztést segítő szoftver tool-okat, és hogy a CMSIS milyen hatással volt ezekre.
18. Mutassa be egy általános 32 bites mikrovezérlő GPIO lábainak kezelését. Hogyan támogatja ezt a CMSIS, majd a Firmware library-k? Milyen a 8 bites vezérlőkre nem jellemző problémák léphetnek fel a GPIO lábak kezelésénél?
19. Hasonlítsa össze a Cortex M3 NVIC-ét az ARM7 megszakításkezelési lehetőségeivel. Mutassa be az NVIC interrupt vektor tábla szervezésének főbb jellegzetességeit (nem kell tudni fejből az IT tábla felépítését)! Hány periféria megszakítást támogat az NVIC, mindegyiket ki szokták ezek közül használni?
20. Mutassa be az Cortex M3 NVIC prioritás kezelésének alapjait. Mire jók a megszakításmaszk regiszterek, és mi az értelme a Vector Table offset regiszternek? Mutassa be a Cortex M3 NVIC-jének megszakítás végrehajtási folyamatát! Mi az a tail-chaining, mi történik ilyenkor?
21. Mutasson példát a DMA kezelés használatára! Milyen átviteli és működési lehetőségeket kínál egy általános DMA blokk. Milyen paramétereket kell általában beállítani egy DMA átvitelhez?
22. Mutassa be röviden a mikrovezérlőkben alkalmazott DMA-k tulajdonságait: milyen előnyökkel ad a DMA a hagyományos elrendezéshez képest? Milyen beállítási lehetőségei vannak egy tipikus mikrovezérlős DMA-nak? Mi az a Scatter and gather DMA? Hogyan próbálják a DMA hatékonyságát növelni az új sorozatú vezérlőknél (memória elrendezés)? Hogyan függ a DMA sebessége attól, hogy memória – memória, vagy periféria memória átvitelre van szükség?
23. Mutassa be röviden a beágyazott rendszereknél tipikusan alkalmazott szoftver architektúrákat!
24. Röviden mutassa be a beágyazott operációs rendszerek és az asztali operációs rendszerek közötti különbségeket! Tipikusan hogyan szerveződnek az operációs rendszerek forrás file-jai, mik az alapszolgáltatások, amiket mindegyik tud, és hogyan konfigurálhatóak?
25. Röviden mutassa be a FreeRTOS felépítését és jellegzetességeit! Ismertesse a FreeRTOS könyvtárszerkezetét és a portolásánál végrehajtandó lépéseket.
26. Tipikusan milyen felhasználási módjai vannak a mutex, semafor, queue objektumoknak? Mindegyikre adjon példát. Kell e különbséget tenni a kezelésben, ha interruptban használjuk ezeket az objektumokat?
27. A FreeRTOS milyen memóriakezelési lehetőséget nyújt, ezek milyen viszonyban vannak a normál *malloc* C utasítással és a heap-el? Milyen lehetőségeink vannak a

- FreeRTOS-ben a stack méretének monitorozására? Hogyan tudjuk a FreeRTOS esetében követni az ütemező működését?
28. Mutassa be, hogy milyen lehetőségeink vannak a mikrovezérlők **aktív** fogyasztásának befolyásolására 8 bites és 32 bites mikrovezérlők esetében. Hasonlítsa össze nagyvonalakban a 8 bites és 32 bites mikrovezérlők aktív fogyasztásának alakulását az elmúlt pár évben.
  29. Mutassa be, hogy egy általános 8 bites és 32 bites mikrovezérlőnek milyen energiatakarékos módjai vannak (általánosan jellemző módok kellene, nem kell tudni, hogy melyik vezérlőnél hogy hívják ezeket). Milyen újdonságokat tartalmaz a Silabs EFM32 sorozata az energia takarékos módoknál?
  30. Tipikusan miért vannak hátrányban a 32 bites vezérlők a 8 bites társaikhoz képest az energiatakarékos fogyasztás (passzív állapotok) tekintetében. Milyen technikákat, trükköket vetnek be ezek kompenzálására? Milyen előnyei lehetnek egy 32 bites vezérlőnek egy 8 bites vezérlővel szemben energiatakarékos alkalmazásoknál (miért lehet fontos a hatékony utasítás végrehajtás)?
  31. Mi az a ROM monitor (más néven GDB stub), tradicionálisan hogy és hol lehet használni, miért nem a legjobb megoldás a modern 32 bites mikrovezérlők esetében?
  32. Mi az a GDB, milyen jellegű parancsok használhatóak a GDB-ben a target debuggolására? Mi az a GDB RSP, milyen tulajdonságai vannak, miért játszik fontos szerepet a beágyazott rendszerek debuggolásában!
  33. Mi az az Open-OCD, milyen főbb komponensekből áll, mire kell odafigyelni, amikor kész Open-OCD forításokat próbálunk használni saját eszközünkhöz?
  34. Mutassa be nagyvonalakban, hogy milyen lépések játszódnak le akkor, amikor egy Eclipse-es GDB, alapú debugger felületen le szeretnénk egy debug állapotban lévő target egy változójának értékét kérdezni az OpenOCD segítségével (nem kellene a parancsok, csak az, hogy milyen rétegeken megy át a kommunikáció és azoknak mi a főbb jellemzői)!
  35. Mutassa be, hogy milyen blokkokból áll a Cortex M3 Coresight debug rendszere, mennyivel nyújtanak ezek a blokkok több lehetőséget egy tradicionális hibakereséshez képest. Mik azok a Trace blokkok, mire képes az ITM, DWT és ETM blokk?
  36. Milyen lehetőségei vannak beágyazott rendszerekben a háttértár funkció létrehozására? Röviden mutassa be az SD kártyák típusait és kezelési módjait, illetve a FAT file rendszert, és annak kezelésére alkalmazható szoftver könyvtárak tulajdonságait!
  37. Mutassa be egy USB csomag általános felépítését! Milyen főbb csoportokba oszthatók a csomagok, mi ezek jellemzője? Jellemezze az USB négy transzfer típusát! Mikor melyiket érdemes/kell használni? Milyen szabályok vonatkoznak az USB keretek transzferekkel való feltöltésére? Mondjon példákat!
  38. Mi az az USB pipe és az endpoint, mi a különbség az IN és OUT típusú endpoint között? Milyen viszonyban van az eszköz-, konfiguráció-, interfész- és a végpont leíró? Melyik "mire való"?
  39. Tipikusan milyen TCP/IP protokollok szükségesek egy beágyazott megvalósításban? Röviden mutassa be ezeknek a szerepét!
  40. Milyen megvalósítási problémákkal egyszerűsítésekkel találkozhatunk egy beágyazott TCP/IP protokoll stack esetében (IP-, ICMP-, TCP korlátok, memóriakezelés,

párhuzamosság)? Kis teljesítményű, kis erőforrású vezérlő esetében miért nem mindegy, hogy TCP, vagy UDP alapú alkalmazási rétegbeli protokollt használunk?