

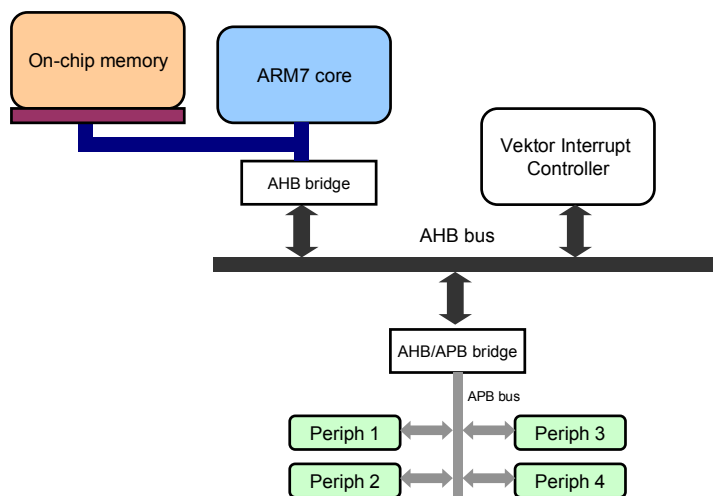
A 32 bites ARM magú mikrovezérlők vizsgálata

Kutatási beszámoló a Pro Progressio alapítvány számára

Scherer Balázs, 2012.

Az első generáció 32 bites ARM7 magú vezérlők belső felépítése

A hagyományos 8 bites mikrovezérlőknél a szokásos rendszer architektúra úgy lett kialakítva, hogy a 8 bites adatbuszra fűztek fel egységesen minden perifériát. Ezeknél az egységeknél jellemzően a perifériák is a processzorral egyező órajelen futnak. A 32 bites mikrovezérlők megjelenésével ez a trend megtört. Már az első verzióknál, mint a 2003-ban megjelent LPC2106-osnál [1] a belső felépítés két buszt tartalmazott a rendszer buszon kívül (1. ábra). A gyors Advanced High-performance Bus-ra (AHB) került a vektoros interrupt controller, míg a lassabb Advanced Peripheral Bus-ra (APB) a perifériák. A külön buszon helyet foglaló perifériáknak esetében, pedig lehetőség volt a busznak az órajelét is befolyásolni. Általában fogyasztástakarékos megfontolásokból nem is a processzor órajelén, hanem annak felén, negyedén hajtották a periféria buszt. (Később az új Cortex-es vezérlőknél is folytatódott ez a tendencia és az újabb, gyorsabb perifériák megjelenésével már nem egy, hanem általában 2 perifériabusz került a rendszerbe.)

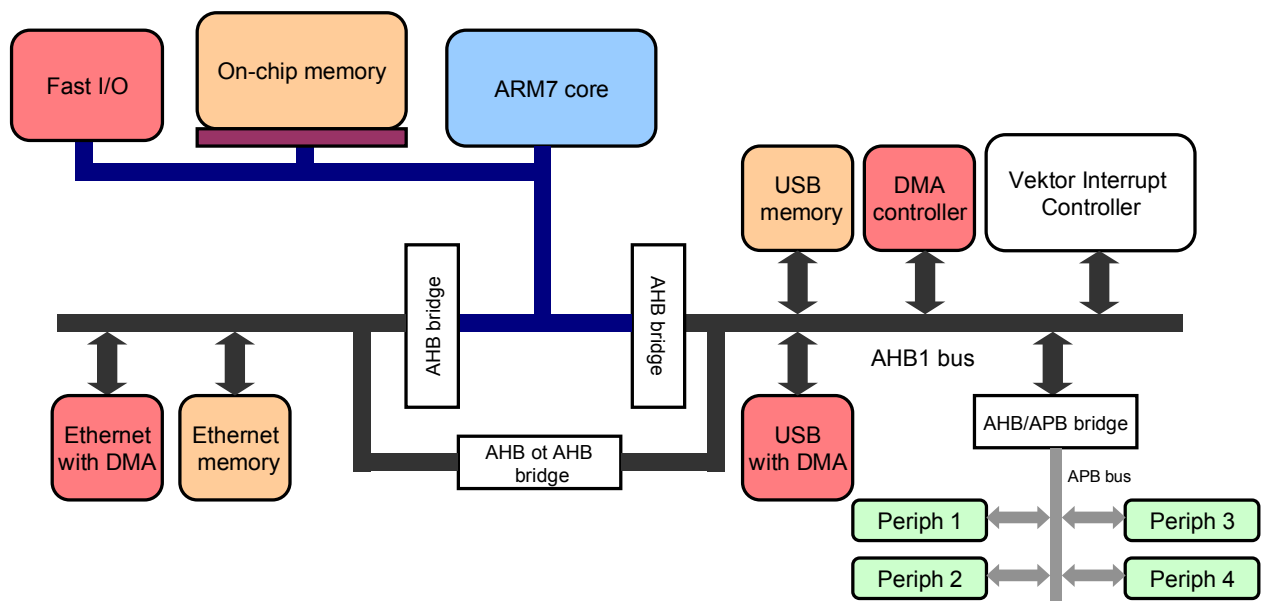


1. ábra: Az NXP LPC2106-os ARM7 magú vezérlőjének belső felépítése

A későbbi ARM7-es sorozatoknál, – amelyek belső felépítését erősen befolyásolta a 2005 körül megjelenő on-chip Ethernet és USB vezérlők – ez a hagyományos felépítés annyiban módosult, hogy a Neumann architektúrás ARM7 mag rendszerbuszához nem egyetlen AHB (Advanced High-performance Bus) csatlakozott, hanem kettő, és mindegyik egy-egy gyors periféria kiszolgálásáért volt felelős. Az NXP sorozatnál külön megemlítendő különbség, hogy bizonyos I/O blokkokat direktben a system buszról hajtottak, így lehetőséget adva a nagyon gyors I/O láb billegtetésre (ez a technológia az NXP védjegye volt). A két megjelenő AHB busznál megfigyelhető, hogy mindkét buszon külön DMA vezérlő helyezkedik el, illetve a két busz között egy AHB bridge is képes kapcsolatot teremteni. Ennek a szellemes elrendezésnek köszönhetően például a mikrovezérlő képes olyan feladatok végezni, mint

például az Ethernet-ről érkező adatok átmásolása az USB-re anélkül, hogy a rendszerbuszt, és így a processzor működését ezzel gátolná. Szintén érdemes megfigyelni, hogy mindegyik dedikált AHB busszal rendelkező perifériának, mint az Ethernet-nek és az USB-nek is létezik saját kisméretű RAM buffere ugyanazon a buszon. Tehát egyszerű üzenetfogadás és üzenetküldés történhet gyakorlatilag párhuzamosan a két nagysebességű periférián DMA alkalmazással, anélkül, hogy ezzel blokkolnánk a processzor működését.

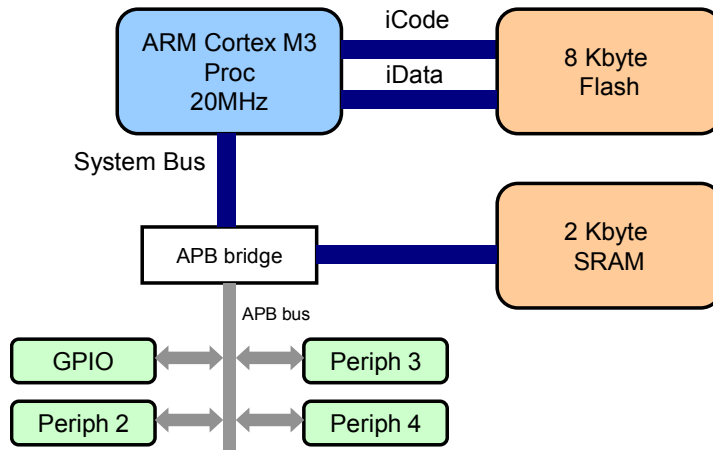
Az 2008-ban megjelenő LPC23xx sorozatnál [2] (2. ábra) is megtartották a lassabb perifériák külön órajel tartományra szeparálását az USB AHB-buszára csatlakozó APB busszal. A periféria bridge-nek az USB AHB buszára helyezése több szempontból is logikus választás volt, egyrészt az USB kevésbé erőforrás igényes, mint az Ethernet, másrészt az USB interfész felőli irányból nagyobb esély van olyan kommunikációs feladatra, amely közeli interakciót igényel valamilyen más perifériával (például SD kártya illesztés mass storage device-ként).



2. ábra: A NXP LPC23xx sorozatú ARM7-es magú vezérlőinek belső felépítése

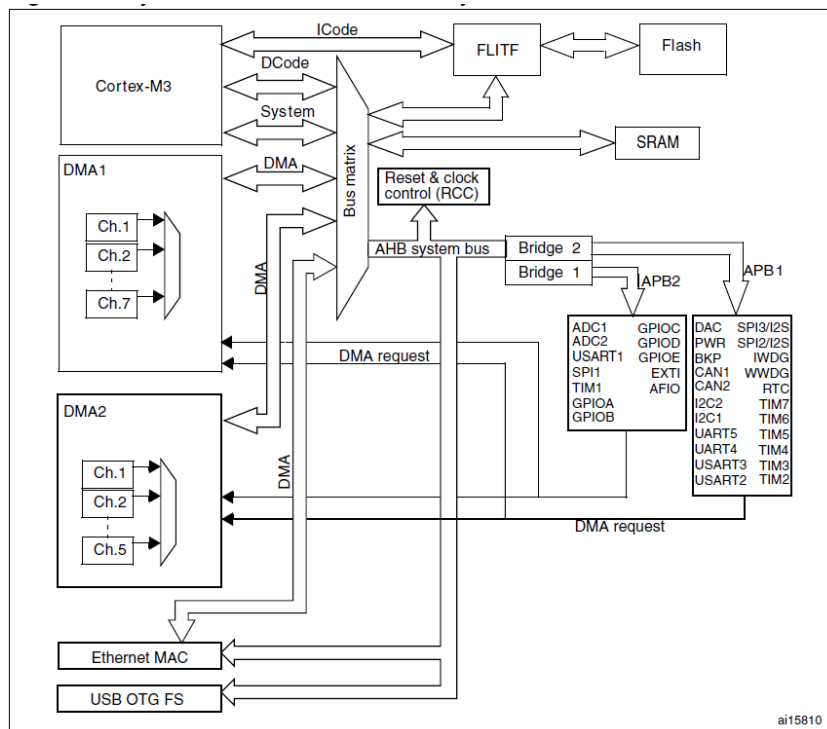
A Cortex M3-M4 magú vezérlők által hozott belső felépítési változások

A 2000-es évek második felében megjelenő Cortex M3 sorozatok jelentős változást okoztak az ARM-magú mikrovezérlők belső felépítésében. A jelentős változás egyrészt az volt, hogy a Cortex M3 mag Harvard architektúrás, így nem egyetlen rendszerbusszal tartja a kapcsolatot a külvilág felé, hanem külön utasítás, adat és rendszerbusszal rendelkezik. További újdonság, hogy a Cortex M3 mag reformjának következtében már többet nem volt szükség külön Vektoros Interrupt kezelőre, hiszen az NVIC része lett a Cortex M3 mikrovezérlő core-nak. Ugyanakkor, ha megnézzük a 2006-ban megjelent első sorozatú Luminary LM3S102 Cortex M3 magú mikrovezérlő [3] belső felépítését (3. ábra) látható, hogy a Harvard architektúrából adódó eltéréseket leszámítva kísértetiesen hasonlít az első ARM7 magú sorozatokhoz. Ez elsősorban azzal indokolható, hogy míg azok az ARM7-es sorozatok az LM3S102 vezérlő is egy „tanulmány” vezérlő rendkívül kevés perifériával, így nem indokolt a belső felépítés elbonyolítása. A későbbi sorozatok, ahol megjelennek az on-chip USB és Ethernet kontrollerek lényegesen bonyolultabb felépítéssel bírnak.

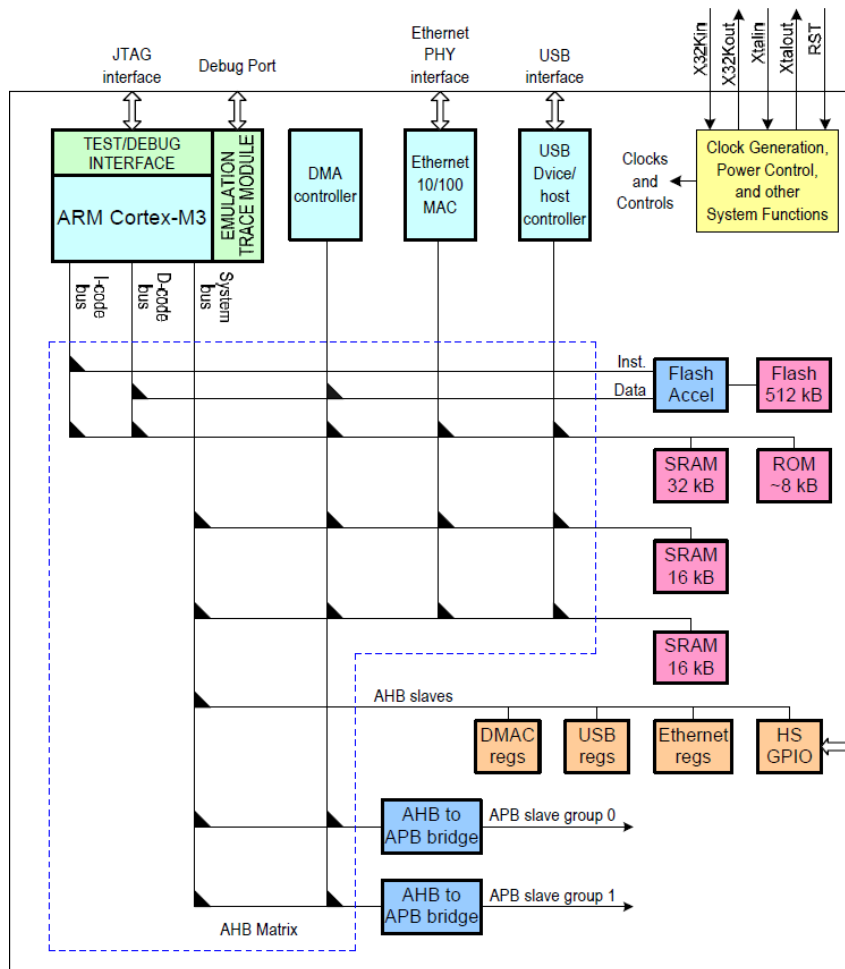


3. ábra: A 2006-ban megjelent Luminary LM3S102 Cortex M3 magú mikrovezérlő belső felépítése

Nézzük meg két 2008 után megjelent vezérlő belső felépítését: A ST Microelectronics STM32F107 vezérlőjének [4] és az NXP LCP1768 vezérlőjének [5] belső felépítését. Ha ránézünk a 4. és 5. ábrára, akkor azt fogjuk tapasztalni, hogy nagyvonalaiiban hasonló a két elrendezés: a gyors perifériák, mint az USB, Ethernet a DMA vezérlővel és a processzorral direktben kapcsolódnak az AHB busz mátrixra, míg a kisebb sebességet igénylő perifériák, mint a CAN, I/O lábak, UART stb., egy bridgen keresztül csatlakoznak egy-egy APB buszra. Mindkét megoldás esetben két APB buszra vannak csoportosítva a perifériák az STM32F1xx sorozatnál például az egyik periféria busz az APB2 72MHz-es sebességgel tud futni, míg a másik az APB1 csak 36MHz-el.



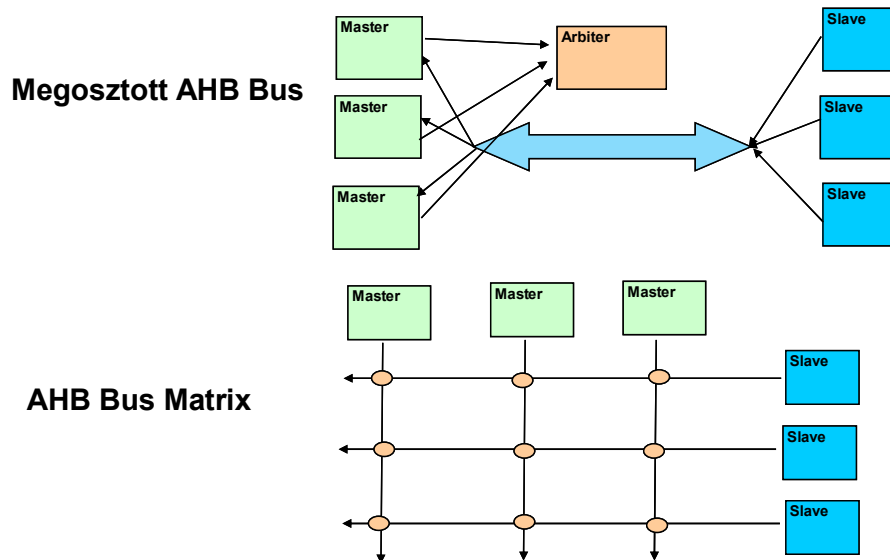
4. ábra: Az STM32F107 belső architektúrája [4]



5. ábra: Az LPC1768 belső architektúrája [5]

Az egyik lényeges eltérés a két vezérlő belső felépítésében, hogy az NXP architektúrájában az USB is, mint AHB busz master jelenik meg, míg az ST-nél ez nem így van (a következő STM32F2xx sorozatnál [6] ez módosul is). Illetve az NXP sorozatnál a gyors I/O lábak az AHB-buszra kerültek.

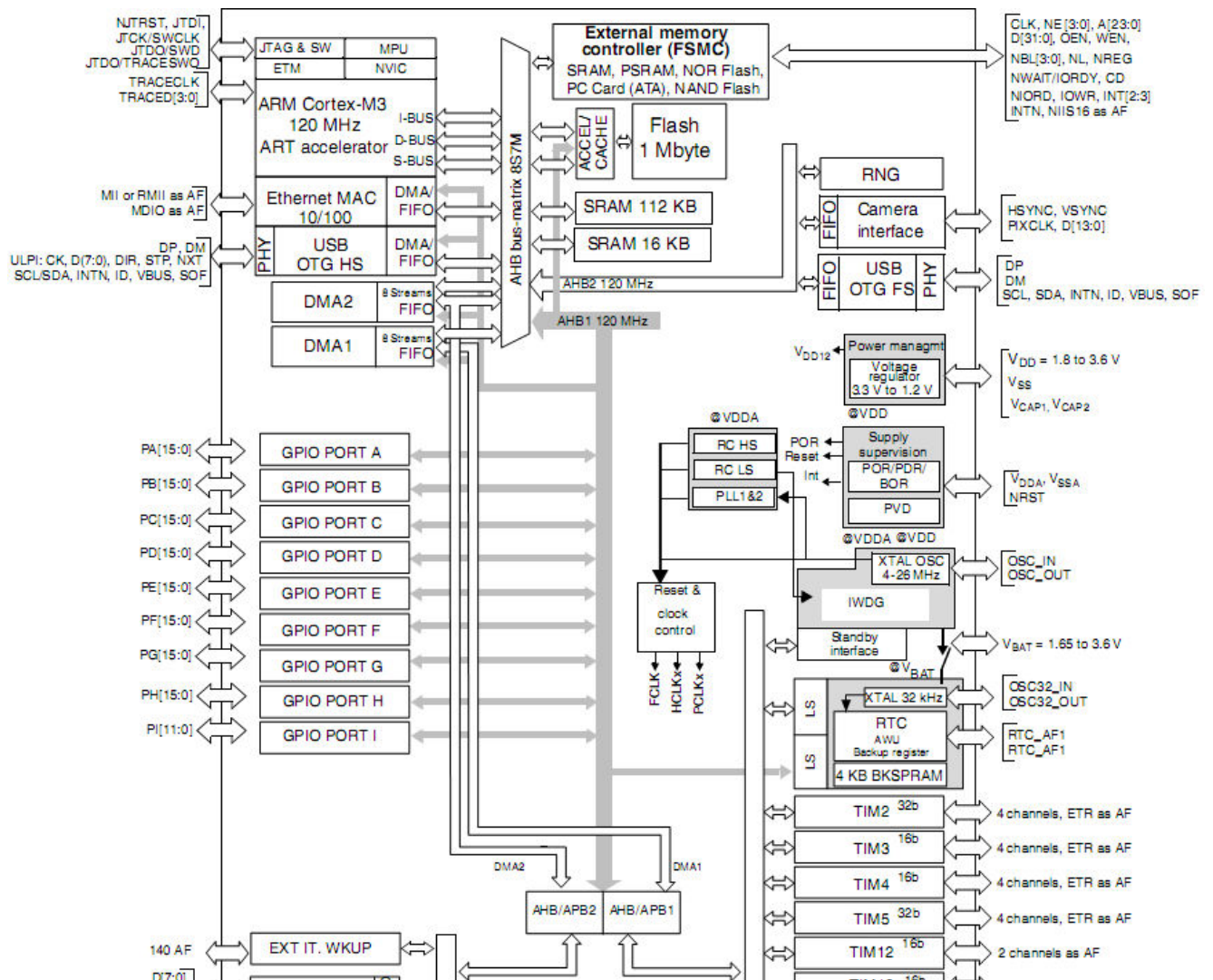
Mindkét megoldásnál az újdonság a busz mátrix bevezetése volt, amely jelentősen javít a DMA-t használó perifériák és a processzor buszhasználatának egymás melletti párhuzamos működésén, ahogy azt a 6- ábra szemlélteti.



6. ábra: Az AHB buszmátrix előnyei

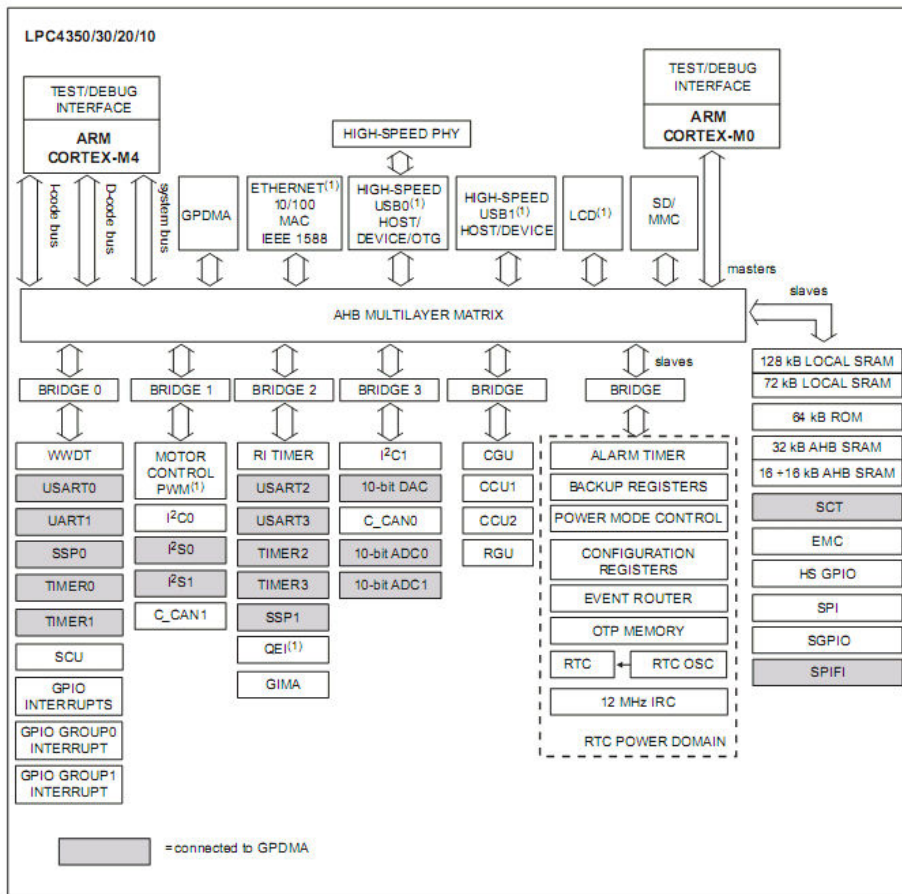
A két vezérlő között ebben lényeges eltérés, hogy az LPC17xx sorozat esetében a memória is több részre van szegmentálva, ezáltal tovább növelve a DMA használat hatékonyságának a növelését, mert míg a processzor az egyik memória szegmensből olvassa ki az adatait, addig egy periféria nyugodtan DMA-zhat egy másik blokkba párhuzamosan, anélkül, hogy blokkolná a processzor hozzáférését. Ezt a trükköt az új STM32F2xx 2011-ben megjelenő sorozatnál is átveszik.

A 2011-es STM32F2xx sorozatnál az ST Microelectronics alaposan áttervezte a vezérlőjének belső felépítését. Hasonlóképpen az NXP sorozathoz az I/O lábak egy gyors AHB buszra kerültek. Az SRAM több szegmensre lett bontva. Illetve a High-speed USB is DMA masterként csatlakozik direktben az AHB busz mátrixra. Ezeket a változtatásokat és néhány új nagysebességű periféria, mint a direkt egy slave-ként egy AHB buszra csatlakozó Camera interfészt megjelenését leszámítva azonban lényeges eltérés nincs az előző sorozathoz képest, ez az elrendezés annak a tapasztalatait felhasználó finomítás.

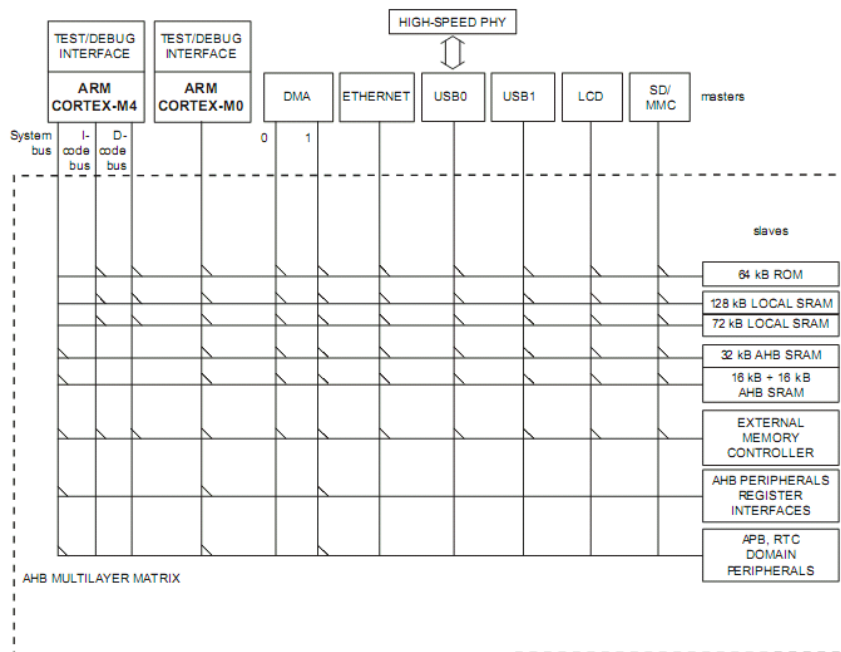


7. ábra: Az STM32F2xx belső felépítése [6]

Jelentős változást a kétmagú mikrovezérlők megjelenése hozta. Erre a 2012 végén megjelenő NXP LPC43xx sorozat az egyik példa. Ezek a vezérlők egy nagy számítású DSP utasításokkal bővített Harvard architektúrás Cortex M4-es magok és egy a perifériák vezérlésére szánt kis komplexitású Neumann architektúrás Cortex M0 magot tartalmaznak. Teljes leírás ezekről a vezérlőkről még nem elérhető, de az előzetes információk alapján a Cortex M0 mag, mint teljes értékű busz master fog csatlakozni az AHB buszra hasonlóképpen a DMA-hoz és a többi nagy sebességű perifériához (8. ábra). Az LPC43xx sorozat [7] belső elrendezésénél jól látható, hogy a perifériatartomány busz bekötése tovább szegmentálódik és egyre több független periféria busz jön létre, amely segít egyrészt az energiafelhasználás optimalizálásában, másrészt a DMA és a két master processzor perifériákhoz való hozzáférését is optimalizálja. A két processzor architektúráis különbsége (a Cortex M4 Harvard architektúrás, míg a Cortex M0-ás Neumann architektúrával rendelkezik) jól látható a 9. ábrán ahol az egyes egységek AHB buszra csatlakozását figyelhetjük meg.



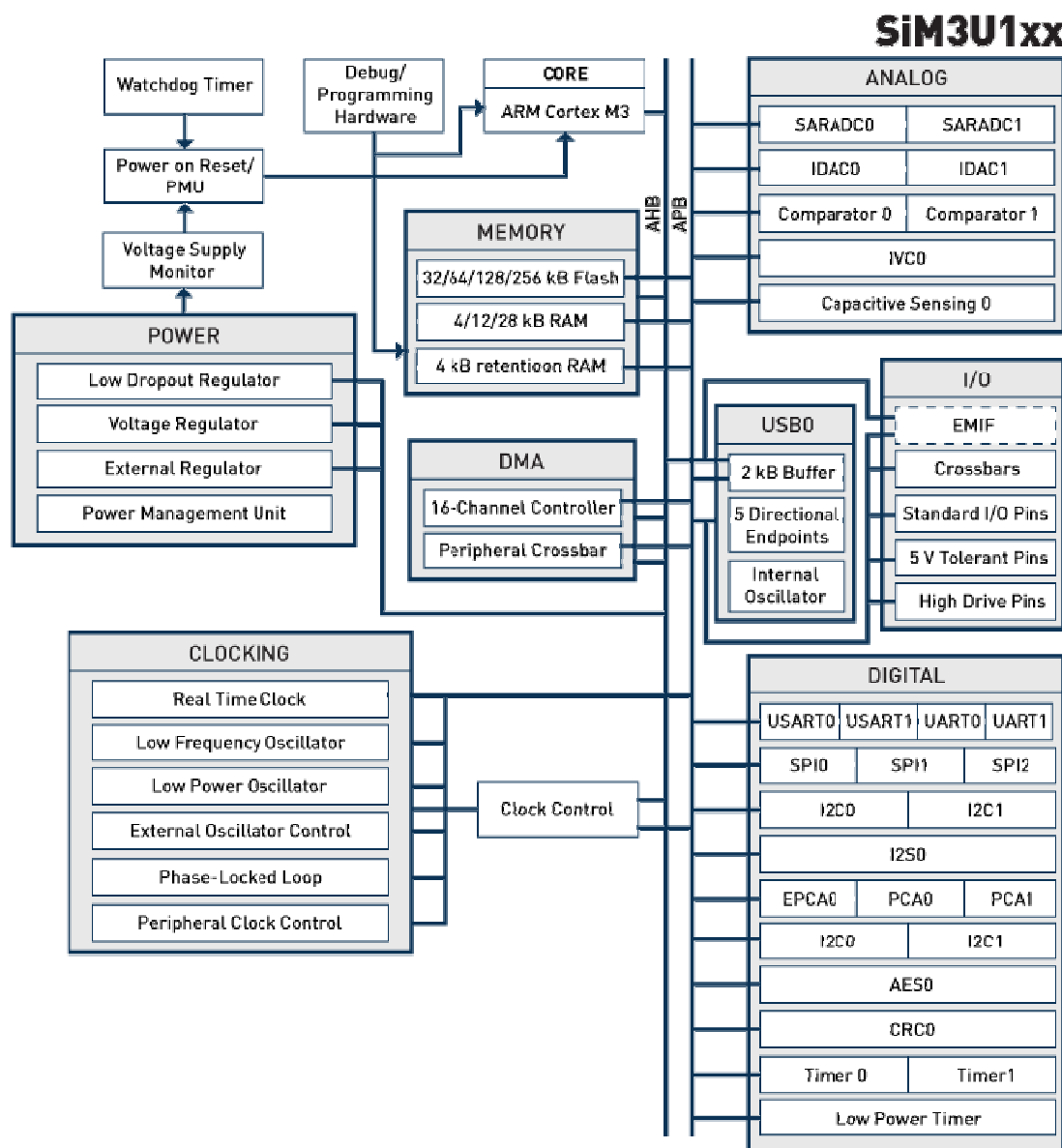
8. ábra: Az NXP LPC43xx sorozatának belső felépítése [7]



9. ábra: Az NXP LPC43xx sorozatának AHB mátrixának belső felépítése [7]

A Silicon Laboratories ARM Cortex magú mikrovezérlői

A Silicon Laboratories 2011 végén lépett a piacra a saját Cortex M3 magú sorozatával. A cég nem az NXP és ST piacvezetők által követett irányt folytatta, hanem a fő célja ennek az új sorozatnak a 8 bites mikrovezérlők kiváltása volt. Ez a törekvés meglátszik a vezérlők piacvezető fogyasztásában, illetve az első 2006-ban megjelent Luminary vezérlőkhöz nagyon hasonló belső felépítésükben. A bonyolultabb Silicon Laboratories által piacra dobott sorozat a SiM3U1xx [8] belső felépítésén is jól látható, hogy az első sorozatú Cortex M3 vezérlőkhöz hasonlóan még nem tartalmaz busz mátrixot, csak egy AHB és egy lassú APB buszt.



10. ábra: A Silicon Laboratories SiM3U1xx sorozatának belső felépítése [8]

A belső felépítés egyszerűségét a dedikáltan a fogyasztásra való optimalizálás és az NXP, ST sorozatokhoz képesti redukált funkcionalitás indokolja. A Silicon Laboratories vezérlő sorozatának jelentős újítása a Dual-Crossbar Architecture, amely segítségével gyakorlatilag

bármely periféria funkcionalitás bármely I/O lábon elérhető. Ez nem szorosan belső architektúrális kérdés, ugyanakkor igen fontos újítás. Szintén több újdonságot hozott a sorozat az energiahatékonyság terén is. A vezérlő feszültsége 1,8V-ig csökkenthető, ami jelentős előrelépés. A belső 4kB „nem elvesző” RAM memória blokk is jóval nagyobb, mint a többi gyártók sorozatai, és még emellett a nem elvesző memória terület mellett is a fogyasztás aktív Real-time Clock mellett 0,35 μ A-re csökkent, ami jelentős előrelépés, és tényleg versenyképessé teszi a sorozatott energiatakarékos megoldásokban is a 8 bites vezérlőkkel szemben. Az egyetlen meglepő a Silicon Laboratories sorozatában, hogyha a cél az egyszerű rendszer architektúra és a 8 bites sorozatok kiszorítása az energiahatékony alkalmazásokból volt, akkor miért nem a még egyszerűbb, olcsóbb és még energiahatékonyabb, ugyanakkor valamennyivel kétségtelenül lassabb Cortex M0 mag lett alkalmazva. Talán azzal még hatékonyabban meg lehetett volna ezt a feladatot valósítani.

Irodalomjegyzék

[1] LPC2106/2105/2104 User Manual, NXP Semiconductors 2003.

[2] UM10211 LPC23XX User manual, NXP Semiconductors 2012.

[3] Stellaris® LM3S102 Microcontroller data sheet, Texas Instruments Incorporated, 2007-2012.

[4] RM0008 Reference manual STM32F101xx, STM32F102xx, STM32F103xx, STM32F105xx and STM32F107xx advanced ARM-based 32-bit MCUs, STMicroelectronics 2011.

[5] LPC1769/68/67/66/65/64/63 32-bit ARM Cortex-M3 microcontroller User manual. NXP Semiconductors 2012.

[6] STM32F205xx, STM32F207xx, Datasheet. STMicroelectronics 2012.

[7] UM10503, LPC43xx ARM Cortex-M4/M0 dual-core microcontroller. User manual. NXP Semiconductors 2012.

[8] SiM3C1xx Technical reference manual. Silicon Laboratories 2012.