

Nagyteljesítményű mikrovezérlők

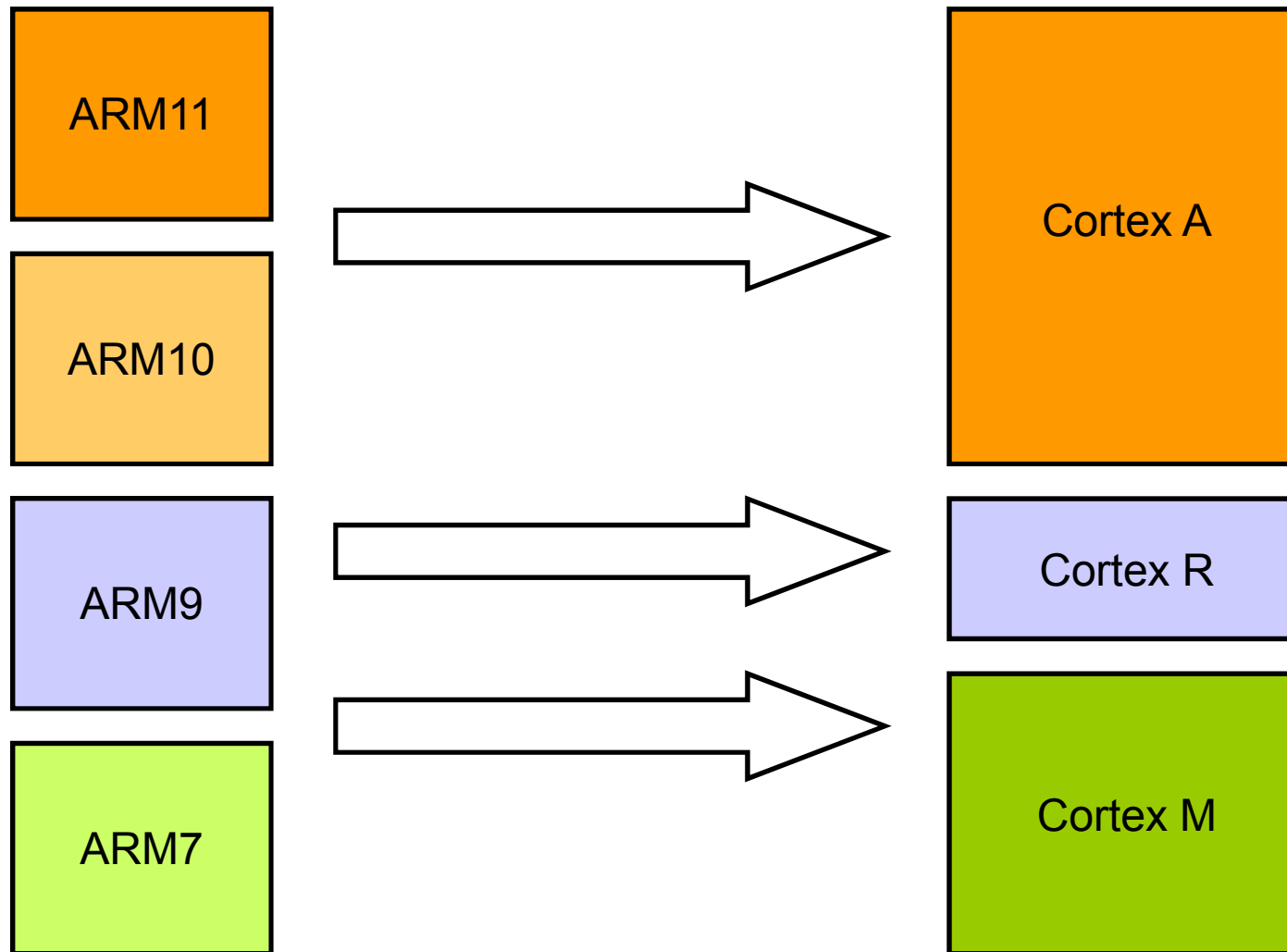
2. ARM7 mag

Scherer Balázs



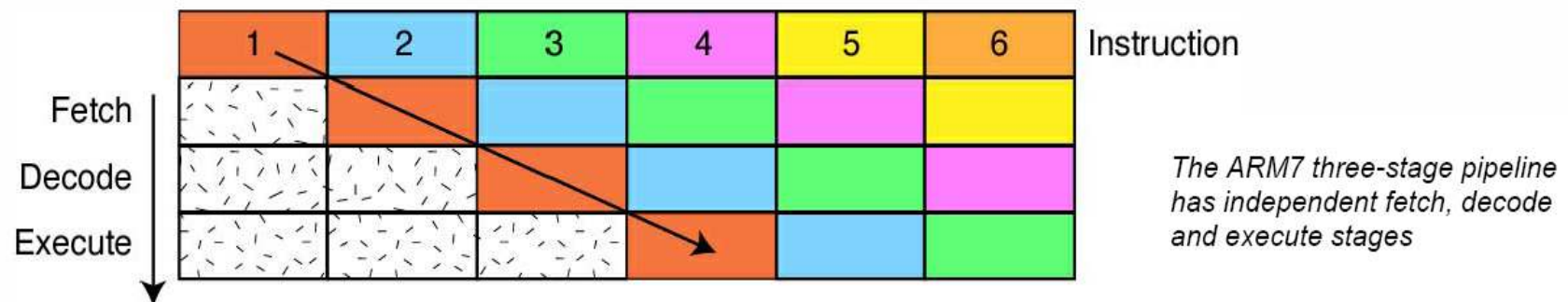
Méréstechnika és
Információs Rendszerek
Tanszék

ARM magok migrációja



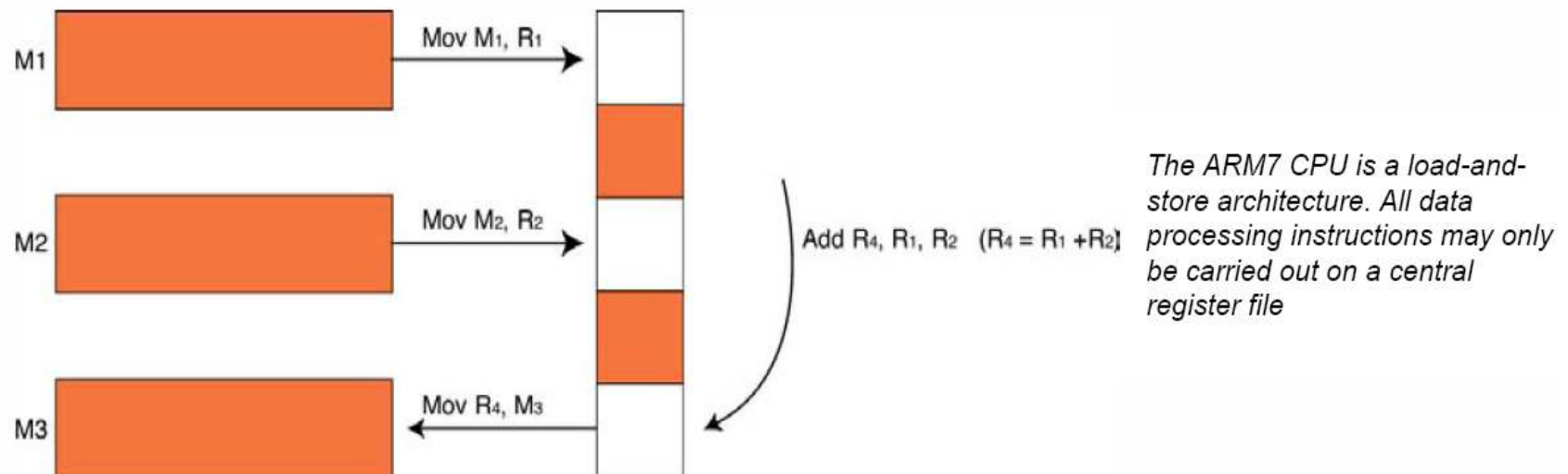
Az ARM 7 mag architektúrája (ARMv4)

- Kevés utasítás, kis komplexitás
 - Kis szilikon felület, gyors, kis fogyasztás
- Pipeline-osított utasítás végrehajtás
 - 3 elemű egyszerű pipeline
 - A legtöbb utasítás egy órajel alatt hajtódik végre
 - A PC 4 byte-al előrébb jár, mint a jelenleg végrehajtott utasítás



Load and Store Architektúra

- Minden adat művelet előtt a memóriából be kell tölteni az adatot egy regiszterbe, azon végrehajtani a műveletet majd kiírni az eredményt



Regiszterek

■ Általános Regiszterek

- 16 általános regiszter

- CPSR

- Operating mode
- IT enable
- Thumb/ARM
- Conditions
 - Zero
 - Negativ
 - Carry
 - Overflow

15 User registers + PC

R0
R1
R2
R3
R4
R5
R6
R7
R8
R9
R10
R11
R12
R13
R14
R15 (PC)

R13 is used as the stack pointer

R14 is the link register

R15 is the Program Counter

Current Program Status Register

CPSR

Működési módok

System & User	FIQ	Supervisor	Abort	IRQ	Undefined
R0	R0	R0	R0	R0	R0
R1	R1	R1	R1	R1	R1
R2	R2	R2	R2	R2	R2
R3	R3	R3	R3	R3	R3
R4	R4	R4	R4	R4	R4
R5	R5	R5	R5	R5	R5
R6	R6	R6	R6	R6	R6
R7	R7_fiq	R7	R7	R7	R7
R8	R8_fiq	R8	R8	R8	R8
R9	R9_fiq	R9	R9	R9	R9
R10	R10_fiq	R10	R10	R10	R10
R11	R11_fiq	R11	R11	R11	R11
R12	R12_fiq	R12	R12	R12	R12
R13	R13_fiq	R13_svc	R13_abt	R13_irq	R13_und
R14	R14_fiq	R14_svc	R14_abt	R14_irq	R14_und
R15 (PC)	R15 (PC)	R15 (PC)	R15 (PC)	R15 (PC)	R15 (PC)
CPSR	CPSR SPSR_fiq	CPSR SPSR_svc	CPSR SPSR_abt	CPSR SPSR_irq	CPSR SPSR_und

The ARM7 CPU has six operating modes which are used to process exceptions. The shaded registers are banked memory that is "switched in" when the operating mode changes. The SPSR register is used to save a copy of the CPSR when the switch occurs

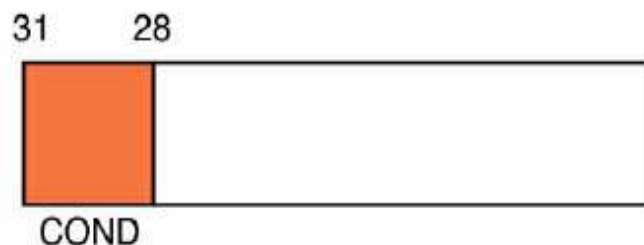
Kivételkezelés

- Alacsony szintű kivételkezelés
 - Interrupt vezérlőt a processzor gyártója ad hozzá
 - Nem túl hatékony, rosszul portolható

Exception	Mode	Address
Reset	Supervisor	0x00000000
Undefined instruction	Undefined	0x00000004
Software interrupt (SWI)	Supervisor	0x00000008
Prefetch Abort (instruction fetch memory abort)	Abort	0x0000000C
Data Abort (data access memory abort)	Abort	0x00000010
IRQ (interrupt)	IRQ	0x00000018
FIQ (fast interrupt)	FIQ	0x0000001C

ARM Utasításkészlet

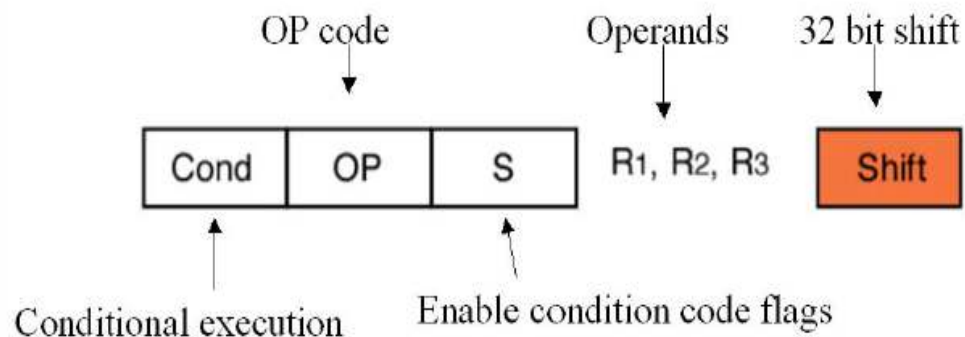
- Little-endian/Big-endian: általában előre fixált (LPC little)
- 32-bites ARM,16-bites THUMB (30% kisebb, de 40%-kal lassabb)
- Minden ARM utasítás feltételfüggő gyakorlatilag minden egyszerű assembly utasításnak 16 variánsa van
- Segíti a pipeline folyamatos kihasználását



Every ARM (32 bit) instruction is conditionally executed. The top four bits are ANDed with the CPSR condition codes. If they do not match the instruction is executed as a NOP

ARM Utasítás típusok I.

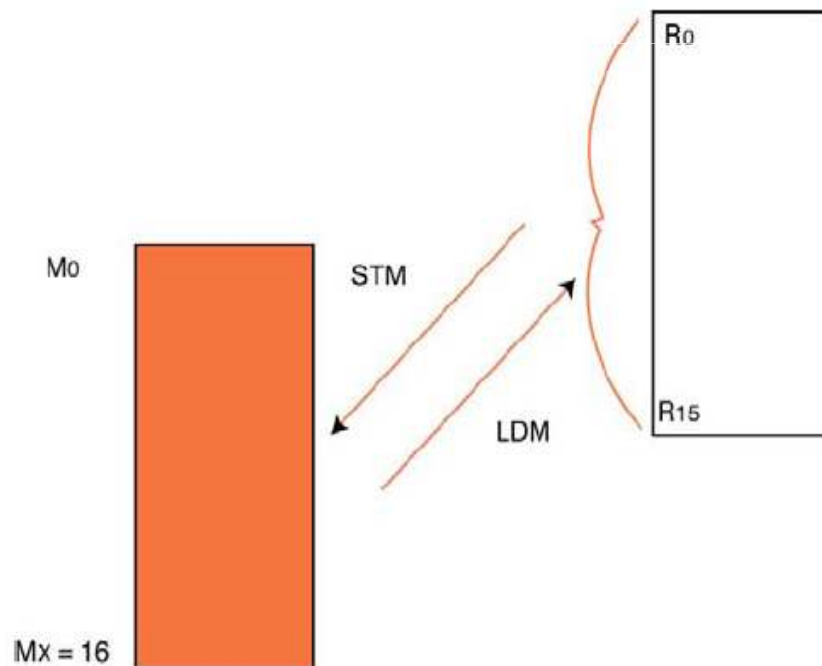
- Branching: Ugró utasítások
 - Link regiszter mentéssel, vagy nélkül
 - ARM/THUMB mód váltási lehetőség
- Data processing
 - Regisztereken hajtódnak végre
 - AND, OR, XOR, ADD, SUB, CMP...
 - If (z==1) R1 = R2+(R3x4)
 - EQADDS R1,R2,R3,LSR #2



The general structure of the data processing instructions allows for conditional execution, a logical shift of up to 32 bits and the data operation all in the one cycle

ARM Utasítás típusok II.

- Utasítás típusok *folytatás*
 - Copying Registers
 - 8,16,32 bites értékek bemásolása, és kiírása
 - Copying Multiple Registers
 - Az egész regiszter bank kiírása, bemásolása



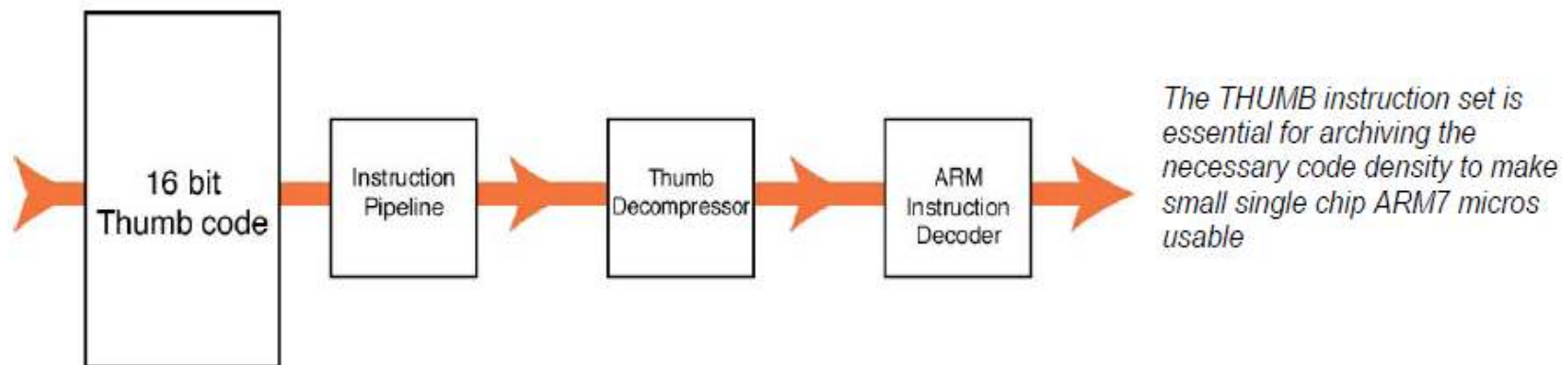
The load and store multiple instructions allow you to save or restore the entire register file or any subset of registers in the one instruction

ARM Utasítás típusok III.

- Utasítás típusok *folytatás*
 - Swap
 - Két regiszter értékének atomikus módon való felcserélése
 - Software Interrupt
 - Supervisor módba ugrik a processzor
 - MAC unit
 - Multiply 32/64 bit-es eredmény (signed / unsigned)
 - Multiply + accumulate 32/64bit-es eredmény (signed / unsigned)

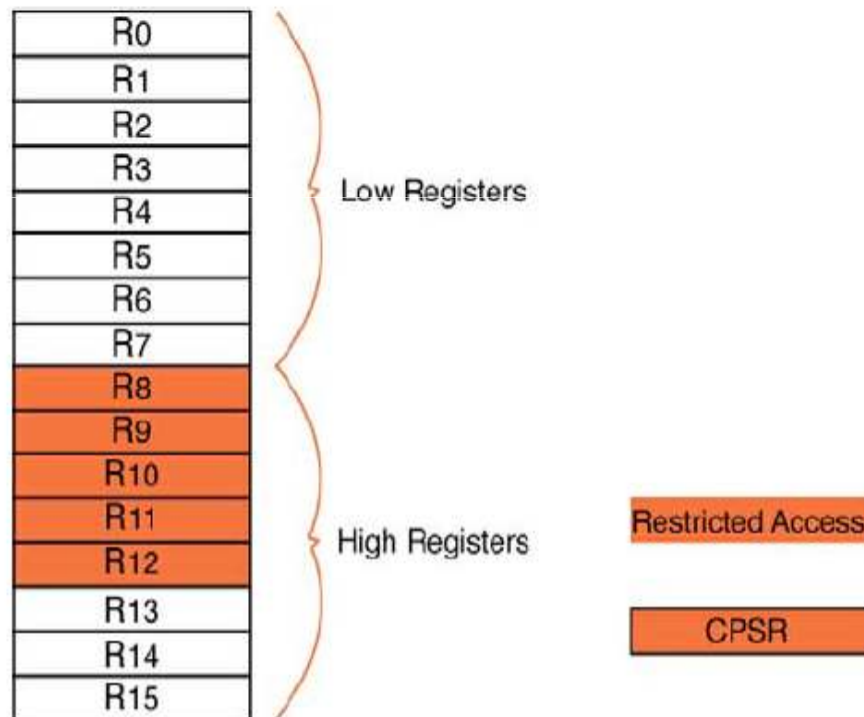
Thumb mód I.

- Másodlagos 16-bites utasításkészlet (30% kisebb, de 40%-kal lassabb)
- A processzor mindig ARM-ban indul programból lehet átváltani Thumb-ra



Thumb mód II.

- Az adatfeldolgozó utasítások legtöbbször csak az R0-R7 regisztereket látja



In the THUMB programmers' model all instructions have access to R0-R7. Only a few instructions may access R8-R12

Fordítók hatása a kódra

Parameter	Compiler			
	Keil CA BETA	GNU V3.22	ARM ADS V1.2	IAR V4.11A
Execution Speed (μ Seconds)	25.4	112.9	16.8	24.4
Dhrystones/sec	39,370.1	8,857.4	59,382.4	40,983.6
Total Code Size (bytes)	10,330	36,004	22,266	19,892
Stack Size (bytes)	208	852	608	356
Total Data Size (bytes)	10,256	11,912	10,256	10,269

All tests were performed under identical conditions using the Keil μ Vision Simulator. The ARM device used was a Philips LPC2294 running at 60MHz in Thumb Mode.