

# *ARM arhitektura i Oracle*

Zlatko Sirotić, univ.spec.inf.  
ISTRA TECH d.o.o., Pula

- ISTRA TECH je novo ime (od 2015.) poduzeća **Istra informatički inženjering**, osnovanog 1990. godine.
- Radim na informatičkim poslovima od 1984. godine.
- Oracle softverske alate (baza, Designer CASE, Forms 4GL, Reports, Java) koristim oko 25 godina.
- Objavljivao sam stručne radove na kongresima / konferencijama HrOUG, JavaCro, CASE, KOM, "Hotelska kuća", te u časopisima "Mreža", "InfoTrend" i "UT".
- Neka moja programska rješenja objavljuvana su na web stranicama firmi Oracle i Quest (danas dio firme Dell).
- Od 2012. sam vanjski suradnik na Fakultetu informatike Pula.
- 2020. godine nije se održao HrOUG, pa mi je najznačajniji osobni događaj bila uspješna operacija srca (2 premosnice).  
**Veliko hvala svima u KBC Rijeka!**

- Prvi put predavač na **HrOUG 2002.** (jedino održano u Puli).
  
- Predavanje se zvalo **TRI IZ III**  
(tri PL/SQL rješenja iz poduzeća Istra informatički inženjering):
  1. Rješavanje "COMMIT poslovnih pravila" na Oracle bazi
  2. Odgođena deklarativna ograničenja baze i Forms
  3. Simulacija ROLLBACK TO SAVEPOINT ponašanja u okidaču baze
  
- Sudjelovao sam 17 puta do sada (uključujući ovu godinu; 2016. i 2017. izostao zbog bolesti) i održao 22 predavanja.
  
- Uz 22 prezentacije, za HrOUG sam napisao 15 radova, ukupno preko 300 stranica teksta.

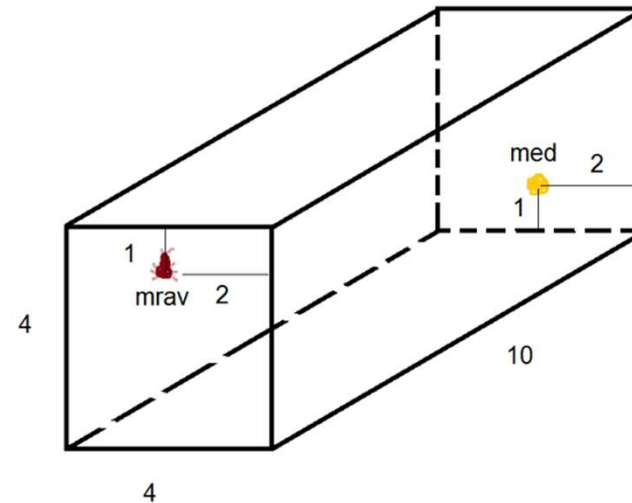
# HrOUG predavanja iz prošlog desetljeća

- **2011. a) Konkurentno programiranje – Oracle baza, Java, Eiffel (najbolje ocijenjeno predavanje 16. konferencije)**
- 2011. b) Kriptografija u Oracle bazi
- 2012. a) Ima neka loša veza - priča o in-doubt distribuiranim transakcijama
- 2012. b) Visoka konkurentnost na JVM-u
- **2013. Transakcije i Oracle - baza, Forms, ADF (63 stranice – od tada sam odustao od pisanja radova :)**
- 2014. Nasljeđivanje je dobro, naročito višestruko - Eiffel, C++, Scala, Java 8
- 2015.a) Povratak u Prolog - verzija 1
- 2015 b) Kada Oracle naredba nije serijabilna?
- 2018. Testiranje konkurentnih transakcija
- 2019. Funkcijska paradigma i baze podataka

## Mrav i med na prizmi (i valjku)

- Na HrOUG 2015. prvi put sam spomenuo zadatak Mrav i med na prizmi (verzija 0), unutar predavanja Povratak u Prolog.

Spomenuo sam ga (verzija 1) i 2019. na početku predavanja Funkcijska paradigma i baze podataka.



- Na stranici <http://www.istrattech.hr/category/blog/> nalazi se najnovija verzija (2), uz još neka predavanja:
  - Mrav i med na prizmi - verzija 2
  - Povratak u Prolog - verzija 2
  - Strukturna složenost algoritama

- ARM ISA arhitekture i mikroarhitekture
- Podjela posla u proizvodnji ARM čipova i mobitela
- Apple Macintosh sada ima ARM čip - M1; Apple Rosetta 2
- Oracle širi ponudu cloud rješenja sa ARM procesorima
- Izazov 1: nestašica čipova
- Izazov 2: Nvidia želi kupiti firmu Arm
- Izazov 3: Memorijski model relaksirane konzistencije
- Problemi emulacije softvera (naročito paralelne emulacije) na različitim ISA arhitekturama
- Poboljšanja specifikacije ARM arhitekture



<= logo ARM arhitekture

logo firme Arm =>



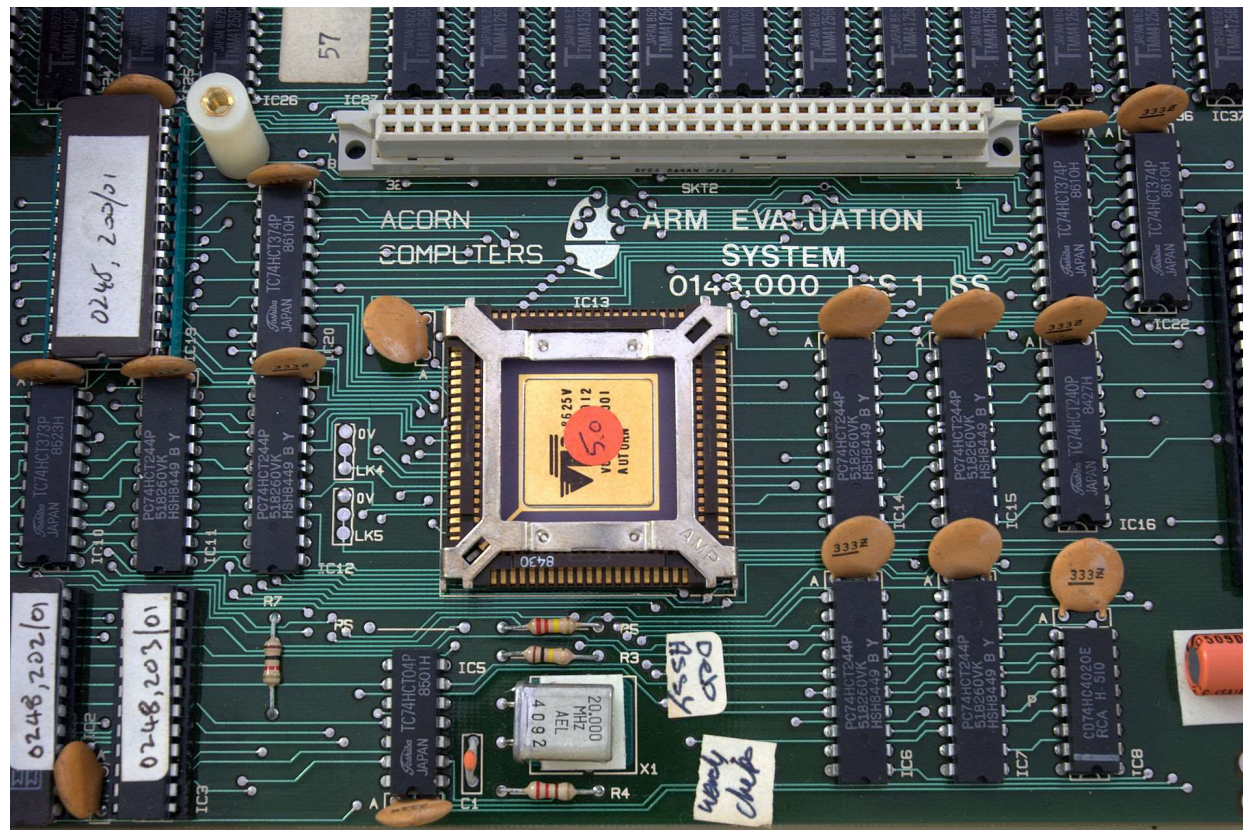
# ARM ISA arhitekture i mikroarhitekture

- Gotovo svi današnji pametni telefoni koriste procesorske čipove temeljene na jednoj od ARM ISA arhitektura.
- Budući da skraćenica ISA već sadrži slovo A, koje znači Arhitektura (Instruction Set Architecture), donekle je redundantno reći ISA arhitektura, ali je često razumljivije.
- Britanska firma Arm Ltd. **dizajnira ARM procesorske arhitekture** (množina!) iz određene ARM porodice, npr. arhitektura ARMv8.2-A iz porodice Cortex-A (64-bit).
- Firma Arm **dizajnira i procesorske jezgre (tj. mikroarhitekture)** temeljene na određenoj arhitekturi, npr. jezgra Cortex-A78 iz arhitekture ARMv8.2-A.
- Firma Arm je jedan od "preživjelih potomaka" firme Acorn. Akronim ARM je prvo korišten 1983. i tada je značio "Acorn RISC Machine", a od 1990. znači "Advanced RISC Machines" (kako se zvala i firma Arm od 1990. do 1998.).



## Prvi ARM čip - ARM1 iz 1985.

- Prvi primjerci Acorn ARM čipova proizvedeni su 1985. godine, pod imenom ARM1. Radili su na taktu od 6 MHz. Prvo su se primjenjivali kao sekundarni procesor za razvojno računalo izvedeno iz računala BBC Micro.



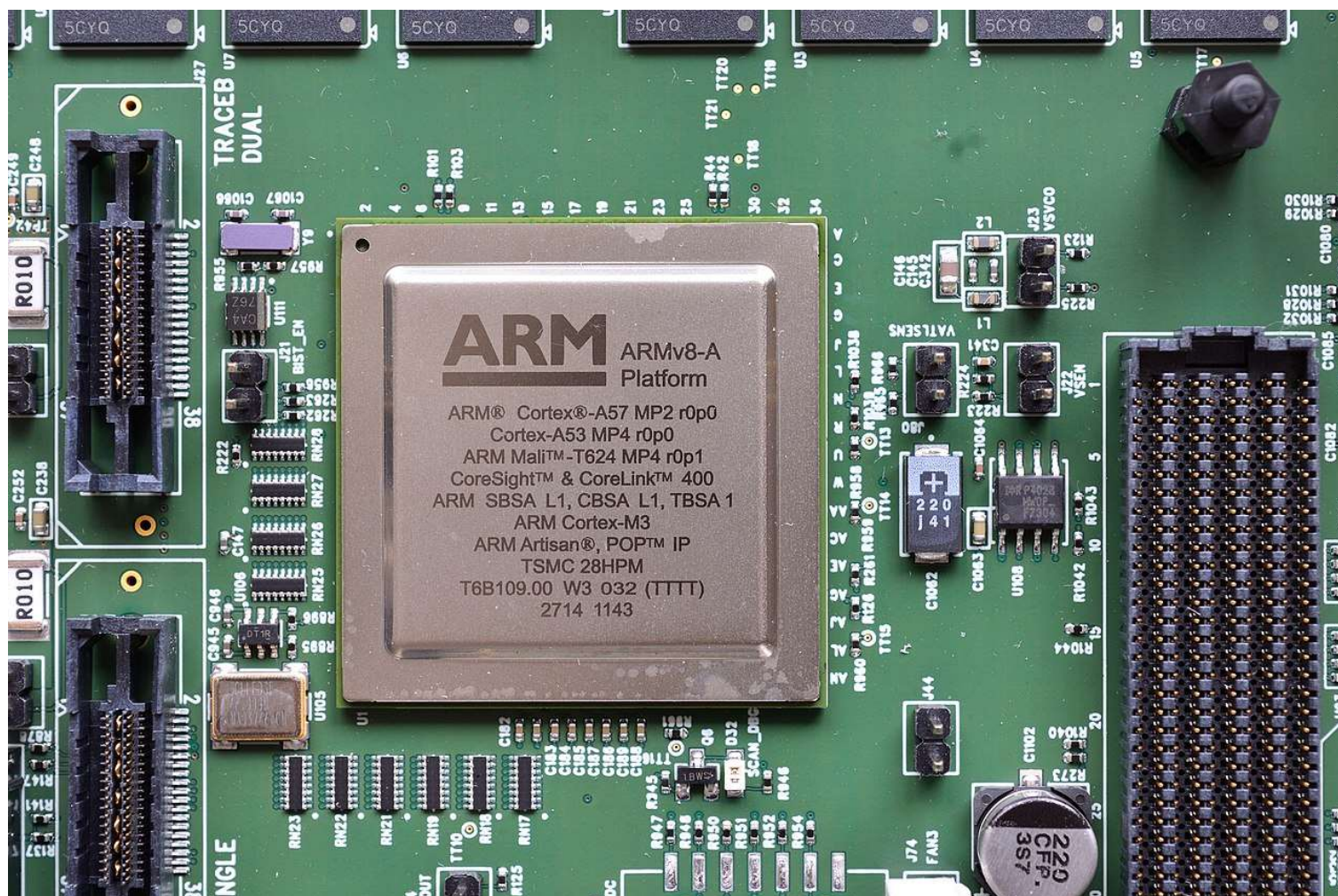


# Evolucija ARM arhitekture

## Architecture Evolution



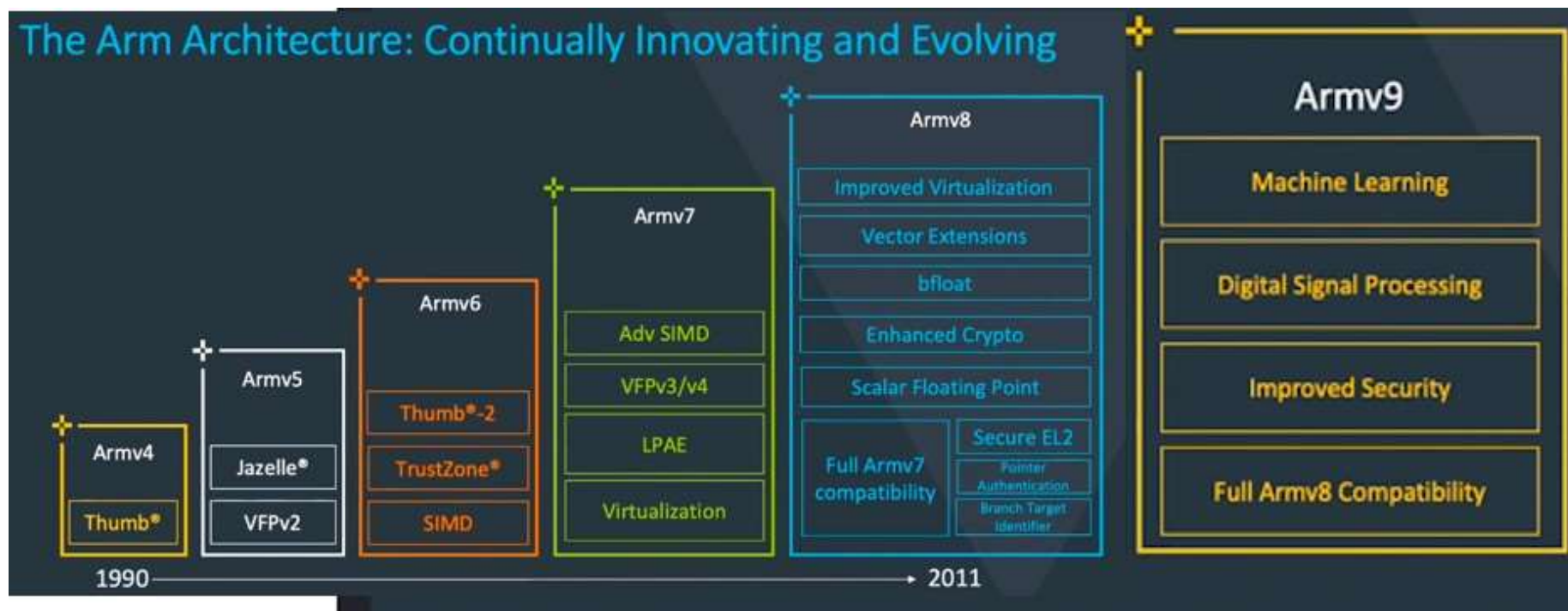
- Na čipu piše koja je arhitektura (ARMv8-A), CPU jezgre (Cortex-A57 i A53), GPU jezgre (ARM Mali-T624) ... ali i ime proizvođača (TSMC).





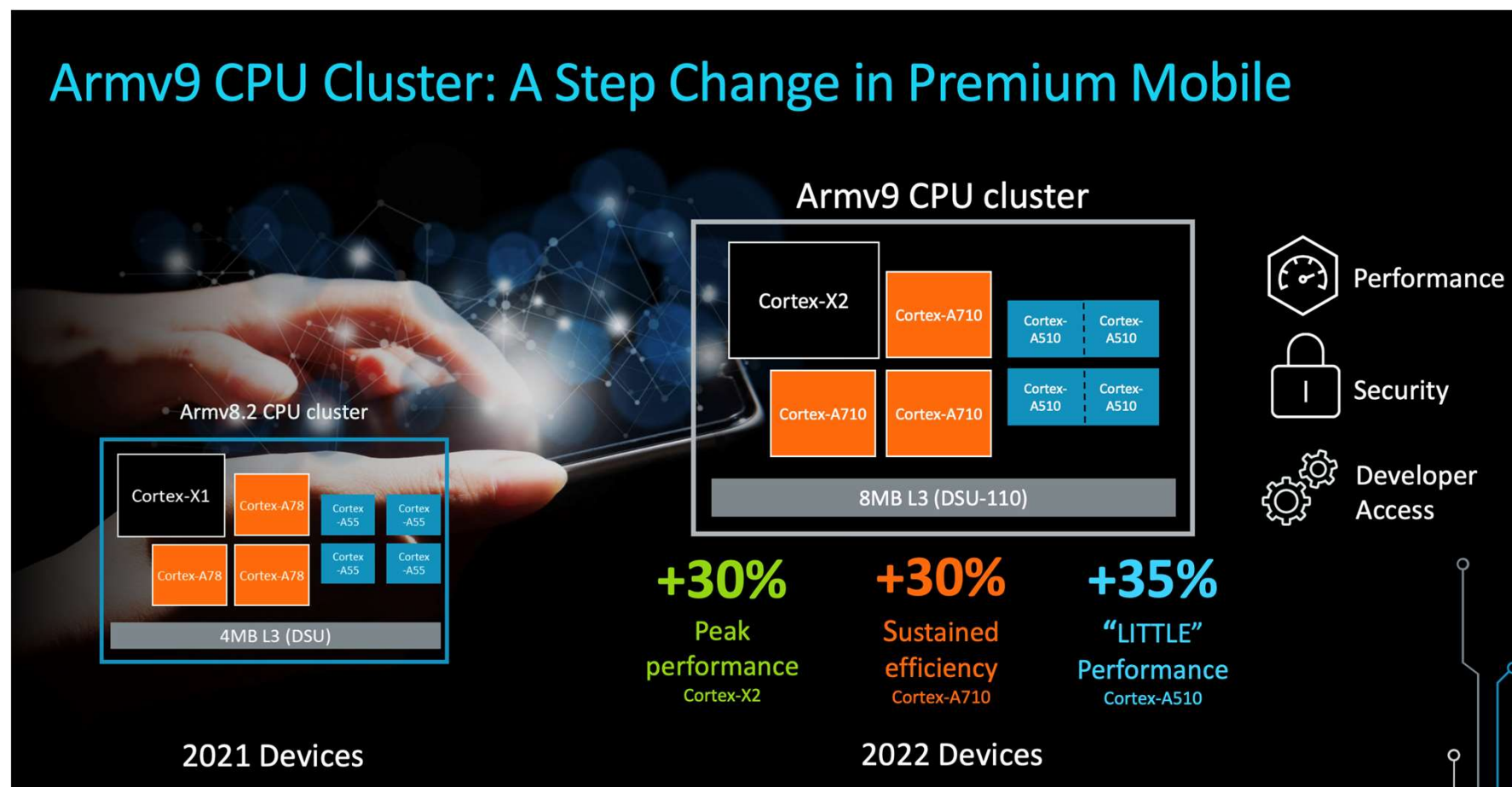
## Ove je godine izašla ARM v9 arhitektura

- 10 godina nakon izlaska ARMv8 arhitekture, ove je godine izašla ARMv9 arhitektura.



## Usporedba performansi v8 i v9 arhitekture

- Kao i kod arhitekture v8, i kod v9 arhitekture Cortex jezgre dijele se na **big**: vrhunska X2 i standardna A710, i **little** (slabijih performansi, ali štedljive) A510.



# Podjela posla u proizvodnji ARM čipova i mobitela

- Druge firme na temelju Arm licenci dizajniraju i vlastite inačice ARM jezgri i **SoC (System on a Chip)** čipove (tako rade npr. Apple, Samsung, Qualcomm), ili dizajniraju samo SoC čipove (npr. Huawei, MediaTek).
- Te čipove često proizvode treće firme. Najbolju (5 nm) tehnologiju proizvodnje procesorskih čipova za sada imaju tajvanski TSMC ("svjetski prvak") i Samsung Electronics.
- Na temelju SoC čipova, četvrte firme mogu dizajnirati npr. mobitele (ali, ti se čipovi mogu koristiti i za druge namjene), a pete firme mogu proizvoditi mobitele.
- Koliko nam je poznato, jedina firma koja na temelju ARM arhitekture radi i dizajn jezgri i dizajn SoC čipova, sama proizvodi čipove, te dizajnira i proizvodi mobitele, je Samsung Electronics.

# Podjela posla u proizvodnji čipova i mobitela

## - primjeri proizvođača i proizvoda

	Redizajn ARM Jezgre	Dizajn SoC čipa	Proizvodnja SoC	Dizajn mobitela	Proizvodnja mob.
<b>Apple</b>	Da	Da: <i>M1 (Mac)</i> <i>A15 (iPhone 13)</i>	Ne	Da	Ne
<b>Samsung</b>	Da	Da: <i>Exynos 2100</i>	Da	Da	Da
<b>Qualcomm</b>	Da	Da: <i>Snapdragon 888</i>	Ne	Ne	Ne
<b>MediaTek</b>	Ne	Da: <i>Dimensity 1200</i>	Ne	Ne	Ne
<b>Huawei</b>	Ne	Da: <i>Kirin 9000</i>	Ne	Da	Da

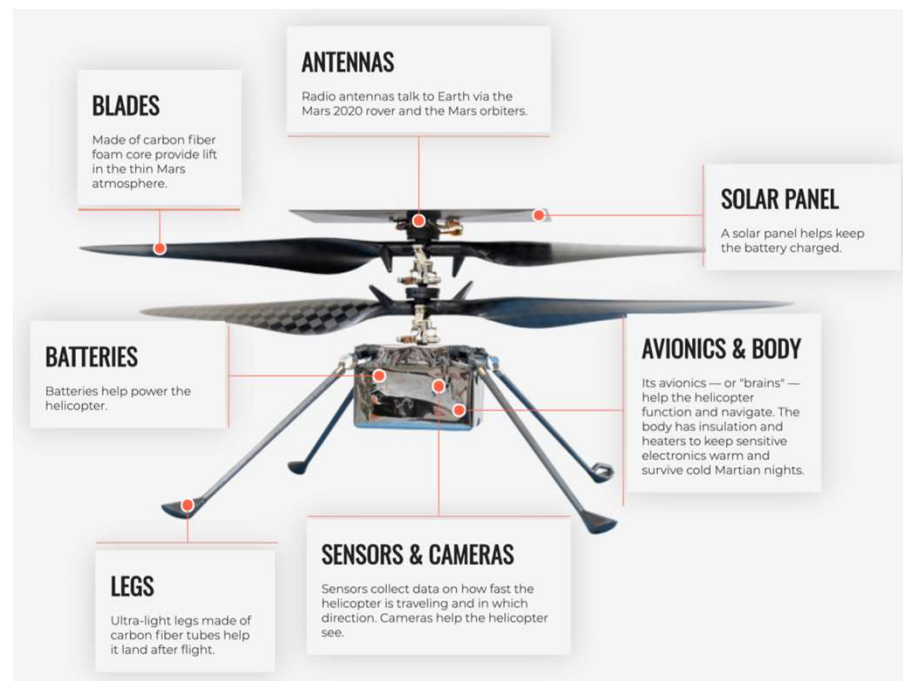
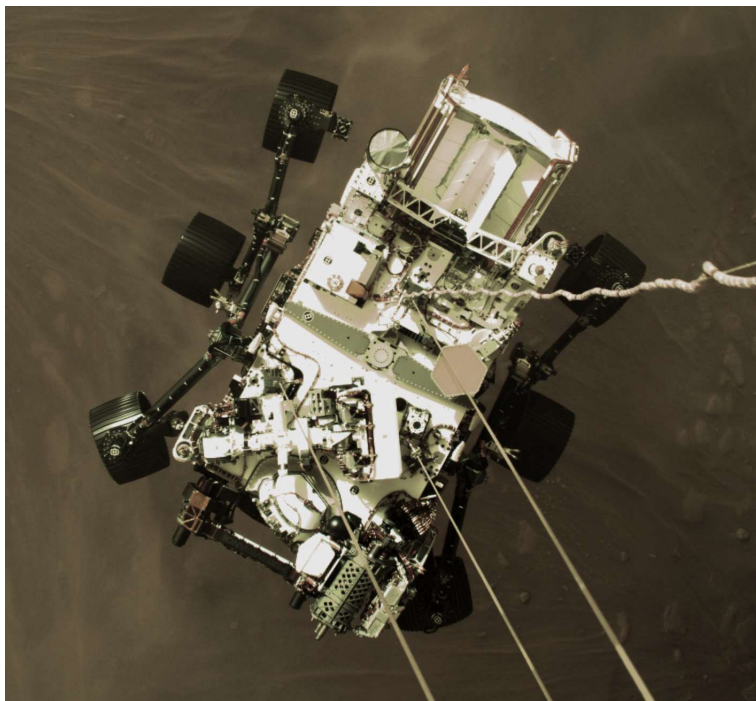
## ARM čipovi se ne koriste "samo" u mobitelima

- Donedavno su se procesori ARM arhitekture u razmišljanjima većine ljudi povezivali "samo" s mobitelima.
- No činjenica je da se ARM arhitektura već dosta davno probila i na servere.
- Npr. **najjače današnje superračunalo Fugaku** (po TOP500 rangiranju), koje je dizajnirao i proizveo japanski **Fujitsu**, ima **158 976 komada 48(+4)-jezgrenih čipova A64FX**, koje je (kao i njegove jezgre) također dizajnirao Fujitsu, na temelju arhitekture ARMv8.2-A
- Taj je čip proizveo TSMC, u 7 nm tehnologiji. Fujitsu je prije toga za svoja računala i superračunala koristio vlastiti čip SPARC64 V.



## ARM čipovi se ne koriste "samo" u mobitelima

- Perseverance rover nema ARM čip - koristi BAE Systems RAD750 računalo (iz 2001.), zaštićeno od zračenja, temeljeno na 32-bitnom IBM PowerPC 750 mikroprocesoru.
- Ali, **helikopter** koristi **Qualcomm Snapdragon 801** procesor, koji pokreće Linux OS. Čip je ARMv7, 32-bitni, iz 2014.

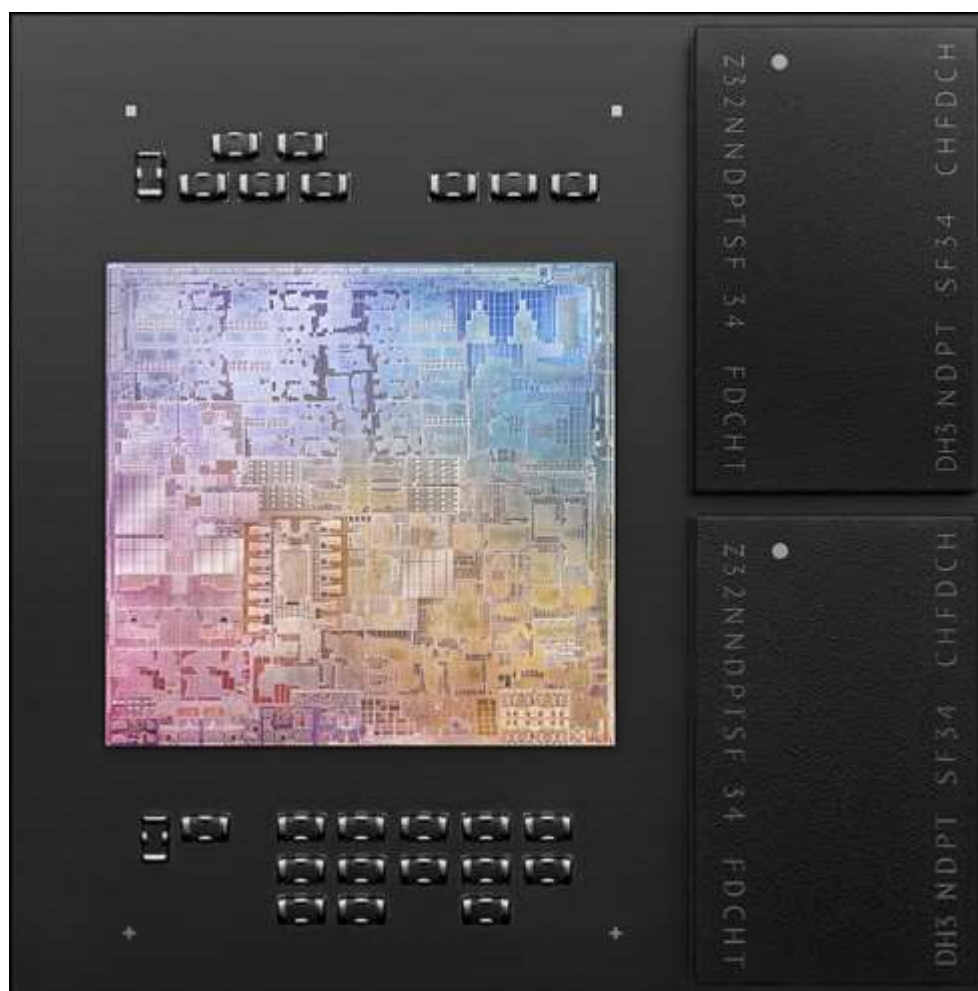


## Apple Macintosh sada ima ARM čip - M1

- Veliko zanimanje za ARM čipove u laptop/desktop korisničkom svijetu uslijedilo je krajem prošle godine, kad je Apple uspješno prešao sa Intelove arhitekture na ARM arhitekturu i kod računala Macintosh. Prije toga je Apple koristio ARM arhitekturu "samo" za mobitele i tablete.
- Ovo je **Appleova peta procesorska arhitektura** (četvrta kod Macintosh računala). Apple I i Apple II računala imala su MOS Technology 6502 8-bitni procesor. Machintosh je prvo koristio Motorola 68000 procesor, pa IBM PowerPC (od 1994.), pa Intel (od 2006.) i sada ARM.
- U Macintosh se ugrađuje **SoC M1** (proizvodi ga TSMC, u 5 nm tehnologiji), kojega je (kao i dosadašnje čipove za mobitele) dizajnirao sam Apple. M1 čip ima CPU jezgre koje je također dizajnirao Apple, na temelju arhitekture ARMv8.4-A, i to četiri jače jezgre Firestorm i četiri štedljivije jezgre Icestorm.

## Apple Macintosh sada ima ARM čip - M1

- Uz 8 CPU jezgri, M1 sadrži i 7 ili 8 GPU jezgri, te 16 Neural Engine jezgri. M1 SoC je upakiran zajedno s 2 DRAM čipa (8 ili 16 GB) u SiP (system-in-a-package).



## Apple Rosetta 2

- Uspjehu Appleovog prelaska na novu arhitekturu sigurno je doprinio i Appleov novi dinamički binarni translator Rosetta 2.
- Prvu verziju, imena Rosetta, Apple je napravio 2006., kod prelaska sa IBM PowerPC procesora na Intelove procesore.
- **Rosetta 2 omogućuje da se binarni kod, pisan za Intelovu ISA arhitekturu x86-64, izvršava na procesoru M1, koji ima ARM arhitekturu.**
- Pritom se translacija binarnog koda pokušava napraviti odmah kod instaliranja programa, tj. pokušava se napraviti **AOT translacija (ahead-of-time)**, kako bi program radio brzo i kod prvog pokretanja.
- U slučaju da AOT translacija (dijela) koda nije moguća, koristi se JIT translacija (just-in-time). Nažalost, postoje i programi koji se moraju preraditi da bi radili.

## Apple Rosetta 2

- No iako je Rosetta 2 omogućila da se većina softvera pisanog za Macintosh računala sa Intel procesorskom arhitekturom uglavnom bez problema (i dovoljno brzo) izvršava na novim Macintosh računalima sa ARM arhitekturom, **ne bi trebalo misliti da je taj prelazak uvijek lako izvesti.**
- **ARM arhitektura, u odnosu na Intelovu, koristi memorijski model koji je više relaksiran.**
- Zato nije lako raditi emulatore / translatore koji omogućavaju izvođenje programa pisanih za "konzervativnije" arhitekture (npr. x86-64) na računalu koje ima više relaksirani (slabiji) memorijski model, kao što ju ima ARM arhitektura.

## Koje čipove danas podržava Oracle baza

- Ako pogledamo stranicu za download Oracle baze 19c, vidjet ćemo da postoje zip datoteke za sljedeće sustave:
  - Microsoft Windows x64 (64-bit)
  - Linux x86-64
  - Oracle Solaris (SPARC systems, 64-bit)
  - IBM AIX
  - HP-UX ia64
  - Linux on System z (64-bit)
  
- Ako zanemarimo OS, **podržane procesorske arhitekture** su:
  - **x64** (ili x86-64, x86\_64, AMD64, Intel 64) = CISC arhitektura
  - Oracle **SPARC** = RISC
  - **Power ISA** (OpenPOWER Foundation; prije IBM) = RISC
  - **IA-64** (Intel Itanium) = VLIW (very long instruction word)
  - **IBM z/Architecture** = CISC (za mainframe računala)

## Koje čipove danas podržava Oracle baza

- Budući da
  - IA-64 (Intel Itanium) se prestao prodavati ove godine;
  - **Oracle SPARC M8 pojavio se još 2017. i nakon toga nema najave novih verzija;** a i Oracle partner Fujitsu je umjesto SPARC procesora svoje najnovije superračunalo Fugaku izgradio na ARM procesorima;
  - i ako znamo da samo IBM dizajnira računala temeljena na svojim IBM z procesorima, pa i POWER procesorima;
  - **vidimo da se priča uglavnom svela na jednu arhitekturu - x64, u dvije varijante: Intel ili AMD.**
- Dakle, izgleda kako je uvođenje ARM arhitekture u Oracle svijet vrlo očekivan korak.



□ Evo što kaže sam Oracle na stranici:

<https://www.oracle.com/cloud/compute/arm/why-arm-processors/>

## Why Arm Processors?

With the introduction of 64-bit computing in the Armv8 architecture and Neoverse family, Arm has entered the server market with partners like Ampere Computing.

Virtualization of hardware and the emergence of cloud computing have accelerated this transition.

**In our industry, we are at an inflection point, where once again multiple CPU architectures will become the norm.**

Customers will once again have diversity and choice for building their next generation of applications, and Arm, X86, and GPUs are leading the way.

## ARM i Oracle

- The **Arm Neoverse N1 CPU architecture** is specifically designed for the data center and cloud infrastructure space. It delivers a high core count server-class SoC subsystem with the performance, features, and scalability needed to accelerate the transformation to a scalable cloud-to-edge infrastructure. It fundamentally changes the performance equation for hyperscale data centers and cloud computing, delivering far more work on far fewer watts than conventional technologies.
- **Ampere Computing's Altra processors** are based on the Neoverse N1 CPU architecture. Ampere took the N1 platform and integrated its own innovations to design an SoC uniquely built for applications across hyperscale cloud data centers. The increased thread counts, higher CPU frequencies, and better core densities result of the N1 Arm processors lead to:

## ARM i Oracle

- The increased thread counts, higher CPU frequencies, and better core densities result of the N1 Arm processors lead to:
- **Revolutionary compute performance:** Neoverse N1 Arm CPUs are designed and optimized for high performance and hyperscale data centers. These CPUs deliver full frequency sustained performance and with **single threaded core** architecture you can run workloads with a consistent and predictable performance while achieving ideal scaling.
- **Better security:** Ampere Altra processors **single thread per core processor** design eliminates the potential thread-security issues and provides isolation for customer workloads. This helps lower the risk due to side channel attacks. By running only a single thread per core, there is no sharing of the execution engine, registers, and L1/L2 cache between threads, which minimizes the attack surface for exploits.
- **Designed for a wide range of workloads**

## ARM i Oracle

- **Ampere's Altra processors**, the industry's first 80-core server processor, brings extreme performance, scalability, and power efficiency to drive the next generation of applications.
  - **Up to 160 cores in dual socket configuration**
  - Neoverse N1 Cores Arm v8 Architecture
  - No noisy neighbors—1 thread / core
  - **3.0 GHZ—Run all cores at the maximum frequency**
  - 64 KB L1 I-cache, 64 KB L1 D-cache on each core
  - Up-to 1TB of Memory
  - Coherent Mesh Network
  - AI inference acceleration
  - 7 nm process technology

## ARM i Oracle

- **What is Oracle doing to support Arm server-side development?**
- ... Oracle Cloud Infrastructure (OCI), in collaboration with partners, provides an easy path to transition, build, and run your Arm-based applications with the best price-performance in the cloud. Ampere's Altra processors based on the Arm aarch64 architecture are available on OCI as both bare metal servers and Virtual Machines (VMs).
- Leading ISV developer tool vendors such as GitLab, Jenkins, ... and technology partners such as Arm and Ampere, **are collaborating with Oracle to provide tools and software to build and run server-side Arm-based solutions.**
- Oracle has released **supported aarch64 distributions of Oracle Linux, Java, and MySQL** to jumpstart your development on the Arm compute platform.

## A gdje je Oracle baza na ARM arhitekturi?

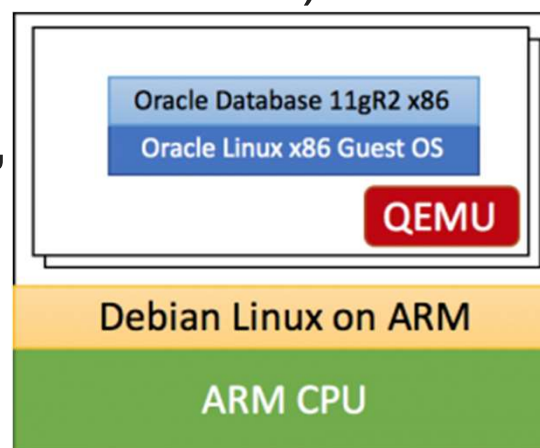
- Teško je to znati nama laicima, ali možda je prilagodba Oracle baze na ARM arhitekturu malo "tvrđi orah"?!  
Na većini sljedećih slajdova prikazat ćemo **jedan od mogućih razloga** zašto taj prelazak nije lagan ("izazov 3").

- Naravno, bilo je pokušaja hrabrih individualaca, koji su probali instalirati Oracle bazu čak i na najmanja ARM računala. Npr.

<https://www.dbarj.com.br/en/2018/06/installing-oracle-database-on-arm-small-single-board-computer/>

- Oracle baza 11.2.0.4 na Asus TinkerBoard (2 GB RAM)

- Budući da Oracle 11g nije podržan na ARMu, koristio je emulaciju: **ARM host emulira x86 guest sustav !!!**



## Izazov 1 – nestašica čipova

- Dobro je poznato da oko godinu dana vlada nestašica različitih vrsta čipova, od najsofisticiranijih procesorskih čipova proizvedenih u 5 nm tehnologiji, do "jednostavnih" i jeftinih.
- Pogođeni su svi tehnološki proizvođači (pa i Apple), ali naročito proizvođači automobila. Pokretani just-in-time filozofijom, u vrijeme početka koronakrize, kad je jako pala prodaja automobila, povukli su svoje narudžbe, vjerujući da će kasnije jednostavno dobiti potrebne količine.
- Nažalost, proizvodnja čipova je "skup sport" – investicije u proizvodne linije (ili nove tvornice) mjere se u desecima milijardi dolara i dugotrajne su.
- Nevjerojatno je da su SAD i Europa pustili da se proizvodnja najsofisticiranijih procesorskih čipova svede uglavnom na TSMC (Tajvan) i Samsung (Južna Koreja). Sad zvuči dobro što je Intel zadržao proizvodnju procesorskih čipova.



## Izazov 2 – Nvidia želi kupiti Arm

- Vlasnik (UK) firme Arm je japanski konglomerat SoftBank, koji ju je 2016. kupio za 32 milijarde \$.
- SoftBank je izgubio puno novaca na lošim financijskim ulaganjima i od 2020. prodaje 90% Arm-a za 40 milijardi \$.
- Apple je prvi kome je to ponuđeno (firma s najviše novaca), ali izjavili su da nisu zainteresirani. Vjerojatno su procijenili da tržišni regulatori nikako ne bi odobrili tu kupovinu.
- **Namjeru za kupnjom dala je Nvidia**, najjači igrač kod GPU čipova, koji se ne koriste samo za igranje i rudarenje kriptovaluta, nego i za "suradnju" s CPU čipovima u serverskim računalima (naročito u cloudu) i superračunalima.
- Posjedovanje ARM arhitekture omogućilo bi da Nvidia ima u svojim rukama obje tehnologije (za GPU i CPU). Time bi Nvidia vjerojatno postala **teško nadmašivi lider u području umjetne inteligencije**, što se konkurenciji ne sviđa.

## Izazov 2 – Nvidia želi kupiti Arm

- Postoji (moguće opravdani) strah da bi Nvidia usporavala svoje konkurente koji dizajniraju procesorske jezgre (i dr.) temeljene na ARM arhitekturi, sakrivajući im bitne informacije ili/i značajno povećavajući cijene licenci.
- To sadašnji vlasnik SoftBank nije radio, jer se on ne bavi proizvodnjom tehnologije, nego financijskim ulaganjima.
- Toj se kupovini jako protivi Qualcomm, tržišni lider u dizajniranju ARM čipova. Jako se protive i Microsoft, Google, Amazon, Tesla, Samsung, dok MediaTek podržava tu kupnju.
- Čistim proizvođačima čipova (kao što je TSMC) ta kupovina vjerojatno ne bi puno značila, jer bi oni i dalje proizvodili sofisticirane čipove, bez obzira tko ih drugi dizajnirao.
- Teško je to znati nama laicima, ali **možda niti na Oracle ne bi značajno utjecala ta kupovina**. Oracle bi i dalje nabavljao ARM čipove – on ih ne dizajnira (naravno, niti ne proizvodi).

## Izazov 2 – Nvidia želi kupiti Arm

- Postoje razmišljanja da će se, ako Nvidia uspije kupiti firmu Arm i ako značajno poveća cijenu licenci ili na drugi način oteža rad drugih dizajnera ARM jezgri i čipova, desiti **"prebacivanje" težišta s ARM arhitekture na open source RISC-V arhitekturu.**
- Naravno, takav prelazak ne bi bio brz i lagan (nije riječ "samo" o hardveru, nego i o pratećem softveru), ali nužda tjera na velike promjene.
- U svakom slučaju, osim ljute konkurencije, postoje i tržišni regulatori, koji odlučuju o toj kupovini. To su regulatori u UK (koji su odložili kupovinu), SAD-u, Europskoj uniji i Kini.
- Kako sada stoje stvari, čini se da ta kupovina neće uspjeti, iako je (uz vlasnike i menadžere firme Nvidia) jako podržavaju i menadžeri i zaposlenici firme Arm.

## Izazov 3

- Zbog nastojanja za povećanjem performansi programa, većina današnjih procesora implementira **relaxed memory consistency model** - memorijski model relaksirane (opuštene) konzistencije.
- To znači da **procesori mogu mijenjati redosljed instrukcija**. To je hardversko mijenjanje redosljeda instrukcija u toku izvođenja, a ne softversko!
- Podsjetimo se da i optimizatori (kod kompajliranja), mogu softverski mijenjati redosljed instrukcija u odnosu na onaj koji je napisao programer.
- **Različiti proizvođači imaju različite modele konzistencije memorije. Zapravo, čak i isti proizvođač može imati više modela, jer oni ovise o ISA arhitekturi.**

## Relaxed memory consistency model

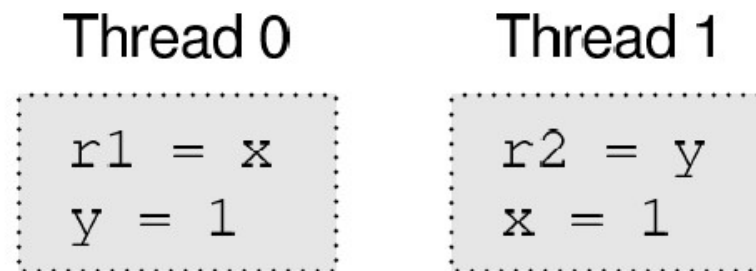
- Najintuitivniji model konzistencije memorije je **sekvencijalna konzistentnost** (Sequential Consistency), kod koje se naredbe procesora izvršavaju striktno **atomarno** (atomically) i u redoslijedu koji je specificiran programom.
- Sekvencijalna konzistentnost se u praksi izbjegava, jer onemogućava hardversku optimizaciju programa.
- Sljedeća tablica prikazuje različite modele konzistencije memorije, kod različitih današnjih procesora (tj. ISA).
- Kvačica uz određeno ograničenje (W → R order, W → W order, R → RW order) pokazuje da je to ograničenje relaksirano, tj. da ga se procesor ne mora pridržavati.
- Vidi se da (poslije Sequential Consistency arhitekture) **najmanje je relaksirana x86 arhitektura, a najviše su relaksirane SPARC-RMO, POWER i ARM.**

# Relaxed memory consistency model

Relaxation	$W \rightarrow R$ order	$W \rightarrow W$ order	$R \rightarrow RW$ order
SC			
x86-TSO	✓		
SPARC-TSO	✓		
SPARC-PSO	✓	✓	
SPARC-RMO	✓	✓	✓
POWER	✓	✓	✓
ARM	✓	✓	✓

# Relaxed memory consistency model

- Primjer dvije OS dretve koje sadrže dvije jednostavne instrukcije (gornja slika), te različite moguće rezultate izvršavanja kod x86 i POWER arhitekture (donja slika). x i y na početku imaju vrijednost 0. Izvor: [1]



Result	x86-TSO	POWER
$r1 = 0, r2 = 0$	✓	✓
$r1 = 0, r2 = 1$	✓	✓
$r1 = 1, r2 = 0$	✓	✓
$r1 = 1, r2 = 1$	✗	✓



## Relaxed memory consistency model – kod paralelizacije sekvencijalnih programa

- Koristeći memorijski model relaksirane konzistencije, u cilju povećanja performansi programa, proizvođači procesora pokušavaju postići (automatsko) **paralelno izvršavanje sekvencijalnih programa**.
- Da bi omogućili korektno izvršavanje sekvencijalnih programa, primjenjuju dvije različite tehnike.
- Jedna je korištenje **instrukcija za sinkronizaciju ogralom (memory fence instructions, skraćeno fences)**, koje može programer ručno dodati u program, ili ih (najčešće) automatski dodaje kompajler u fazi optimizacije.
- Druga je primjena **hardverske transakcijske memorije (HTM)**, kod onih procesora (za sada rijetkih) koji podržavaju HTM, također ili ručno ili automatski (kod kompajliranja).

## Problemi emulacije softvera na različitim ISA arhitekturama, te (pokušaji) rješenja

- Kod emulacije softvera na istoj ISA arhitekturi, gdje virtualni stroj (gost) ima isti memorijski model kao i pravi stroj (domaćin), nema problema.
- Problemi se mogu javiti kod **emulacije softvera na različitim ISA arhitekturama (cross-ISA system emulators)**.
- Ako domaćin ima manje relaksirani (jači) memorijski model u odnosu na gosta, opet nema (značajnih) problema.
- No ako domaćin ima više relaksirani (slabiji) memorijski model u odnosu na gosta (npr. ako se na procesoru POWER ili ARM arhitekture emulira x86 virtualni stroj), **moglo bi se desiti da se program izvršava nekorektno.**
- Naravno, to se ne smije dopustiti!

## Problemi emulacije softvera na različitim ISA arhitekturama, te (pokušaji) rješenja

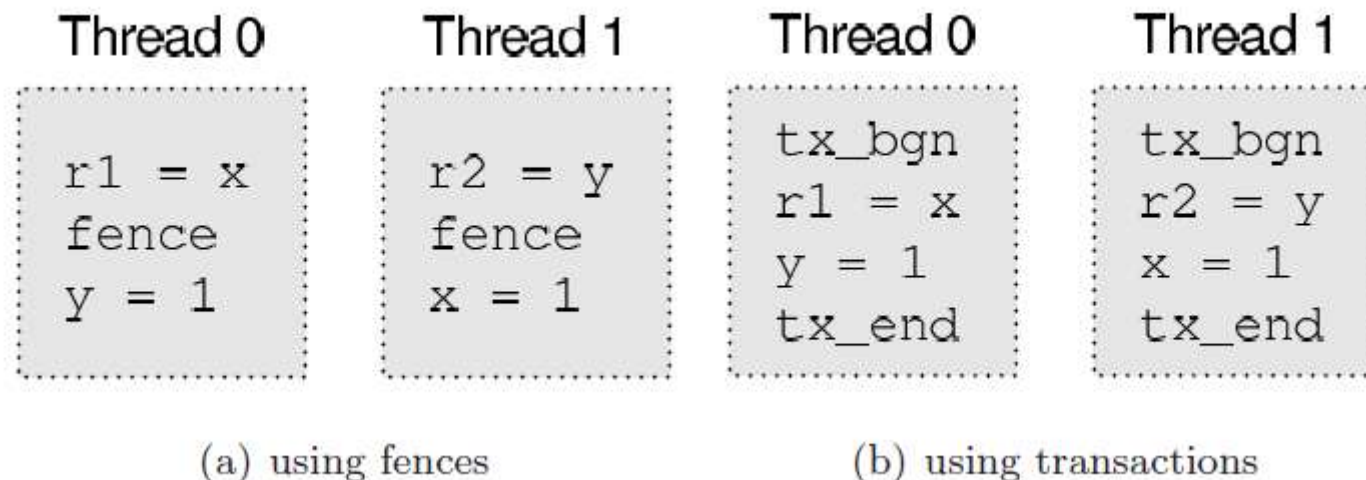
- Jedno rješenje za postizanje korektnosti izvršavanja kod emulacije softvera na različitim ISA arhitekturama je **izbjegavanje paralelizacije.**
- Ako domaćin ima više relaksiran memorijski model u odnosu na gosta, programi koji su na izvornoj platformi (platformi gosta) višedretveni (pa se mogu izvršavati paralelno, na više hardverskih dretvi), izvršavaju se tada na domaćinu sekvencijalno, koristeći **timesharing.**
- **Treba napomenuti da su prije čak i emulatori između istih ISA arhitektura radili na sekvencijalan način.** Naravno, time se gubi na performansi, jer se softver ne izvodi paralelno.

## Problemi emulacije softvera na različitim ISA arhitekturama, te (pokušaji) rješenja

- Današnji (paralelni) emulatori više ne rade tako, tj. oni koriste paralelizaciju kod emulacije.
- No **ograničeni su na emulaciju samo unutar iste ISA arhitekture** (npr. x86 na x86), ili (u najboljem slučaju) **na emulaciju kod koje gost ima više relaksirani (slabiji) memorijski model** u odnosu na domaćina (npr. ARM na x86).
- Postoje i pokušaji (manje ili više uspješni) realizacije paralelizacije u slučaju kada gost ima manje relaksirani (jači) memorijski model u odnosu na domaćina.
- Kao i kod automatske paralelizacije sekvencijalnih programa, tada se isto može primijeniti sinkronizacija ograndom (fences) ili HTM, ali sugeriraju se i **hibridni pristupi** (primjena fences i HTM-a).

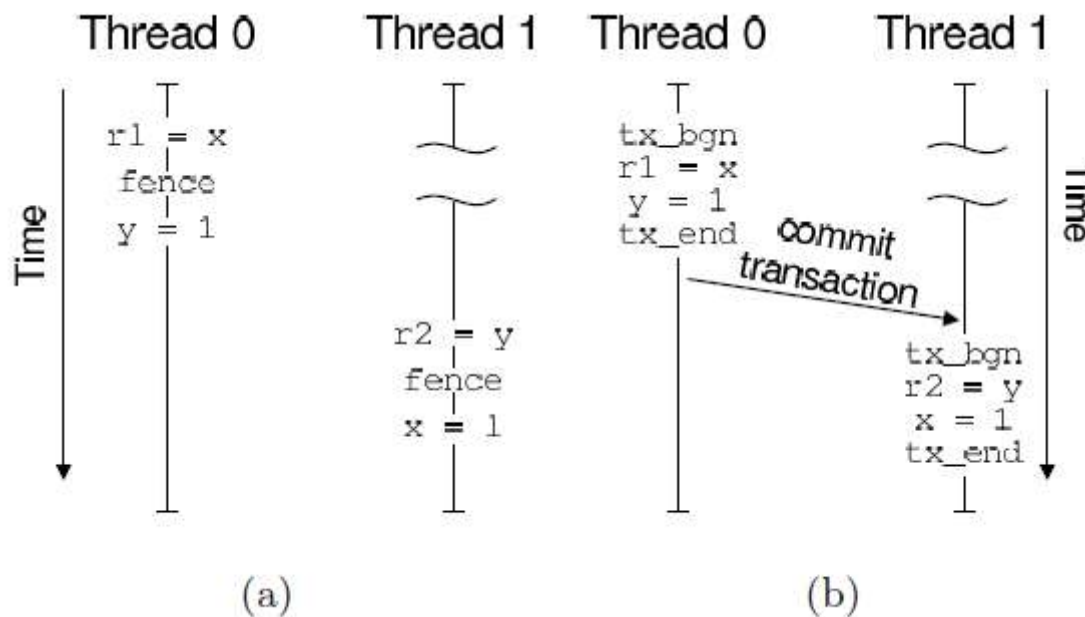
## Problemi emulacije softvera na različitim ISA arhitekturama, te (pokušaji) rješenja

- Slika prikazuje korektnu implementaciju x86 emulacije na POWER arhitekturi (na primjeru s 14. slajda).
- Budući da je x86 arhitektura manje relaksirana od POWER arhitekture, **optimizer dodaje instrukcije za sinkronizaciju ogralom (a) ili za početak i kraj hardverske transakcije (b):**



# Problemi emulacije softvera na različitim ISA arhitekturama, te (pokušaji) rješenja

- Treba napomenuti da je optimizatoru jako teško utvrditi da li u nekom konkretnom slučaju (ne)treba ugraditi instrukcije za fence / transakciju.
- Npr. u ovom slučaju, gdje su se dretve izvele sekvencijalno, instrukcije za fences / transakcije su se izvele nepotrebno:





## Problemi emulacije softvera na različitim ISA arhitekturama, te (pokušaji) rješenja

- R. Natarajan je 2015. u [1] analizirao primjenu fences ili transakcija (realiziranih pomoću HTM-a, konkretno na Haswell procesorima), te primjenu (vlastite) hibridne tehnike.
- Analizirao je dva problema (koja je rješavao na sličan način):
  - korektna emulacija softvera na različitim ISA arhitekturama, u paralelnom načinu rada
  - (automatska) paralelizacija sekvencijalnih programa.
- Njegovi rezultati pokazuju da je primjena transakcija ponekad bolja od primjene fences. Ako je veličina transakcije dovoljna da amortizira dodatne troškove kod transakcije (koji su relativno fiksni), i ako je učestalost konflikta kod dretvi relativno mala, tada su transakcije bolje od fences.
- **Autor sugerira primjenu hibridne tehnike, koja primjenjuje fences ili transakcije, ovisno od karakteristika aplikacije.**

## Poboljšanja specifikacije ARM arhitekture

- R. Natarajan nije prvi koji je (2015. u [1]) analizirao probleme vezane za memorijski model relaksirane konzistencije.
- Jedan od značajnih radova je iz 2012. ([2]), pod naslovom: "A tutorial introduction to the ARM and POWER relaxed memory models".
- U tom radu, koji ima 50-ak stranica, tri autora sa sveučilišta (INRIA i University of Cambridge UK) analiziraju ponašanje prvenstveno ARM i POWER arhitektura, koje imaju više relaksirane memorijske modele u odnosu na Intel i SPARC arhitekture.
- Između ostalog, autori naglašavaju da bi **specifikacija** ponašanja kod memorijskog modela relaksirane konzistencije **u većini ISA arhitektura trebala biti puno preciznija, zasnovana na matematičkim modelima.**

## Poboljšanja specifikacije ARM arhitekture

- Autori rada [2] su nastavili raditi na tome.
- Jedan od značajnih radova je iz 2016. ([3]), pod naslovom: "Modelling the ARMv8 architecture, operationally: concurrency and ISA".
- Taj rad radila su prethodna tri autora, zajedno s još četiri autora sa sveučilišta i jednim arhitektom iz firme Arm. Iz samog naslova se vidi da je ovdje fokus bio specifično na ARMv8 arhitekturi.
- Vrlo pojednostavljeno (i neprecizno), autori su dali dva semantička modela koji služe za preciznije specificiranje ponašanja ARMv8 mikroarhitekture u konkurentnom radu:
  - **Flowing Model**, koji je bliži konkretnoj mikroarhitekturi
  - apstraktniji **POP (partial-order propagation) Model**
  - te matematički dokaz da POP model dobro apstrahira FM.

## Poboljšanja specifikacije ARM arhitekture

- Poboljšanju specifikacije ARM arhitekture naročito je doprinijelo istraživanje opisano u radu [4] iz 2018.: "Simplifying ARM concurrency: multicopy-atomic axiomatic and operational models for ARMv8", koje je radilo šestero autora, od kojih petoro onih koji su radili [3].
- Kako je rečeno, jedan od autora rada [3], ali i [4], bio je i arhitekt iz firme Arm. U firmi Arm on je, između ostalog, koautor ARMv8 memorijskog modela.
- **Taj Arm arhitekt je u prezentaciji [5] iz 2018., "Formalising the Armv8 memory consistency model", široj publici ("informatičkoj", a ne "matematičkoj"), prikazao napore i uspjehe na poboljšanju specifikacije modela konzistencije memorije kod ARMv8 arhitekture.**
- Između ostalog naglasio je da su znanstvena istraživanja omogućila kvalitetnije specificiranje ARMv8 arhitekture.

# Formalising the Armv8 memory consistency model

- Jednostavan, ali zanimljiv, je 8. slajd iz njegove prezentacije, koji prikazujemo ovdje.

## Ok, so what do I need to know?

Architects, CPU vendors and programming languages have helpfully documented their memory models, so we just need to read their specifications...

C++ good intentions and well written, but flawed (thin-air, unsound wrt h/w)

x86 TSO, except where it isn't (IRIW)

Armv7/PPC Mind-bending recursion attempts to place accesses into 'groups'

JMM defined empirically in terms of a cryptic set of tests

Perl6(!) can't tell if it's a joke. I hope that it is.

No formal wording. No common nomenclature. No common abstraction. No official tooling. No accountability.

It mostly works by magic...

...Engineering shouldn't be magic.

# Hardverska transakcijska memorija (HTM) i ARM arhitektura

- U članku <https://en.wikipedia.org/wiki/AArch64> pri kraju piše:  
***In May 2019, ARM announced their upcoming Scalable Vector Extension 2 (SVE2) and Transactional Memory Extension (TME).***
- Podsjetimo se da je prvi visokotržišni procesor s HTM-om bio Intelov Haswell (2013.), kroz tehnologiju **Transactional Synchronization Extensions (TSX)**.
- Nažalost, Haswell i nasljednik Broadwell imali su bugove u TSX-u. **Niti danas situacija nije sjajna, kako se vidi na:**  
[https://en.wikipedia.org/wiki/Transactional\\_Synchronization\\_Extensions](https://en.wikipedia.org/wiki/Transactional_Synchronization_Extensions)  
*However, Intel 10th generation Comet Lake and Ice Lake CPUs, which were released in 2020, **do not support TSX/TSX-NI ... including both HLE and RTM.***



## Hardverska transakcijska memorija (HTM) i ARM arhitektura

- Međutim, možda ipak ima nade s Intelovim procesorima i HTM-om, jer pri kraju navedenog članka piše:  
*In Intel Architecture Instruction Set Extensions Programming Reference revision 41 from October 2020,[30] a new TSXLDTRK instruction set extension was documented and slated for inclusion in the upcoming Sapphire Rapids processors.*
- Nešto slično se možda desilo i s IBM PowerPC procesorima. Na [https://en.wikipedia.org/wiki/Transactional\\_memory](https://en.wikipedia.org/wiki/Transactional_memory) piše:  
*Available implementations ... IBM POWER8 and 9, removed in POWER10 (Power ISA v.3.1)*
- **Nadamo se da će na temelju ovih iskustava firma Arm pažljivo i uspješno implementirati HTM u ARM arhitekturi.**

## Kako je Apple uspio da programi pisani za Intel dobro rade na njegovom M1 čipu ARM arhitekture?

- Vjerojatno je točan odgovor koji se može naći (i) u blogu na: <https://www.infoq.com/news/2020/11/rosetta-2-translation/>  
*... While byte ordering is not a problem for the transition from x86 to ARM, another issue related to memory, namely the memory consistency model total store ordering (TSO), could hamper performance in this case. **To prevent this from happening, Apple added support for x86 memory ordering to the M1 CPU, as Robert Graham noted on Twitter.***
- A ovo je izvorno rekao Robert Graham (Nov 26, 2020):  
*So Apple simply cheated. They added Intel's memory-ordering to their CPU. **When running translated x86 code, they switch the mode of the CPU to conform to Intel's memory ordering.***

## Zaključak

- Procesori ARM ISA arhitekture ne koriste se više "samo" za mobitele i tablete, nego i za serverska računala (pa i superračunala), te laptop/desktop računala (npr. Macintosh).
- Želja je da se programi pisani za jednu ISA arhitekturu mogu sa što manje napora izvršavati na računalima koja imaju drugačiju ISA arhitekturu.
- Problem je u tome što različite ISA arhitekture mogu imati dosta različite memorijske modele. Naročito je značajan različit stupanj relaksiranja konzistencije memorije.
- Lakše je napraviti da program pisan za više relaksiranu arhitekturu (npr. ARM ili POWER) radi na manje relaksiranoj arhitekturi (npr. x86 ili SPARC). Obrnuto je teže.
- ISA arhitekture trebaju biti preciznije definirane.
- Kako kaže ARM arhitekt u prezentaciji [5]:  
It mostly works by magic ... Engineering shouldn't be magic.

## Literatura (dio)

1. Natarajan, R. (2015): *Leveraging hardware support for transactional execution to address correctness and performance challenges in software*, doktorska disertacija, University of Minnesota
2. Maranget, L., Sarkar, S., Sewell, P. (2012): *A tutorial introduction to the ARM and POWER relaxed memory models*, autori iz: University of Cambridge UK, INRIA
3. Flur, S., Gray, K., Pulte, C., Sarkar, S., Sezgin, A., Maranget, L., Deacon W., Sewell P. (2016): *Modelling the ARMv8 architecture, operationally: concurrency and ISA*, autori iz: University of Cambridge UK, University of St Andrews UK, INRIA, ARM Ltd.
4. Pulte, C., Flur, S., Deacon W., French, J., Sarkar, S., Sewell P. (2018): *Simplifying ARM concurrency: multicopy-atomic axiomatic and operational models for ARMv8*, autori iz: University of Cambridge UK, University of St Andrews UK, ARM Ltd.
5. Deacon, W. (2018): *Formalising the Armv8 memory consistency model*, prezentacija, OpenSHMEM workshop, Baltimore MD