

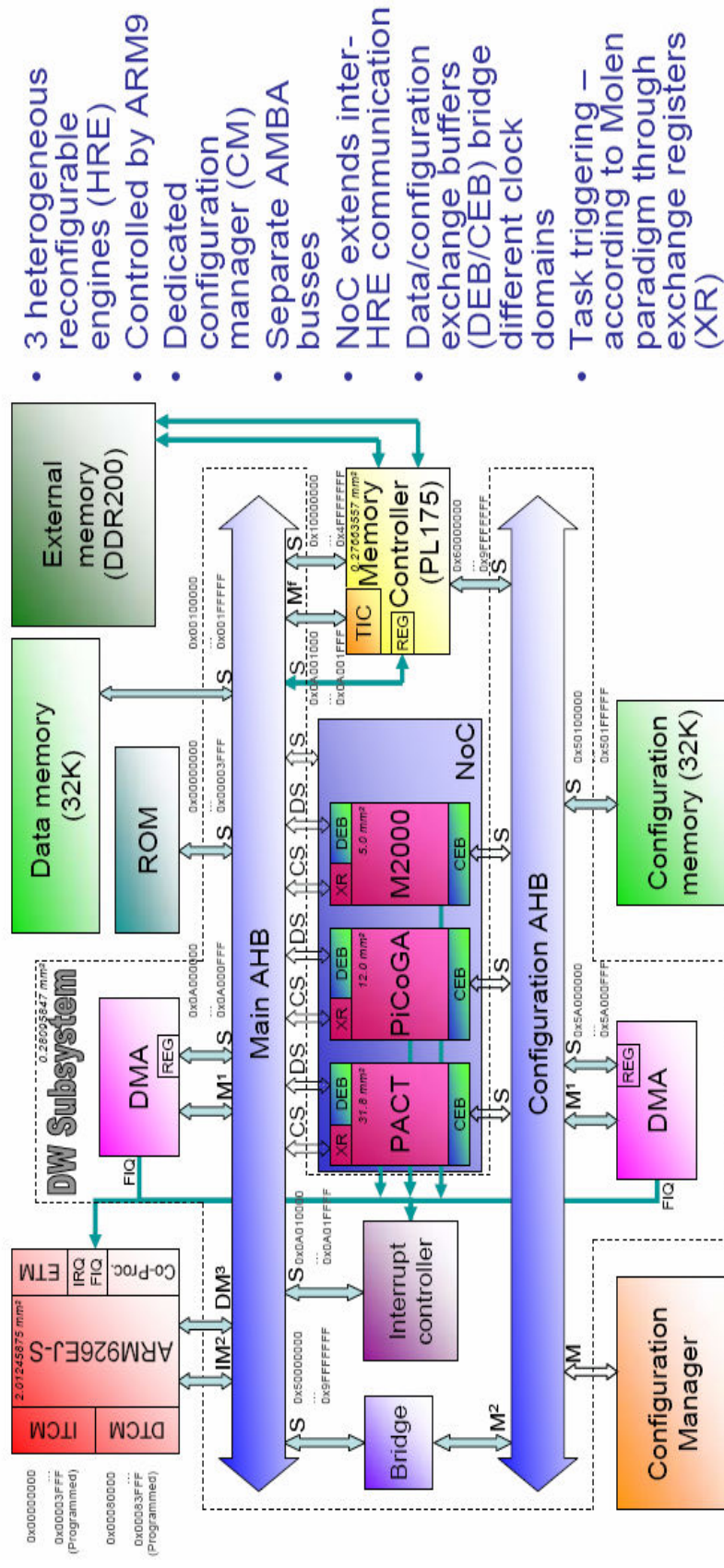
Áramkörön belüli rendszerek

SoC

System on a Chip

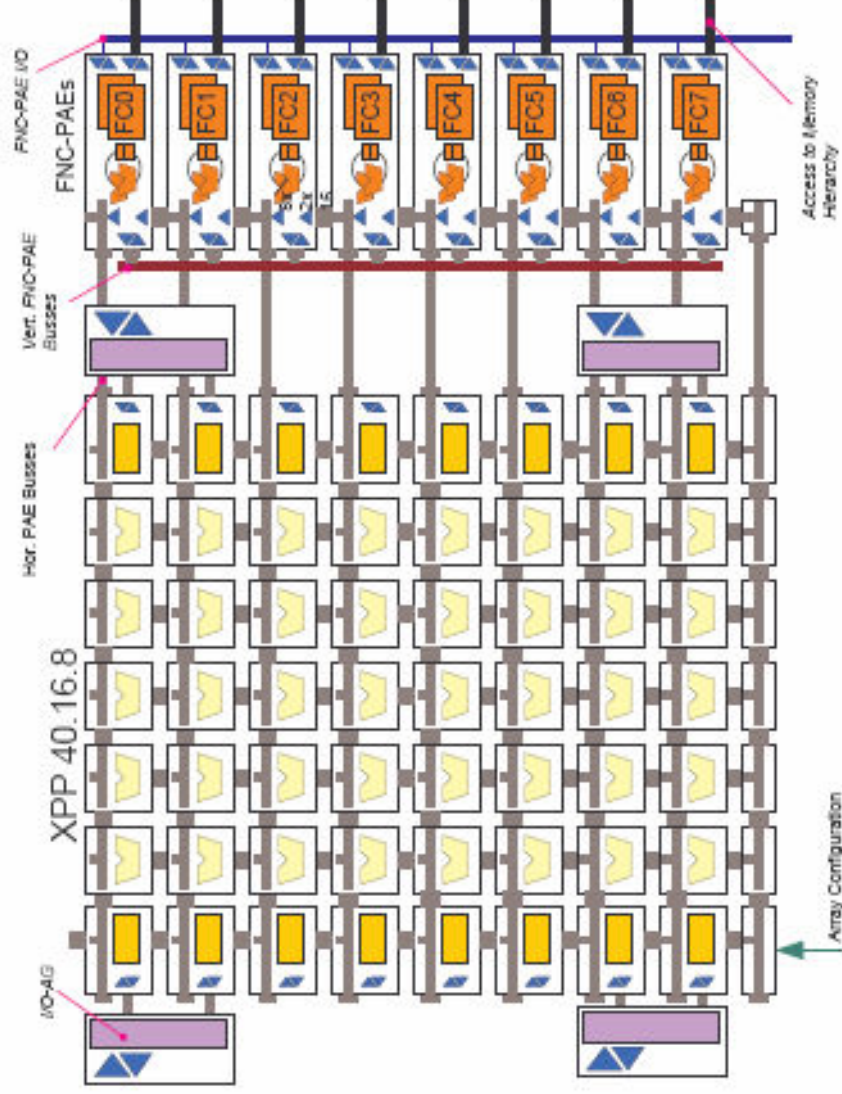
Egy mintapélda

MORPHEUS Architecture

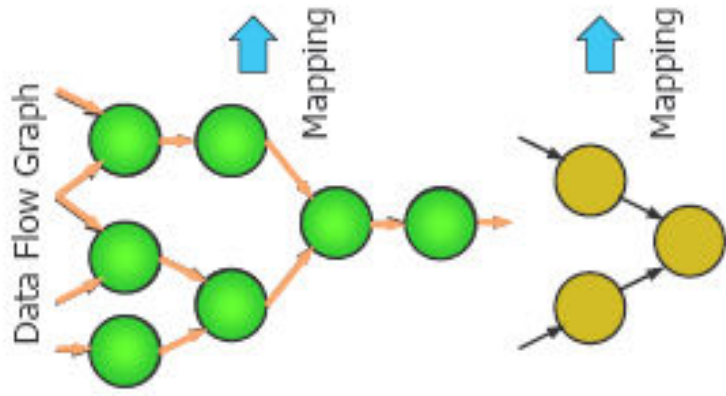
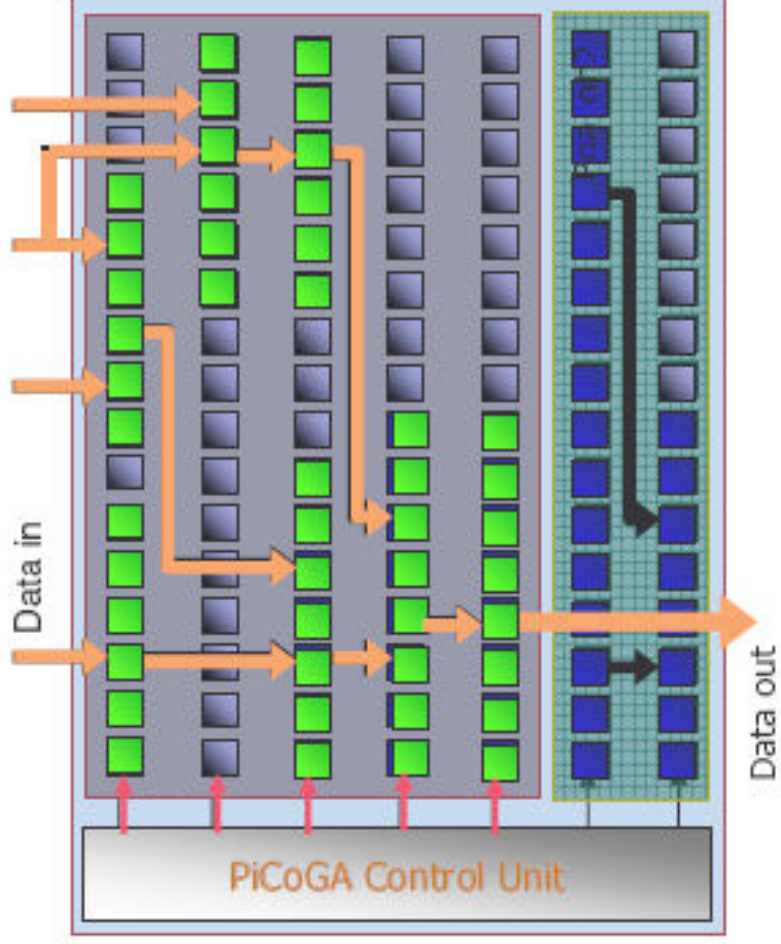


- 3 heterogeneous reconfigurable engines (HRE)
- Controlled by ARM9
- Dedicated configuration manager (CM)
- Separate AMBA busses
- NoC extends inter-HRE communication
- Data/configuration exchange buffers (DEB/CEB) bridge different clock domains
- Task triggering – according to Molen paradigm through exchange registers (XR)

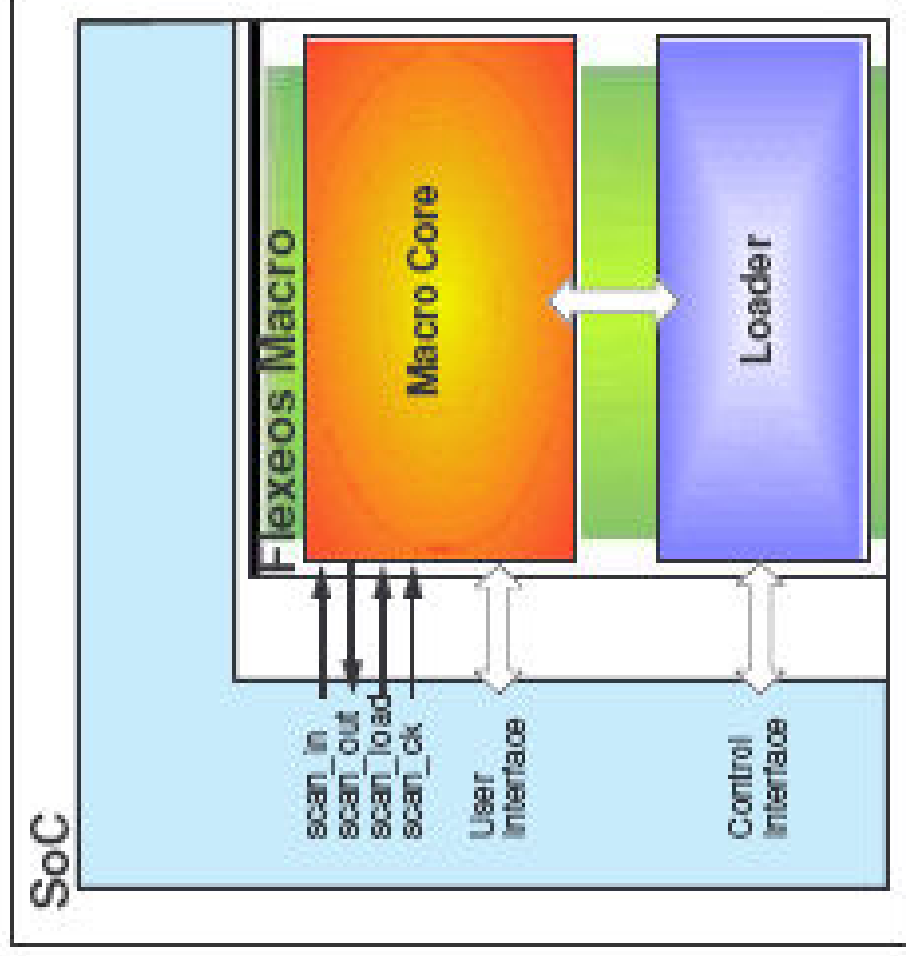
PACT XPP blokk



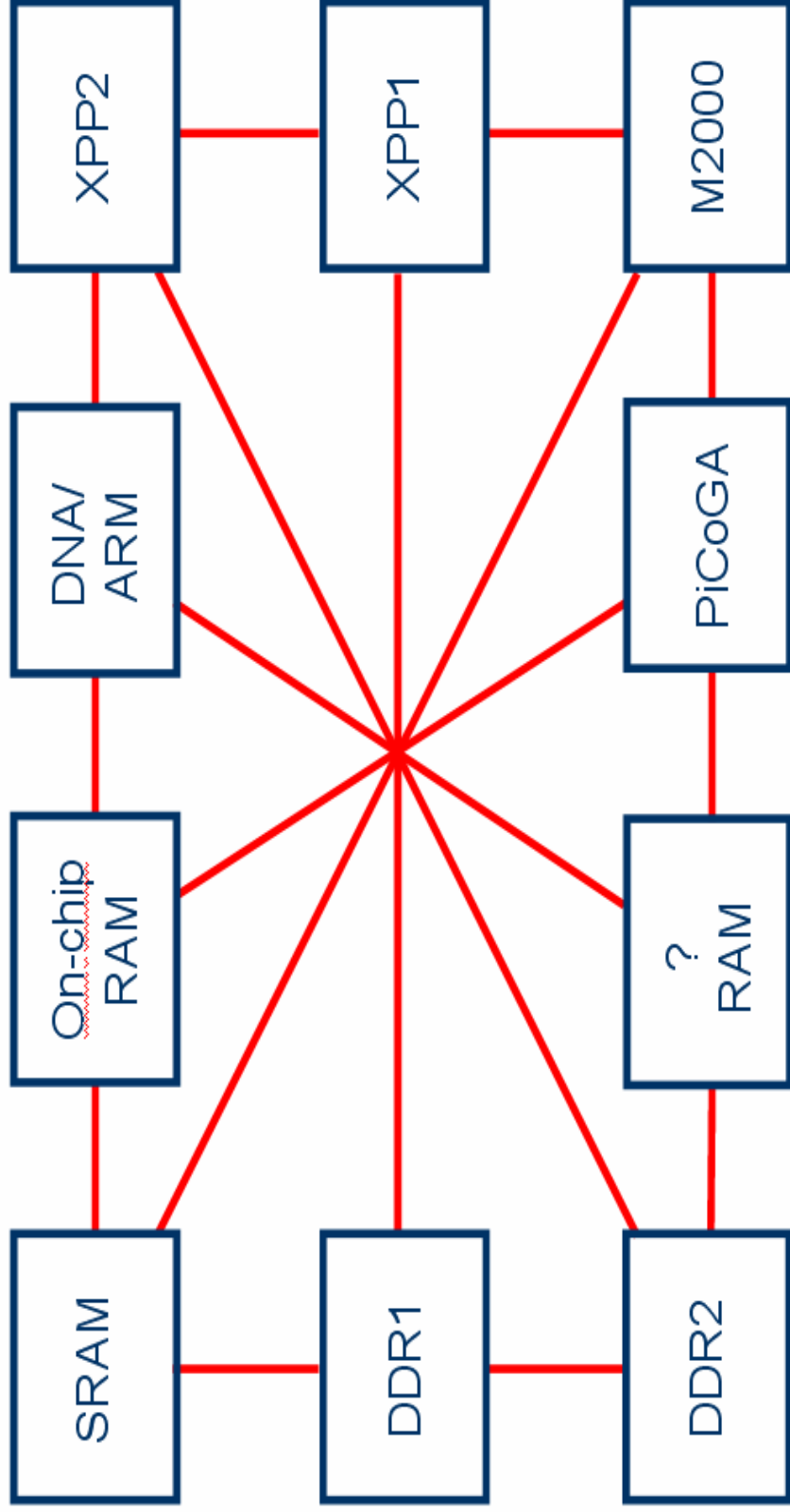
PiCoGa



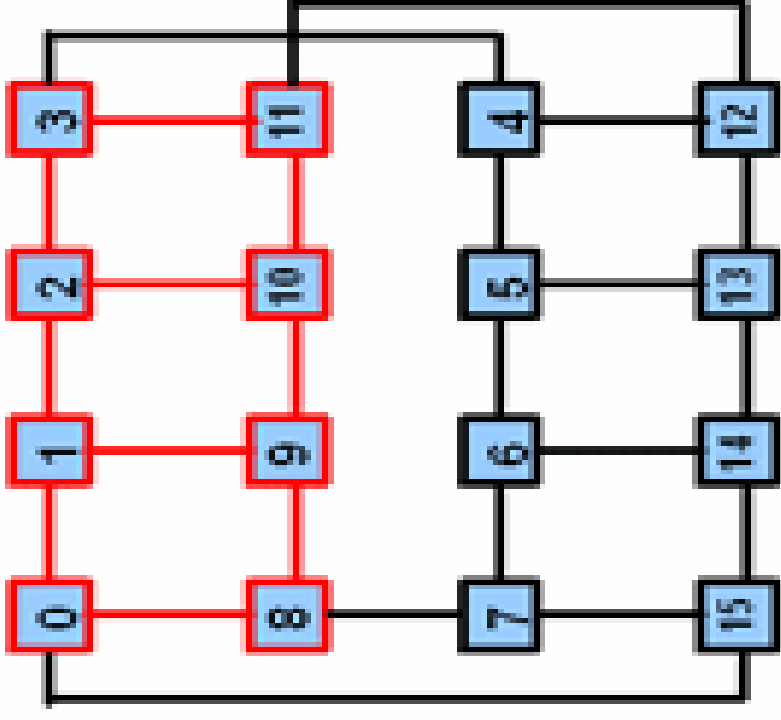
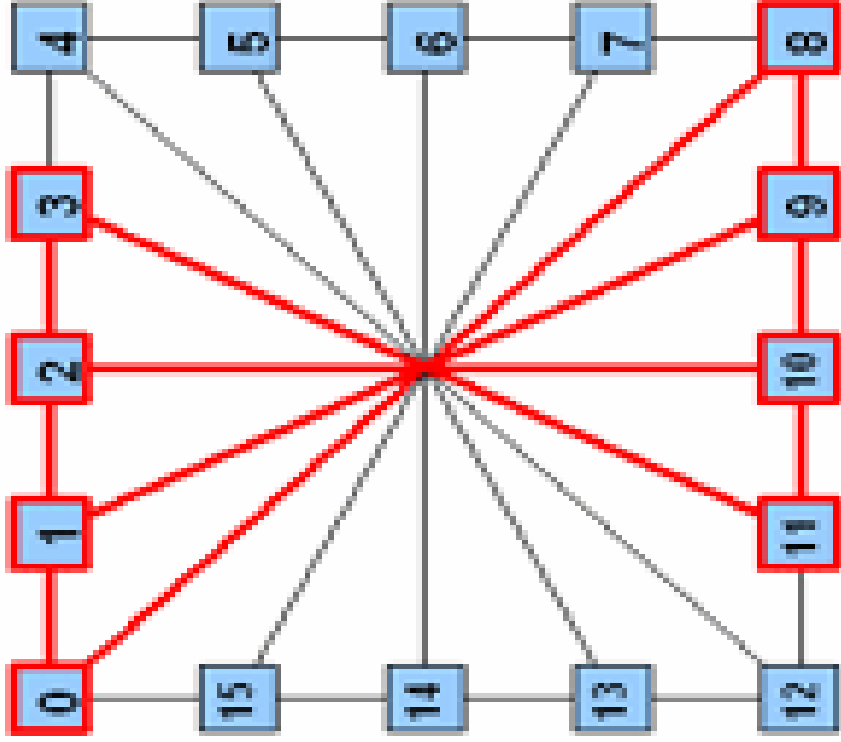
M200 Embedded FPGA



Az NoC részlet



Az STNoC előnye



Áramkörön belüli rendszerek

SoC / SoPC

System on a (Programmable)
Chip

Alapfogalmak

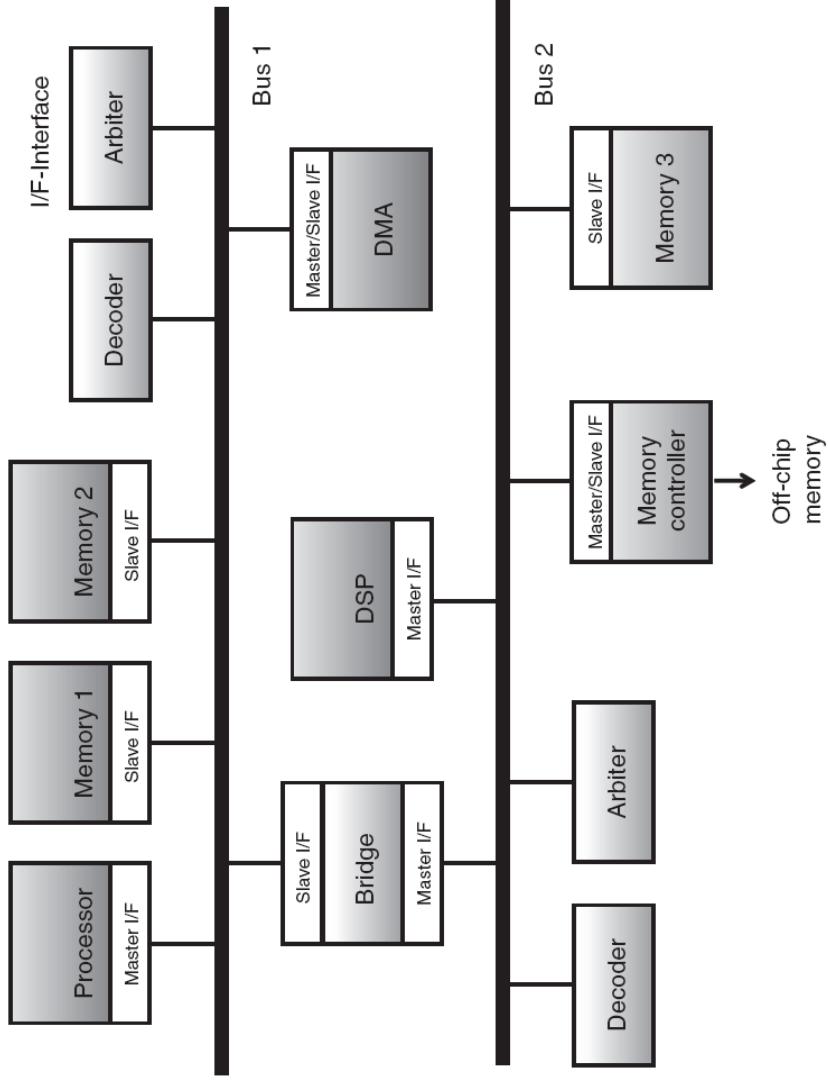


FIGURE 2.1

Example of an SoC with a bus-based communication architecture

Fizikai struktúra 1.

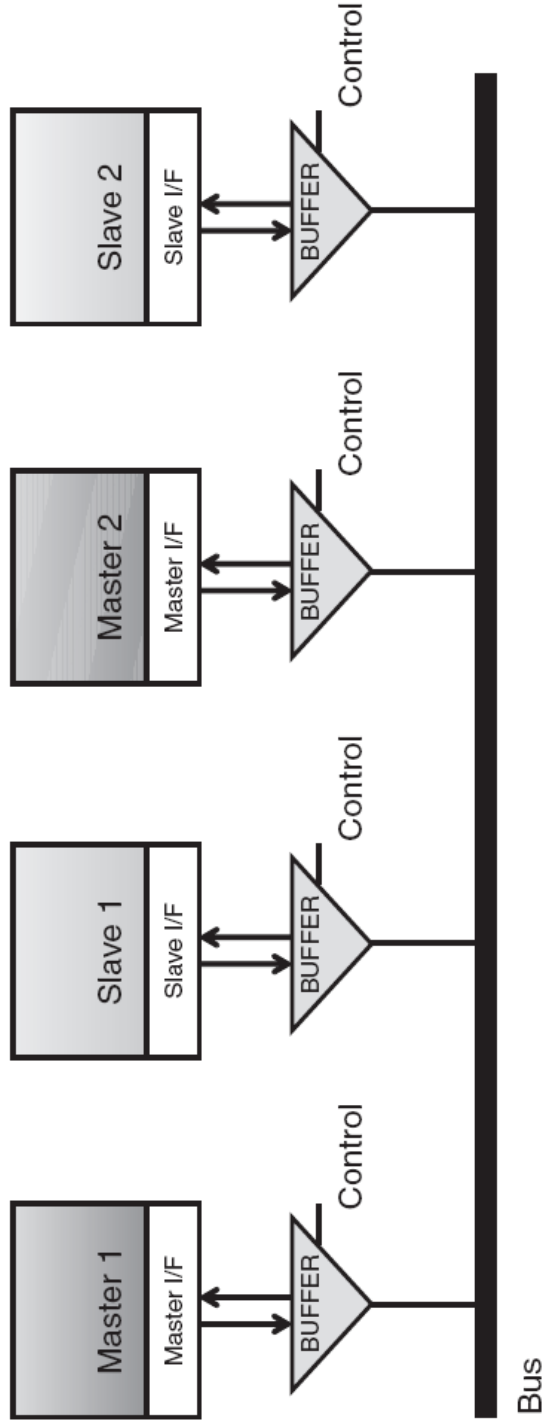


FIGURE 2.3(a)

Shared bus implementation alternatives: tri-state buffer based bidirectional signals

Fizikai struktúra 2.

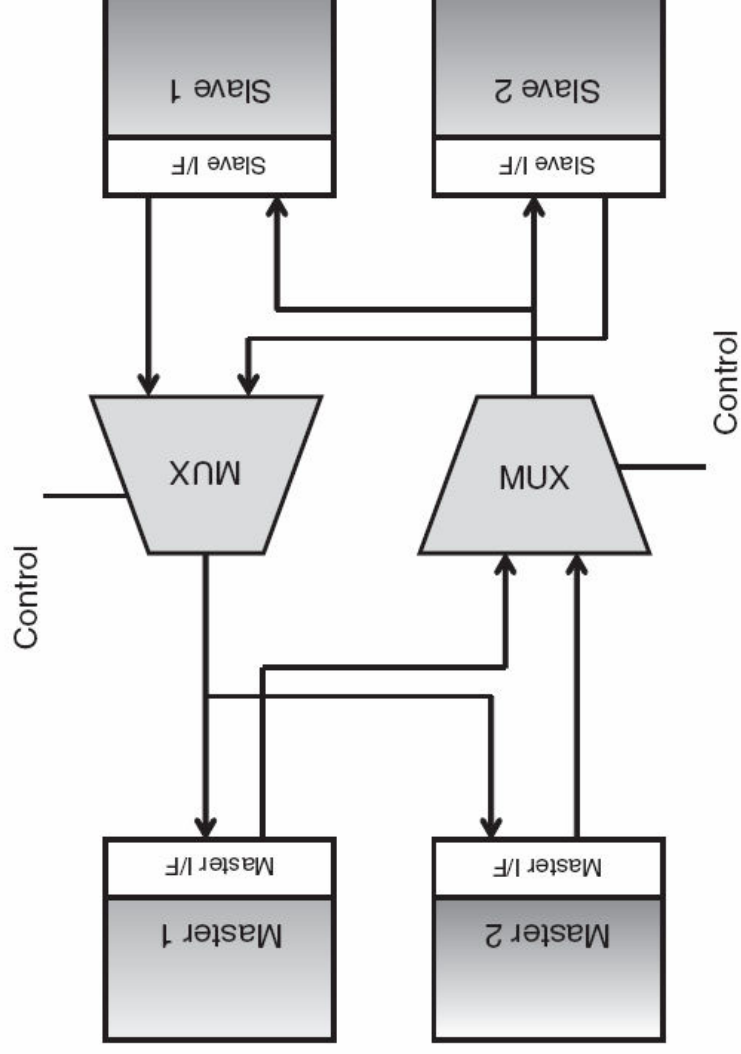


FIGURE 2.3(b)

MUX-based

Fizikai struktúra 3.

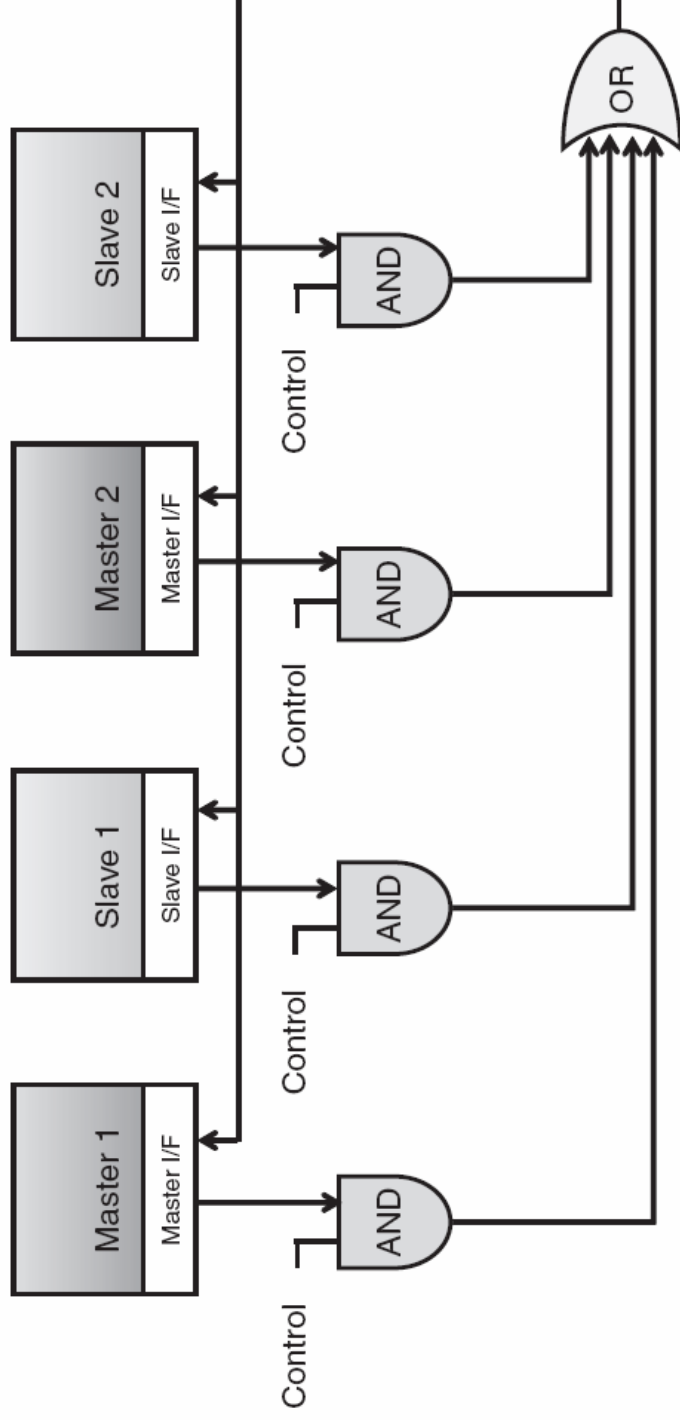


FIGURE 2.3(c)

AND-OR bus

Központi dekódolás

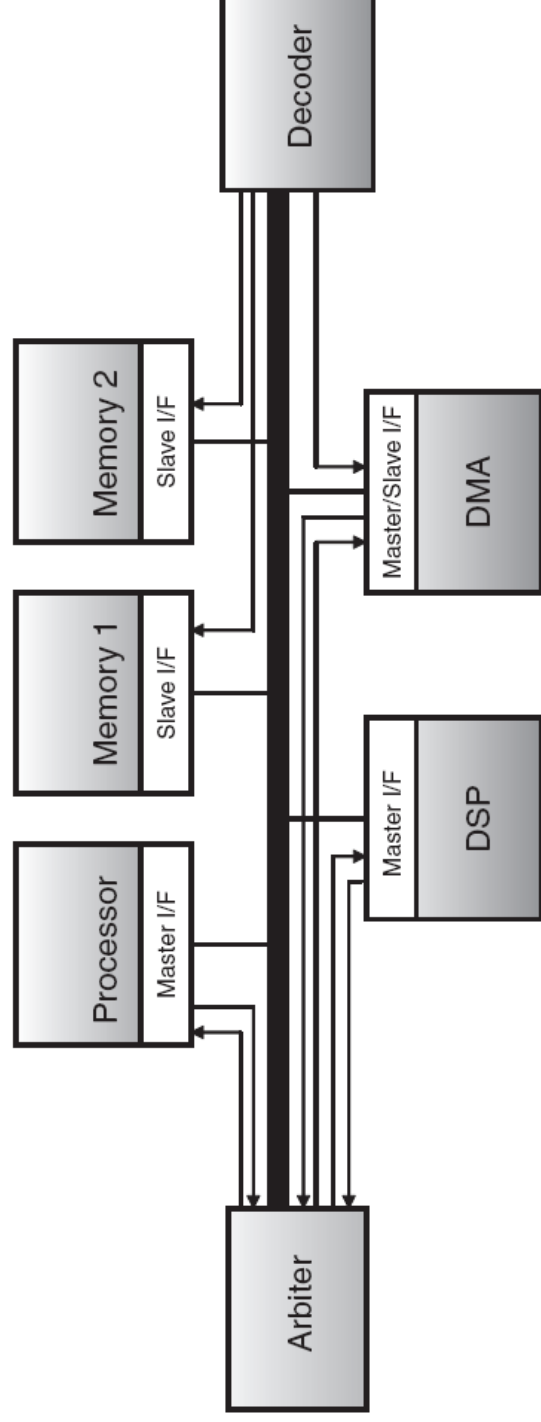


FIGURE 2.5(a)

Different implementation strategies for decoder and arbiter: centralized

Elosztott dekódolás

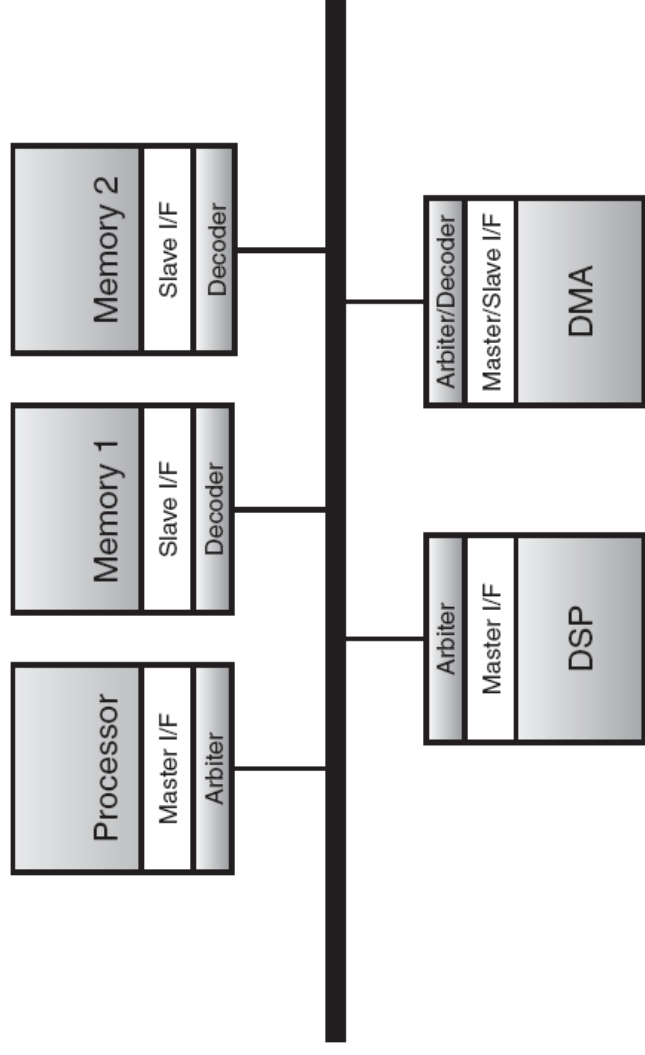


FIGURE 2.5(b)

Distributed

Adatátviteli módok - Egyszerű

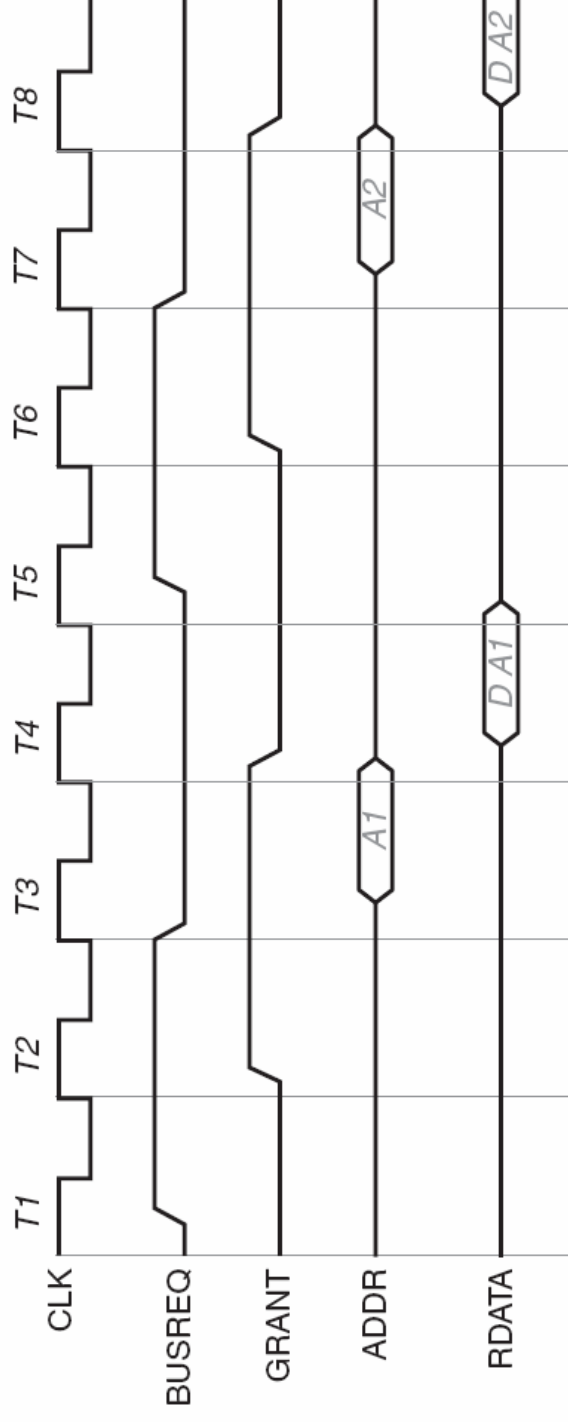


FIGURE 2.6

Single non-pipelined data transfer mode

Adatátviteli módok - Pipeline

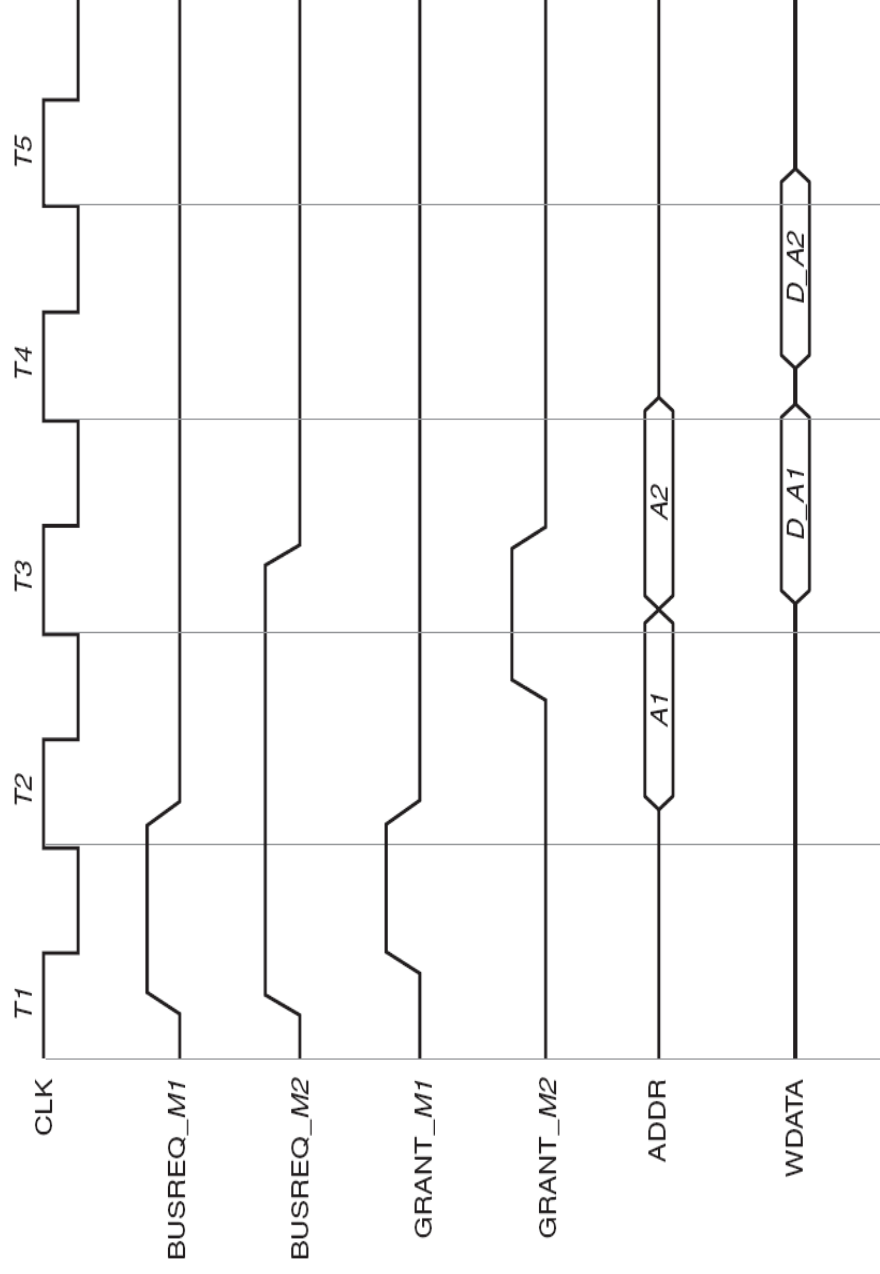


FIGURE 2.7

Pipelined data transfer mode

Adatátviteli módok - Burst

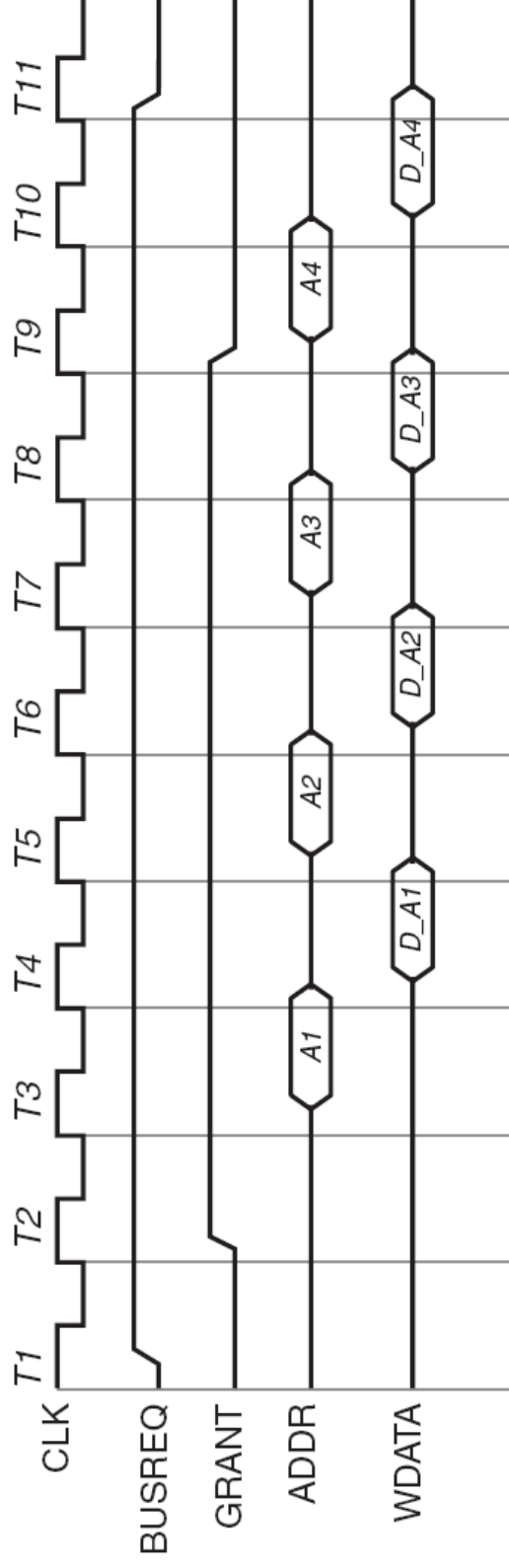


FIGURE 2.8(a)

Example of master writing four data items in burst transfer mode: non-pipelined burst transfer mode

Adatátviteli módok – Pipelined Burst

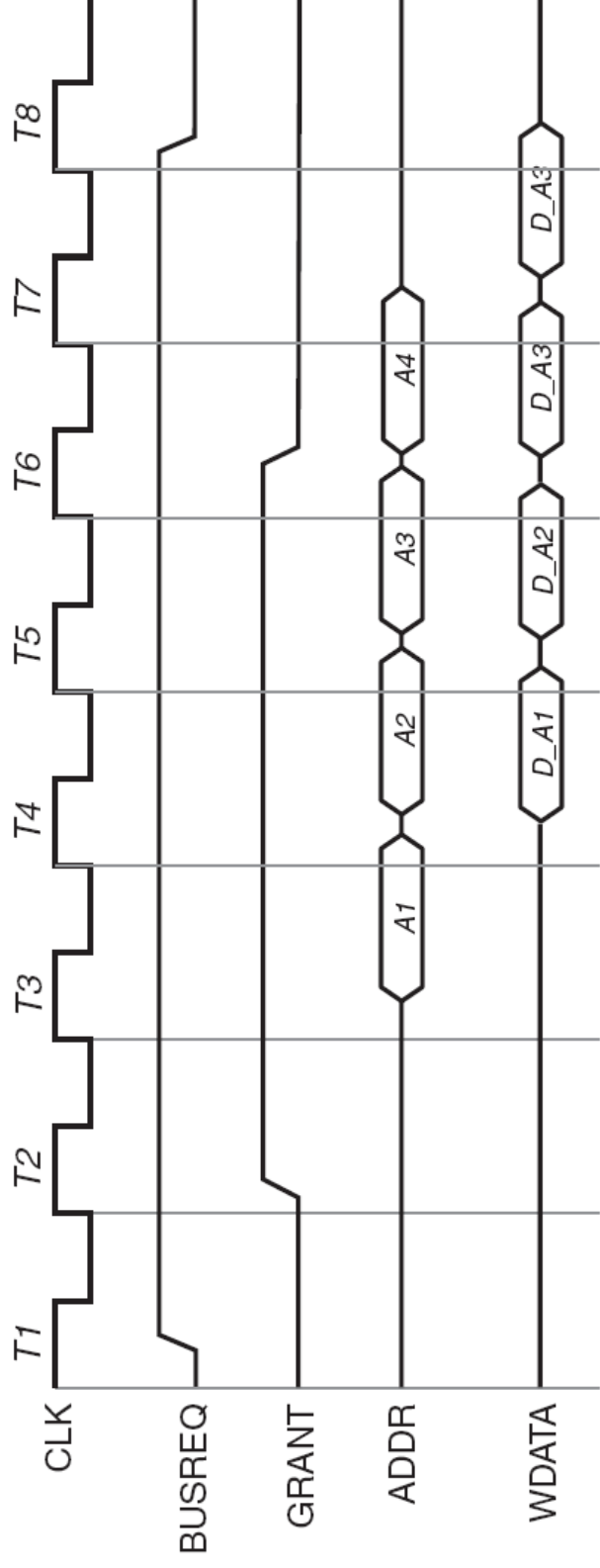


FIGURE 2.8(b)

Pipelined burst transfer mode

Busz topológiák - Egyszerű

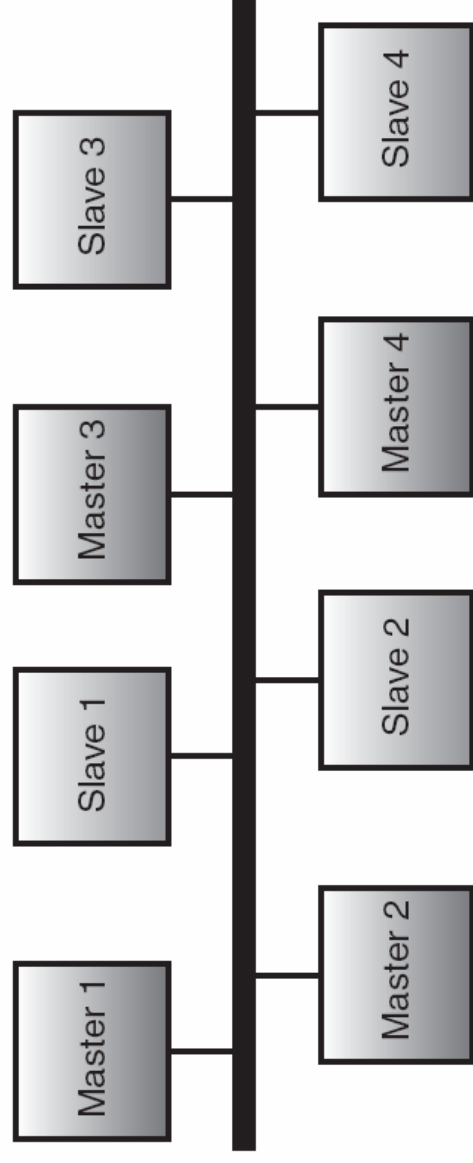


FIGURE 2.9(a)

Different bus-based communication architecture topology structures: single bus

Busz topológiák - Hierarchikus

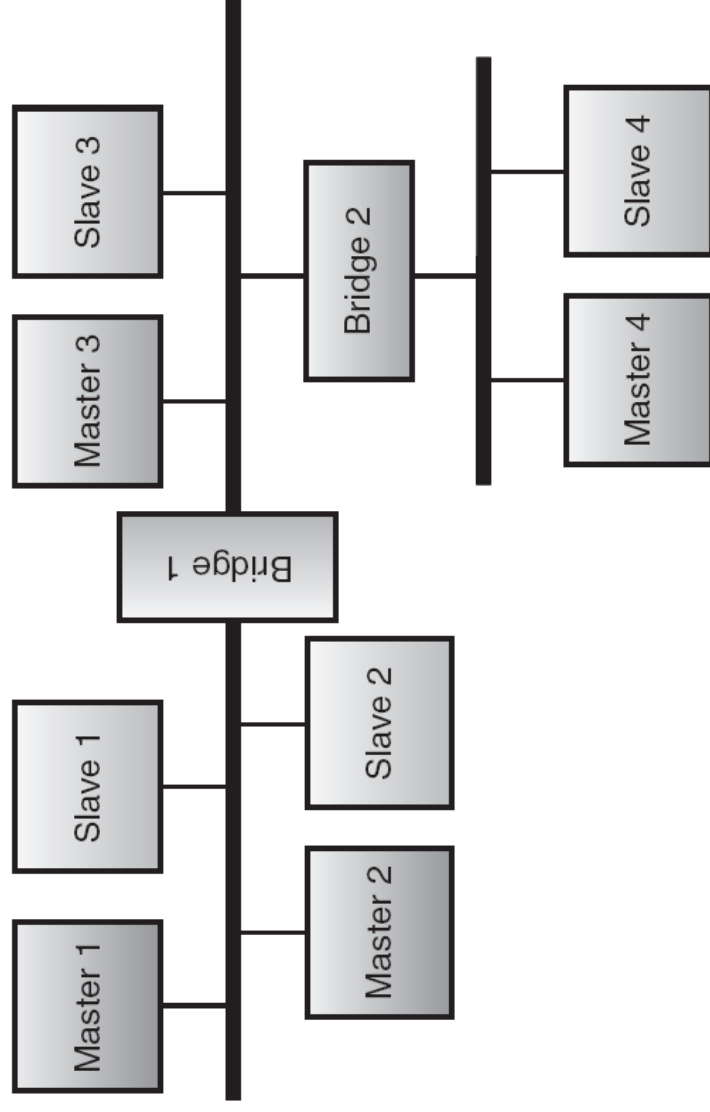


FIGURE 2.9(b)

Hierarchical bus

Busz topológiák - Megosztott

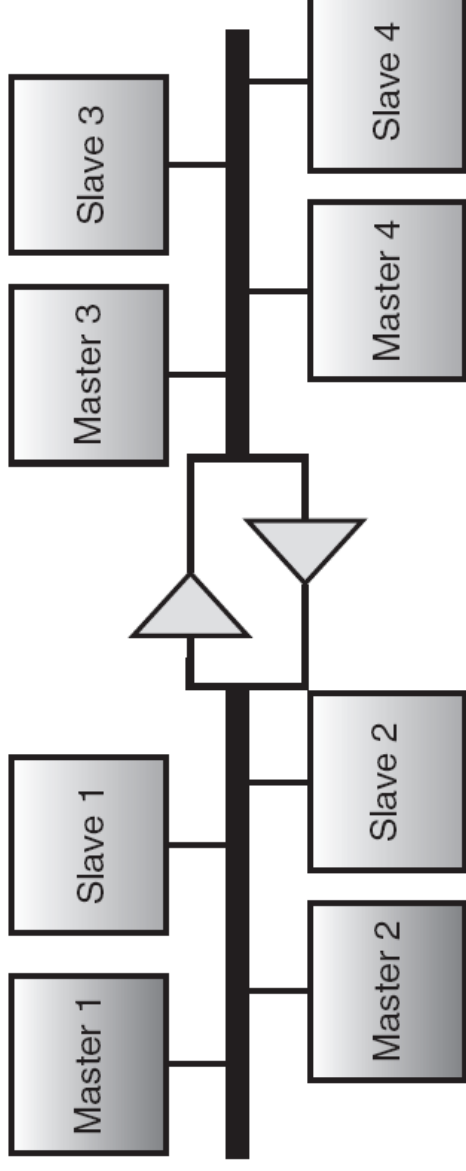


FIGURE 2.9(c)

Split bus

Busz topológiák – Teljes pont-pont

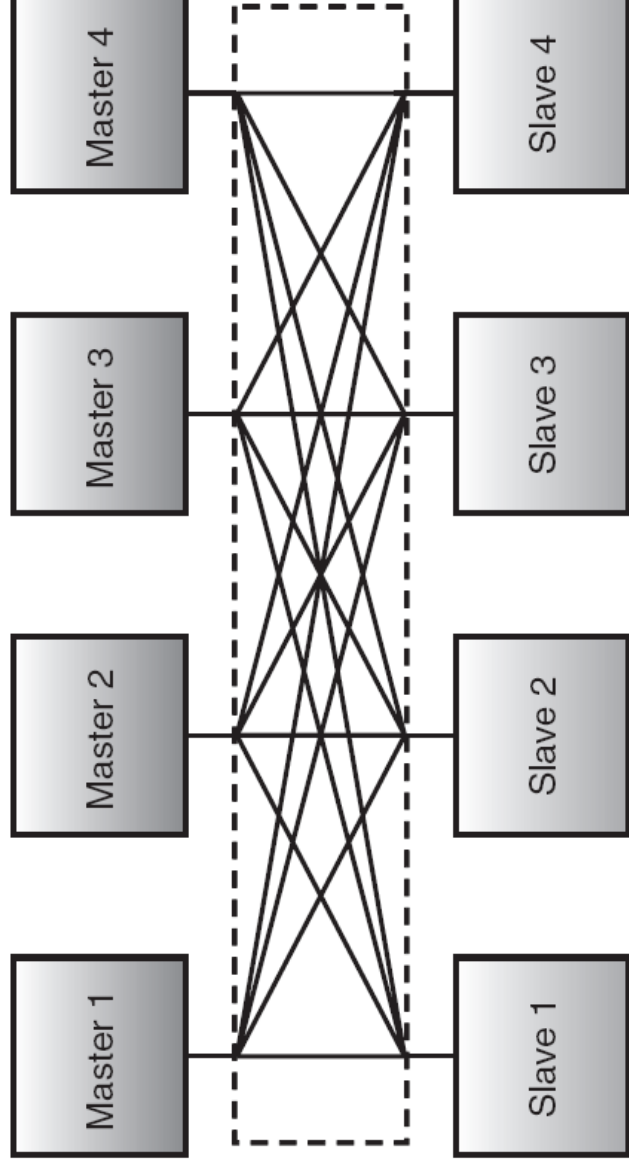


FIGURE 2.9(d)

Full bus crossbar (or point-to-point bus)

Busz topológiák – Részleges

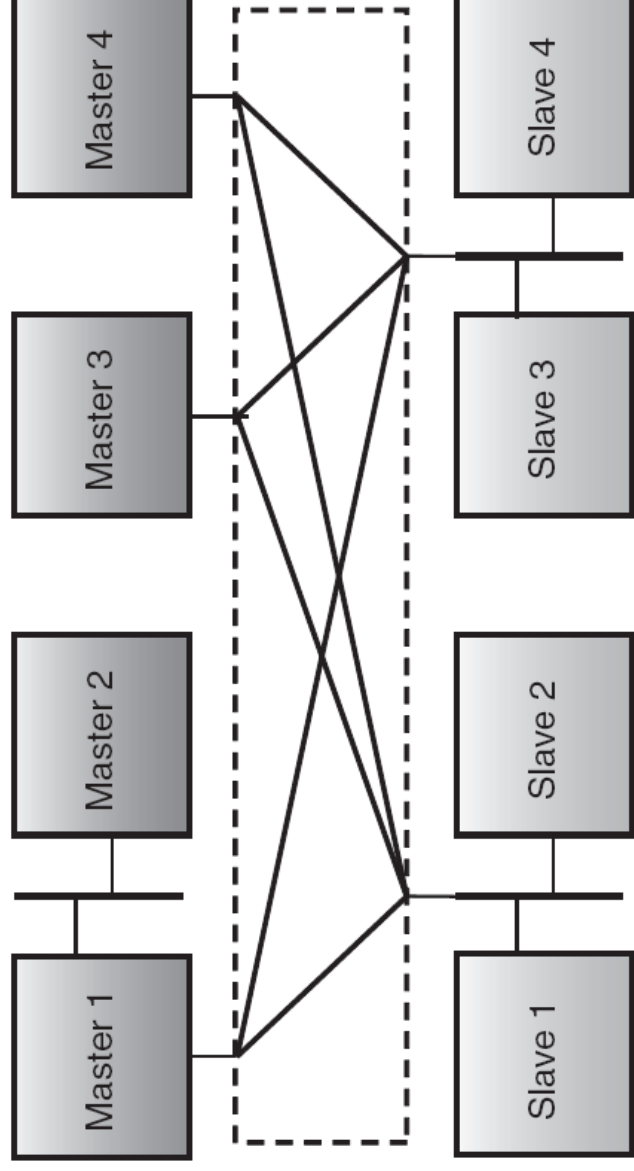


FIGURE 2.9(e)

Partial bus crossbar

Busztopológiák

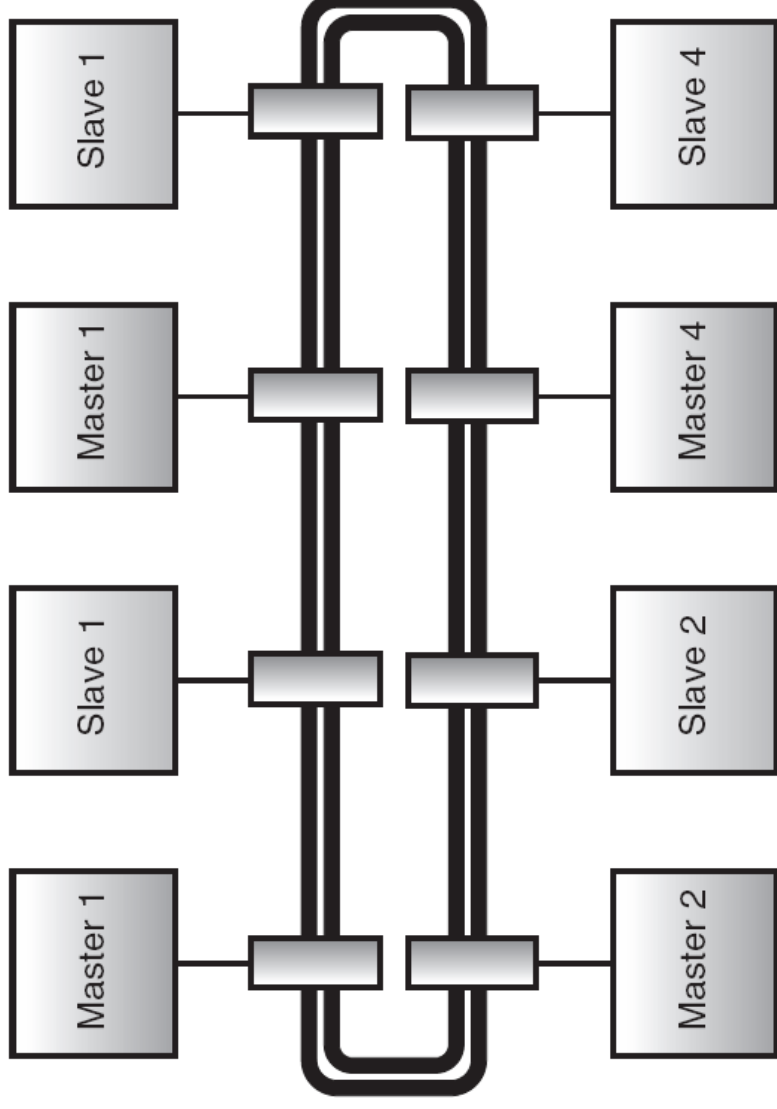


FIGURE 2.9(f)

Ring bus

Áramkörök belüli szabványos buszrendszerek

- ARM AMBA
- IBM CoreConnect
- STM STBus
- Sonic SMART interconnects
- OpenCores Wishbone
- Altera Avalon

ARM AMBA busz rendszere

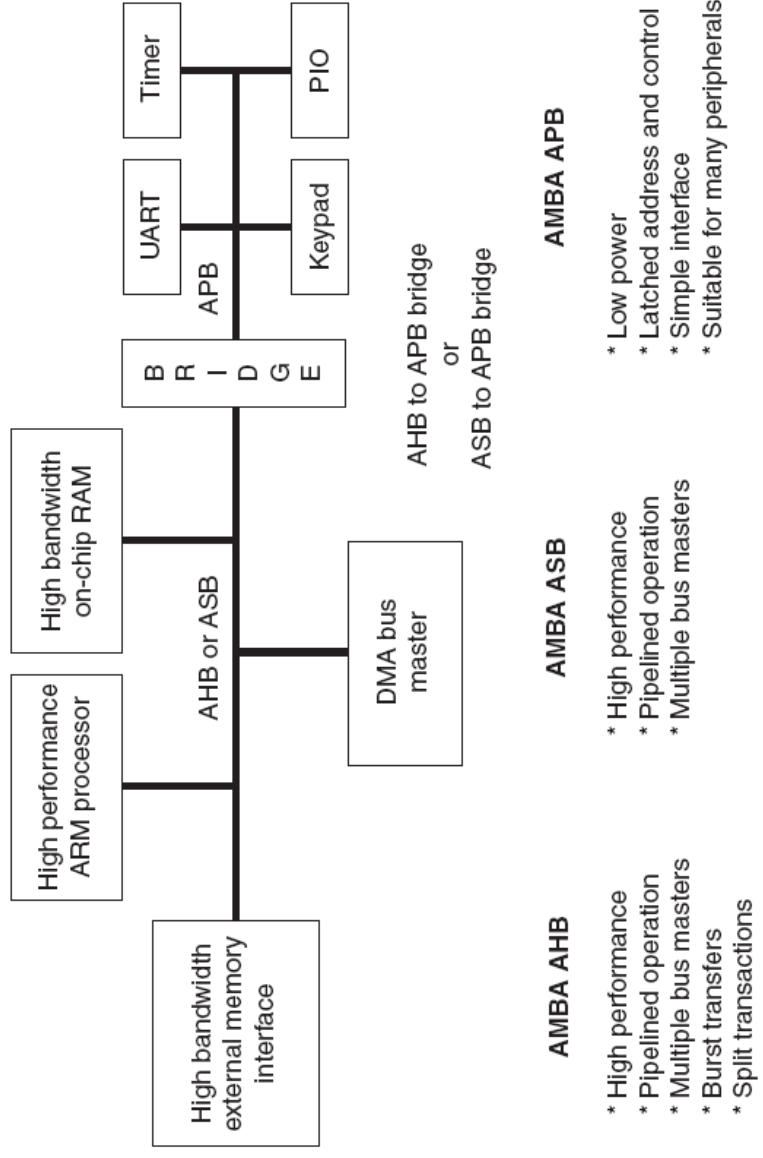


FIGURE 3.1

A typical AMBA 2.0 system [1]

Source: ARM Inc.

AHB topológia

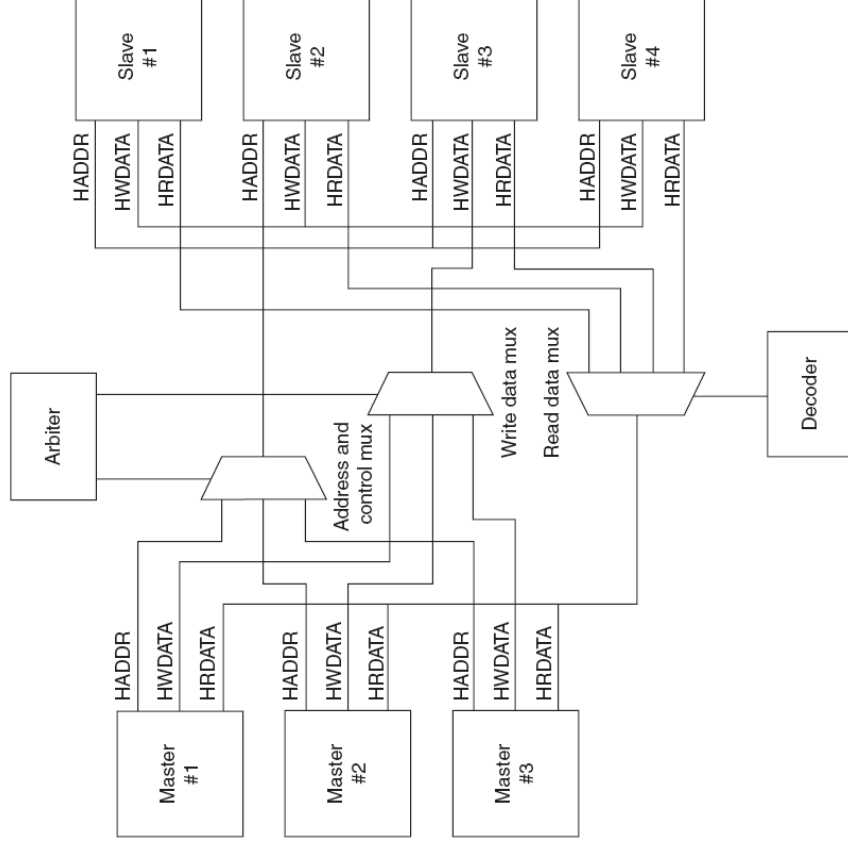


FIGURE 3.2

AHB multiplexer interconnection scheme [1]

Source: ARM Inc.

AHB busz adatátvitel

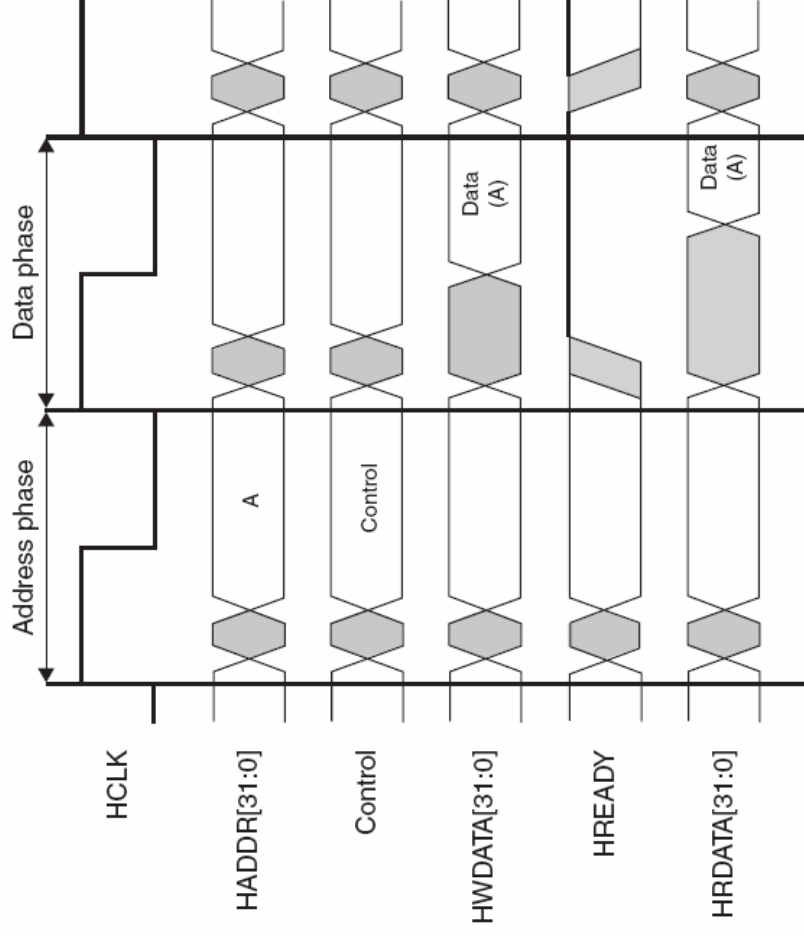


FIGURE 3.3

Basic data transfer on AHB bus [1]

Source: ARM Inc.

AHB busz adatátvitel várakozással

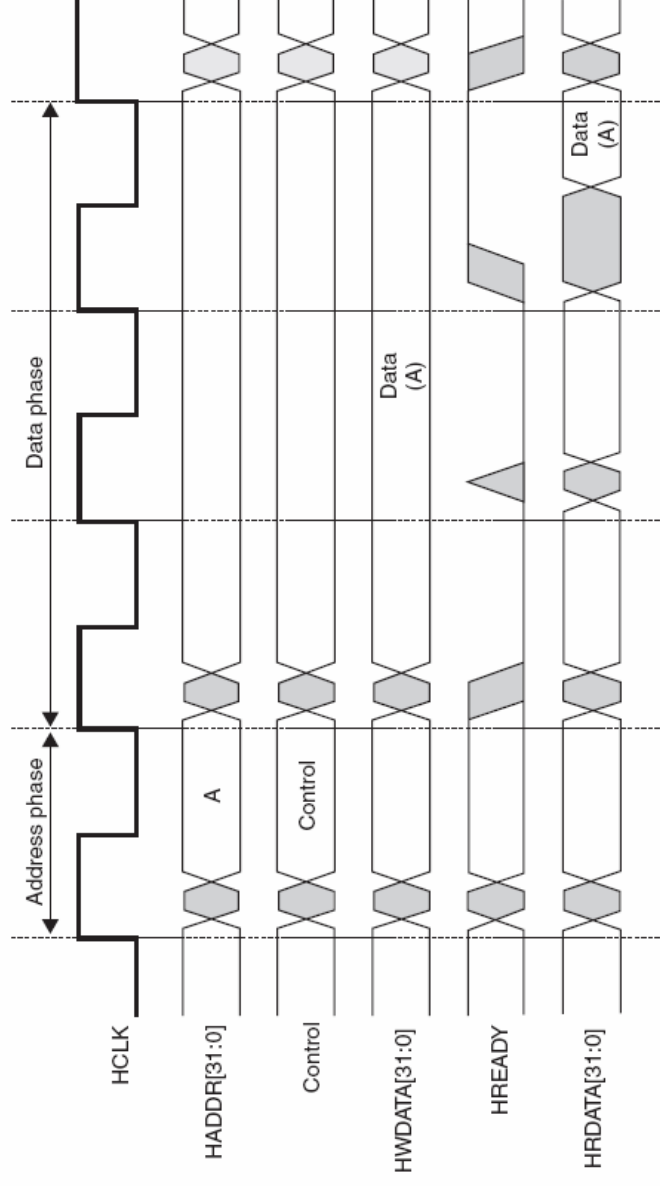


FIGURE 3.4

Basic data transfer on AHB bus with slave wait states [1]
Source: ARM Inc.

AHB bus adatátvitel- pipeline

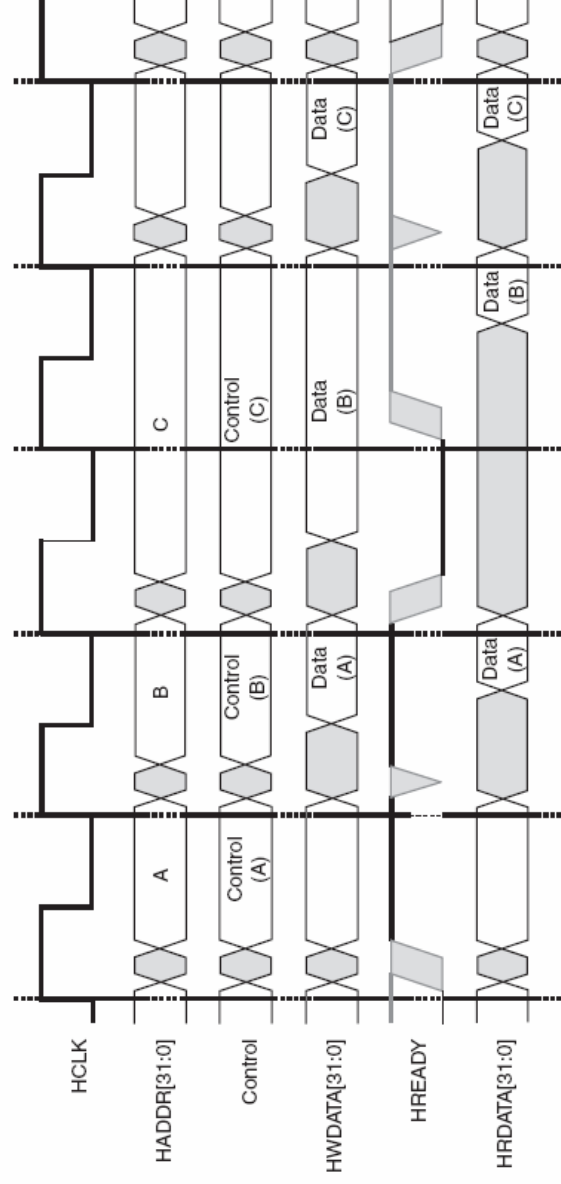


FIGURE 3.5

Pipelined data transfers on AHB bus [1]

Source: ARM Inc.

AHB ciklusok

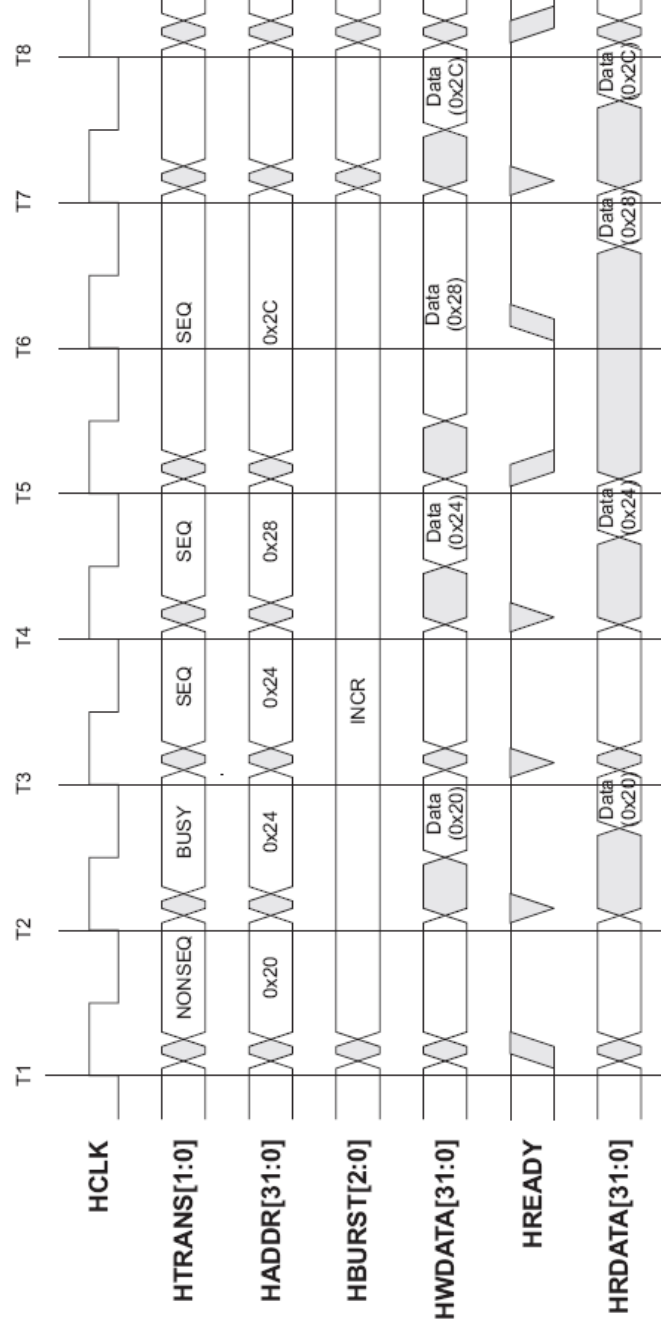


Figure 3-6 Transfer type examples

AHB átviteli típusok

- HTRANS[1:0]
 - IDLE
 - BUSY
 - NONSEQ
 - SEQ
- HBURST[2:0]
 - SINGLE
 - INCR
 - WRAP4
 - INCR4
 - WRAP8
 - INCR8
 - WRAP16
 - INCR16
- 1Kbyte address limit

AHB WRAP8 burst

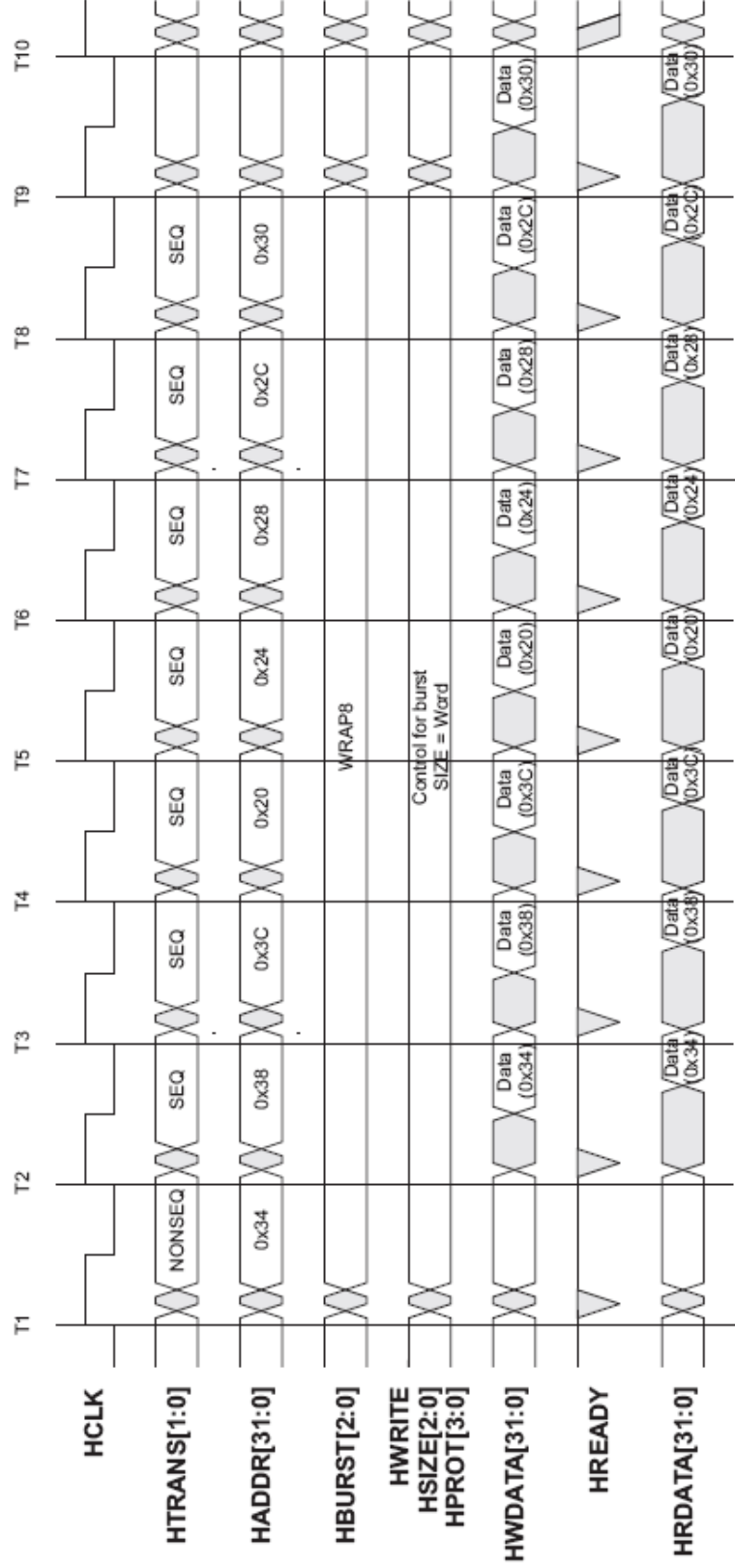


Figure 3-9 Eight-beat wrapping burst

AHB INCR8 burst

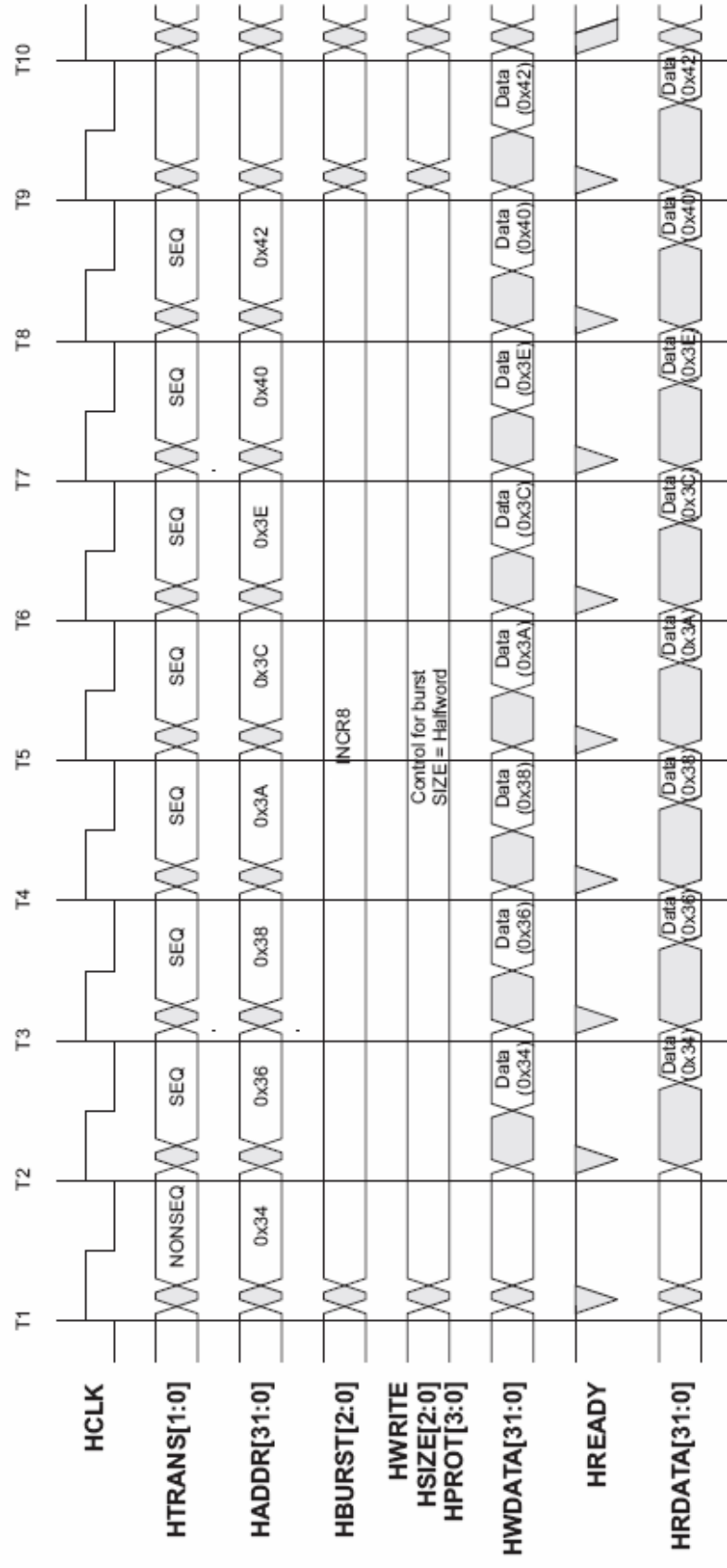


Figure 3-10 Eight-beat incrementing burst

AHB nem-fix burst

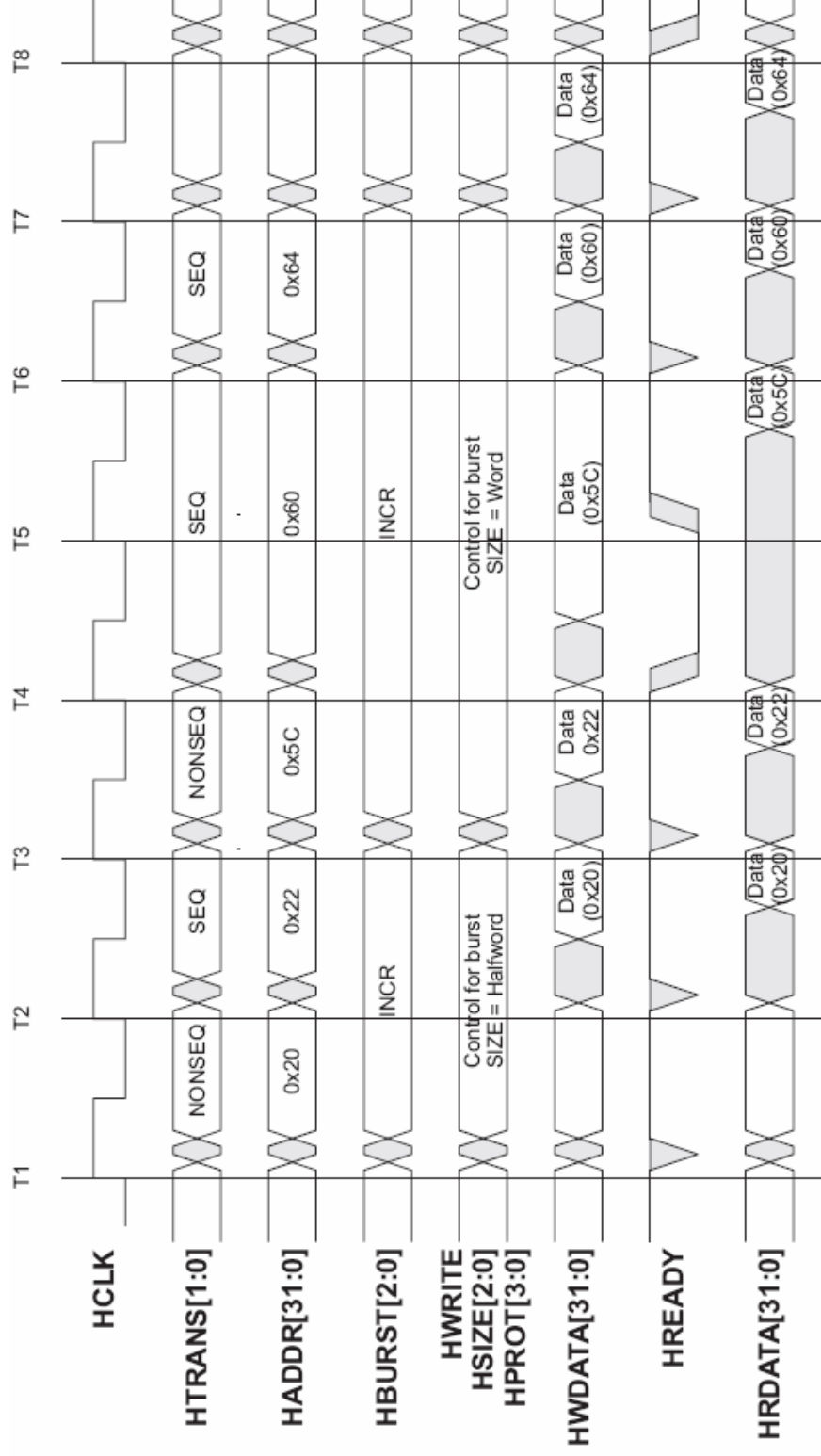


Figure 3-11 Undefined-length bursts

AHB burst szabályok

- A burst mindig egy 1K bájttartományon belül marad! (MASTER felelőssége!)
- Adatmennyiség = Adtaméret * Ütemek száma
- INCR használható SINGLE helyett
- Minden átvitel méret szerinti címhatárra illesztett
- Korai BURST lezárás (NONSEQ vagy IDLE)

AHB arbitració

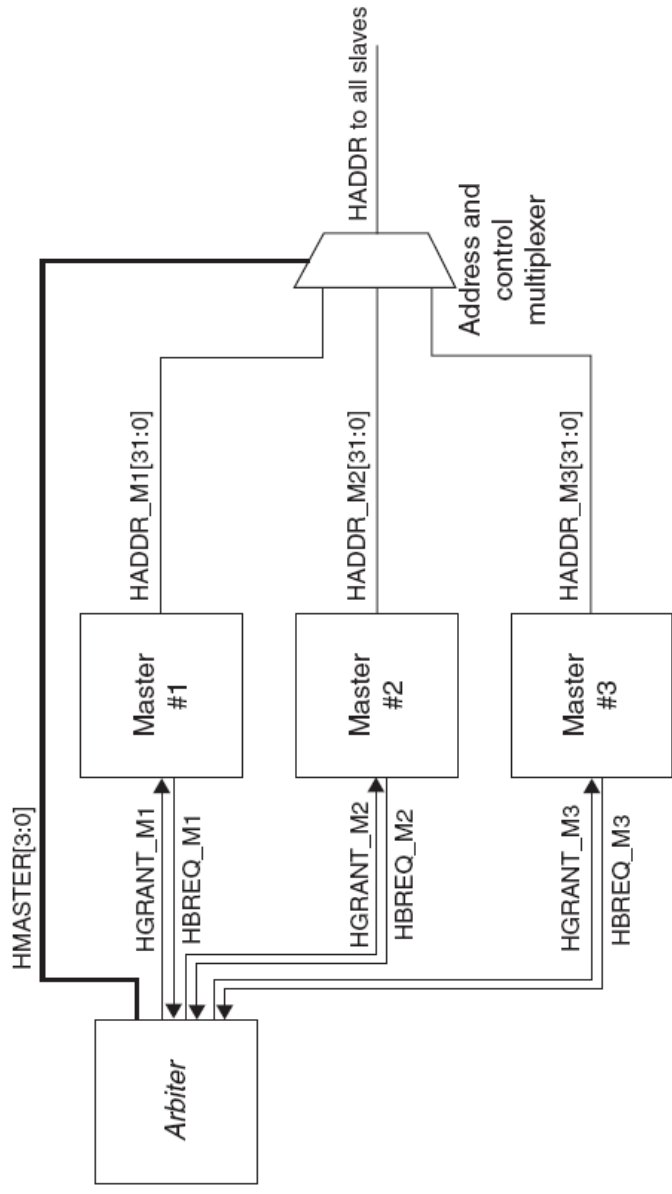


FIGURE 3.6

Arbitration on AHB bus [1]

Source: ARM Inc.

AHB arbitráció folyamata

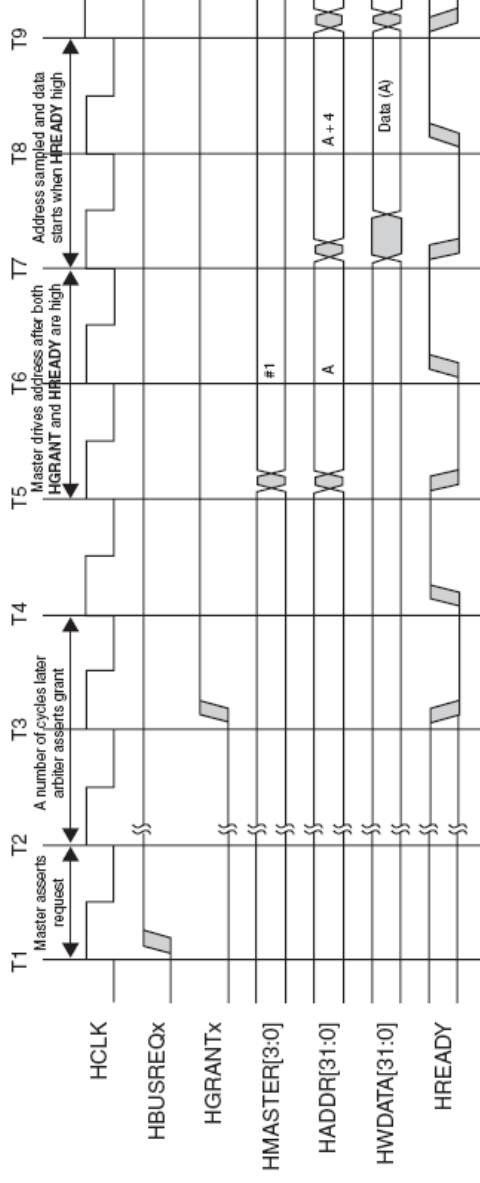


FIGURE 3.7

Cost of arbitration on AHB bus [1]

Source: ARM Inc.

AHB burst példa

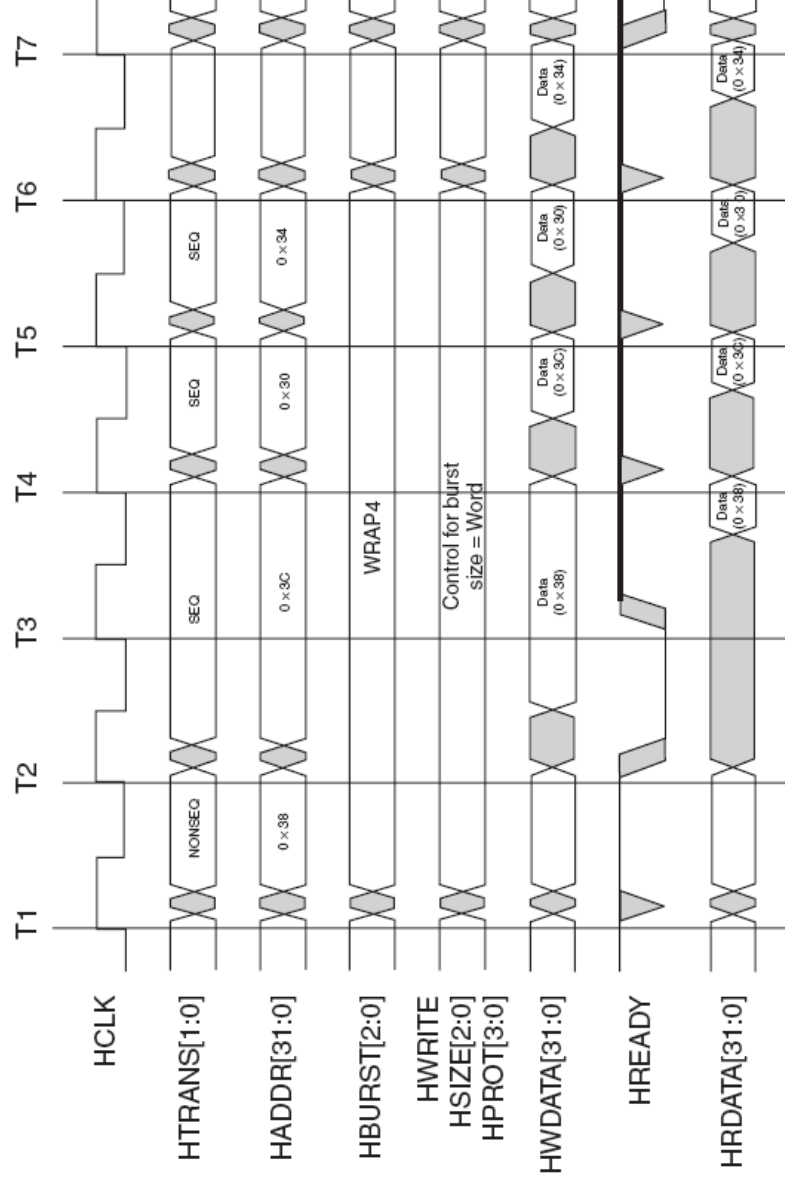


FIGURE 3.9

Example of a burst transfer—a wrapping burst of length 4 on the AHB bus [1]

Source: ARM Inc.

AHB split adatátvitel

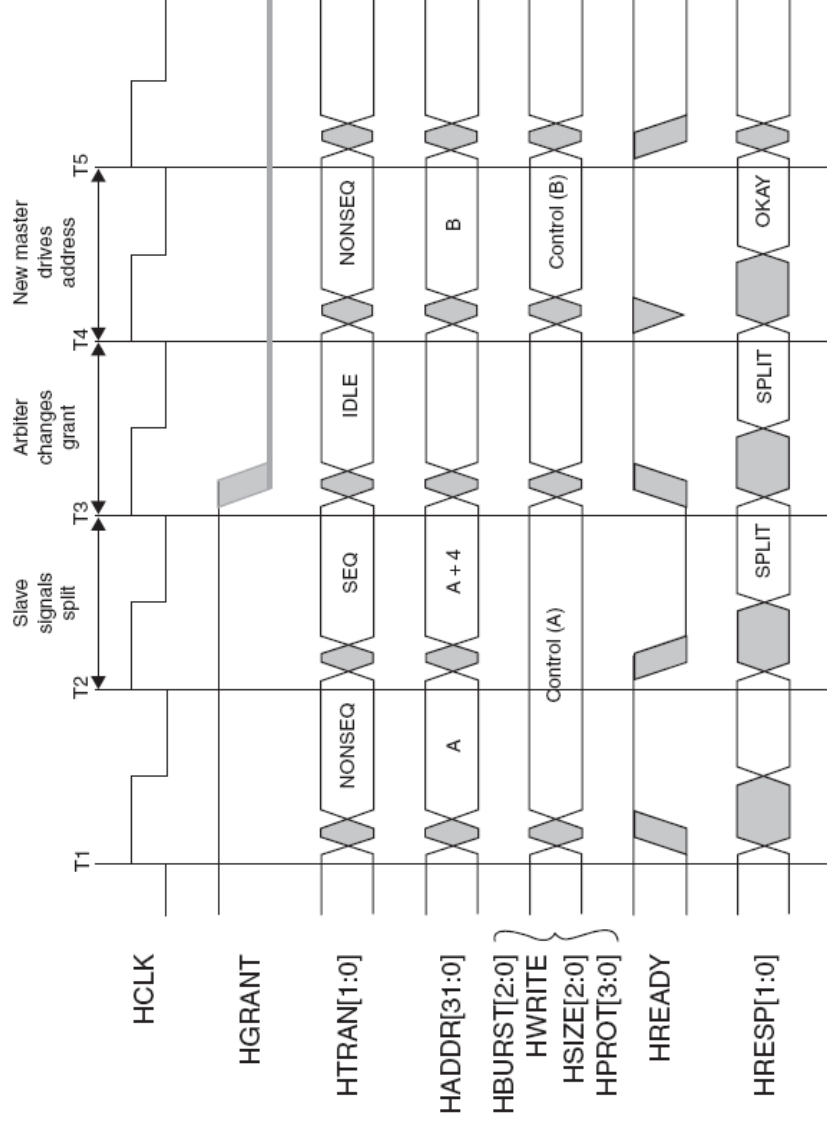


FIGURE 3.10

SPLIT transfer on the AHB bus [1]

Source: ARM Inc.

Komplex AHB rendszerek

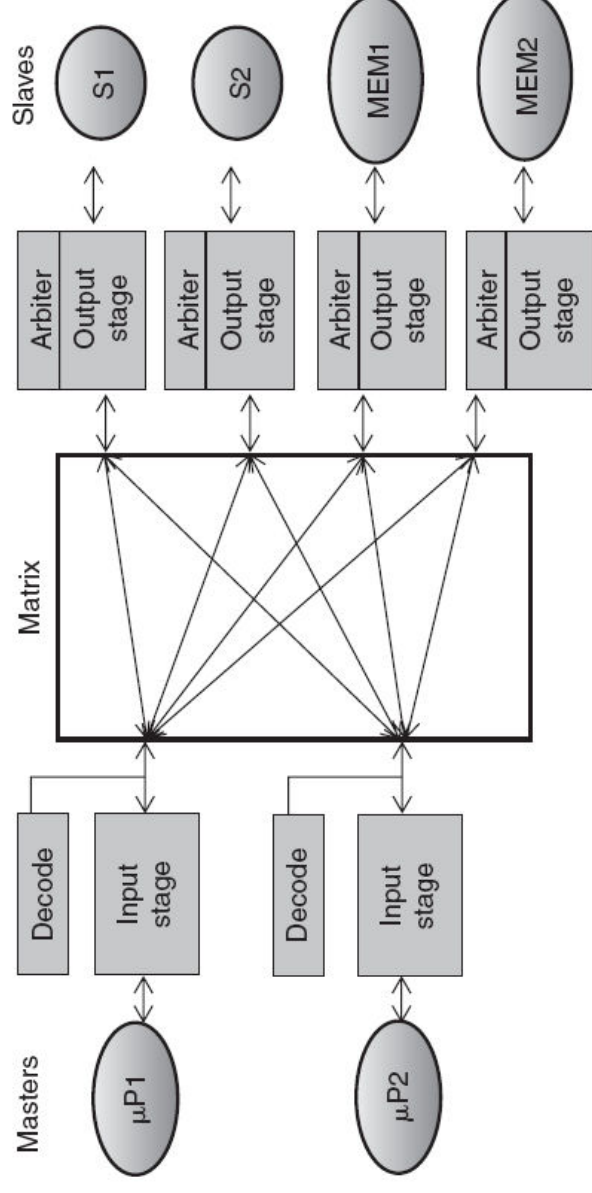


FIGURE 3.11(a)

An example of a 2 master, 4 slave AHB: full bus matrix topology

Komplex AHB rendszer 2.

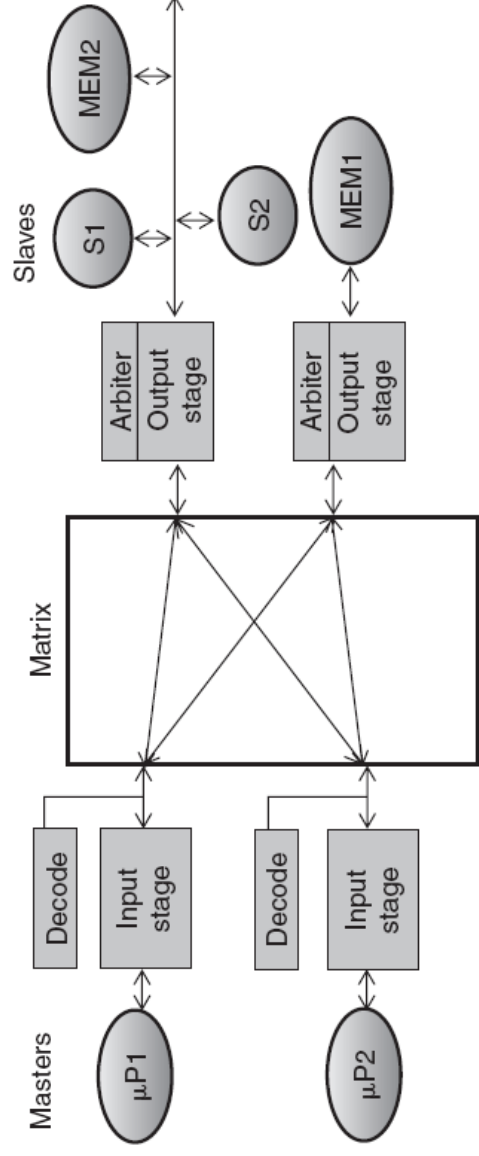


FIGURE 3.11 (b)

Partial bus matrix topology

APB busz

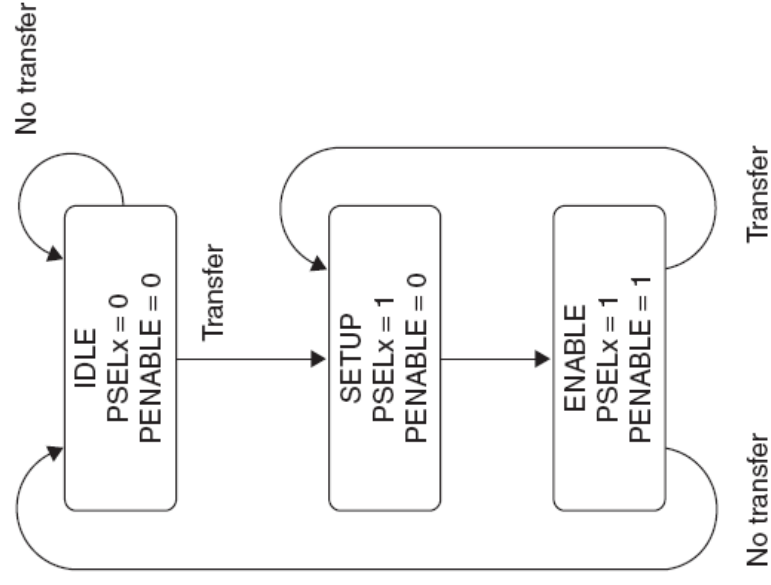


FIGURE 3.12

State diagram representing activity of the APB bus [1]
Source: *ARM Inc.*

APB olvasás buszciklus

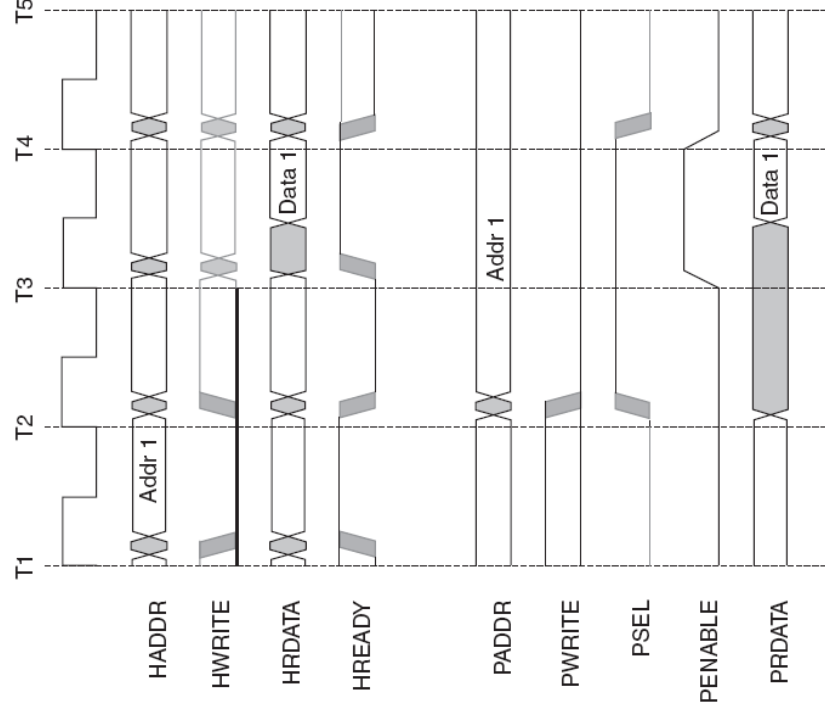


FIGURE 3.13

Read data request from the AHB to the APB bus [1]

Source: ARM Inc.

APB írás buszciklus

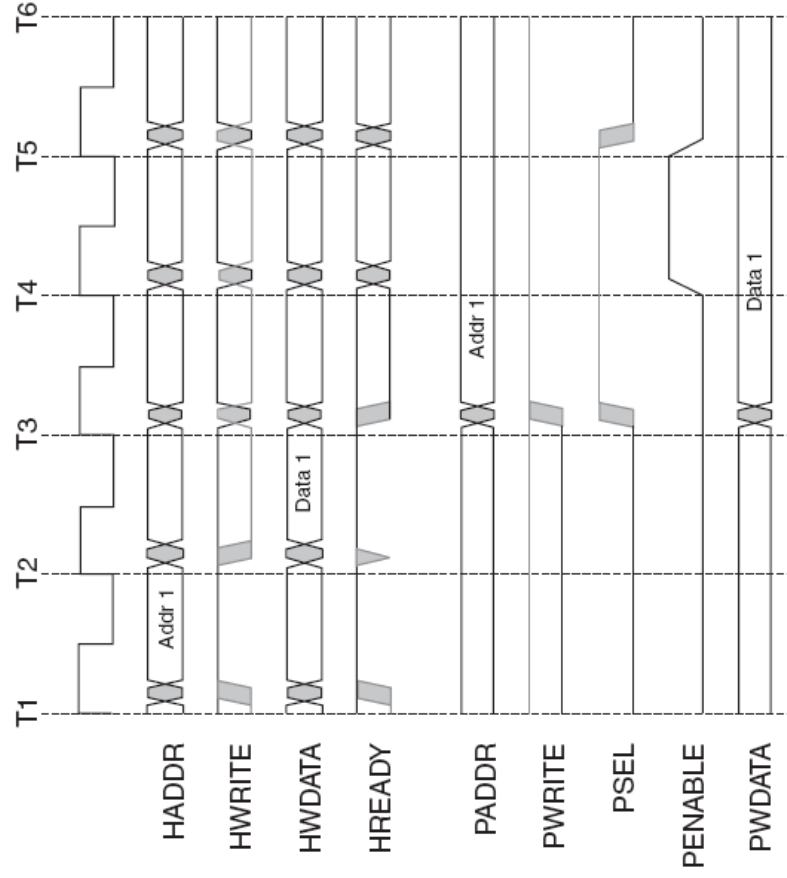


FIGURE 3.14

Write data request from the AHB to the APB bus [1]

Source: ARM Inc.

AMBA 3 - AXI

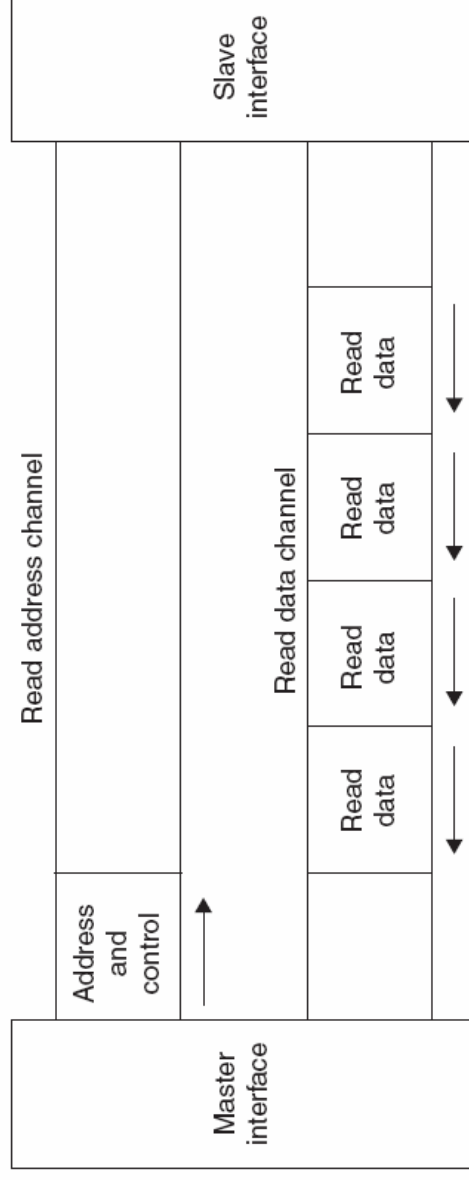


FIGURE 3.15(a)

AMBA AXI channel architecture: read address and read data channels

AMBA 3 - AXI

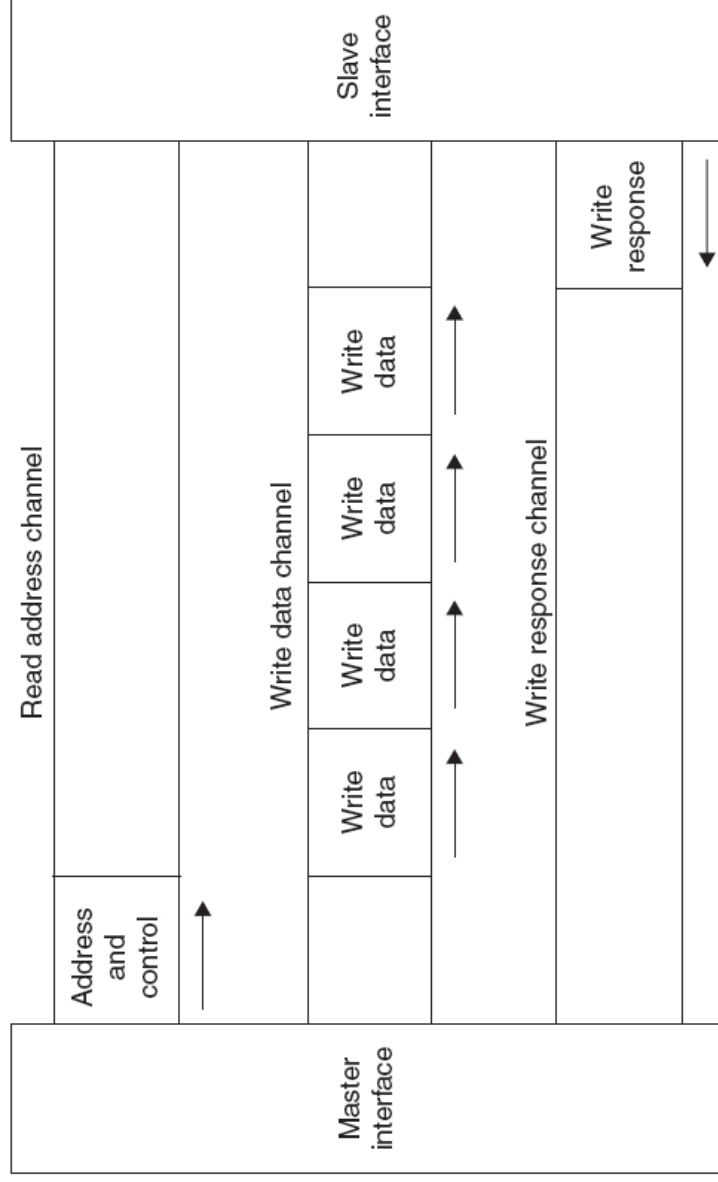


FIGURE 3.15(b)

Write address, write data, and write response channels [2]

Source: *ARM Inc.*

AXI – AHB

Table 3.1 Contrasting features of AXI and AHB

AMBA 3.0 AXI	AMBA 2.0 AHB
Channel-based specification, with five separate channels for read address, read data, write address, write data, and write response enabling flexibility in implementation.	Explicit bus-based specification, with single shared address bus and separate read and write data buses.
Burst mode requires transmitting address of only first data item on the bus.	Requires transmitting address of every data item transmitted on the bus.
OO transaction completion provides native support for multiple, outstanding transactions.	Simpler SPLIT transaction scheme provides limited and rudimentary outstanding transaction completion.
Fixed burst mode for memory mapped I/O peripherals.	No fixed burst mode.
Exclusive data access (semaphore operation) support.	No exclusive access support.
Advanced security and cache hint support.	Simple protection and cache hint support.
Register slice support for timing isolation.	No inherent support for timing isolation.
Native low-power clock control interface.	No low-power interface.
Default bus matrix topology support.	Default hierarchical bus topology support.

AXI tulajdonságok

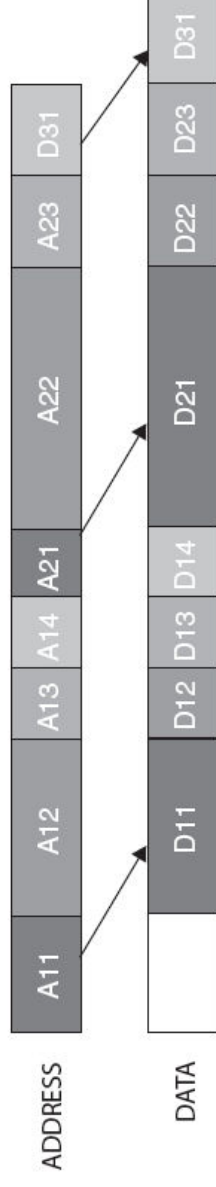


FIGURE 3.16(a)

Burst addressing modes for AMBA 2.0 AHB bus

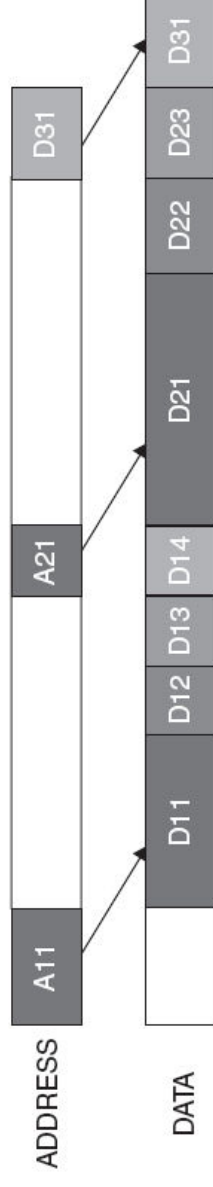


FIGURE 3.16(b)

AMBA 3.0 AXI bus

AXI tulajdonságok

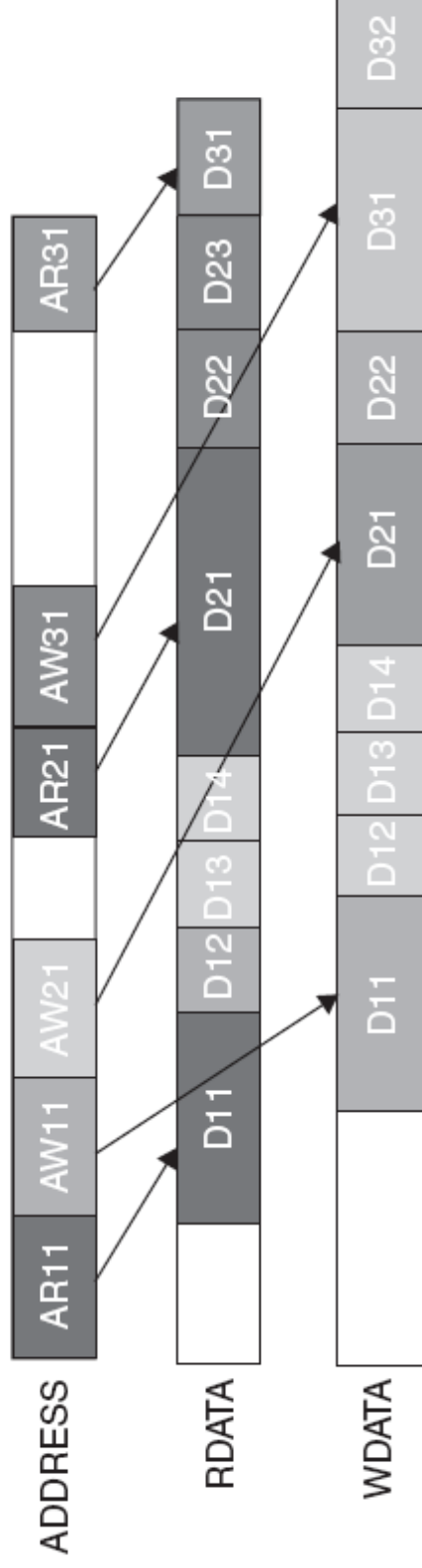


FIGURE 3.17

Better utilization of data buses in AXI

AXI tulajdonságok

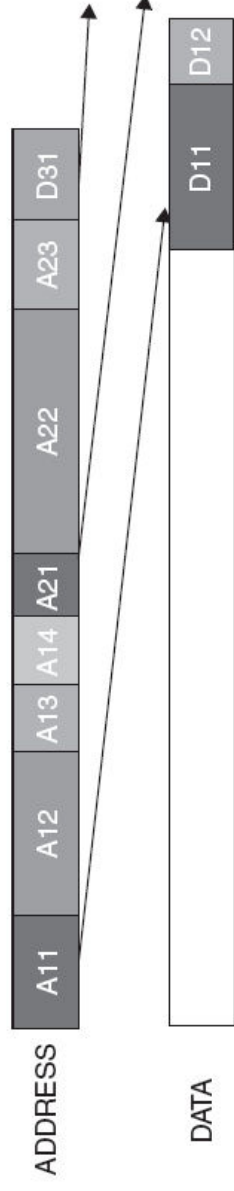


FIGURE 3.18(a)

Transaction sequence for access to a slow slave on AHB bus

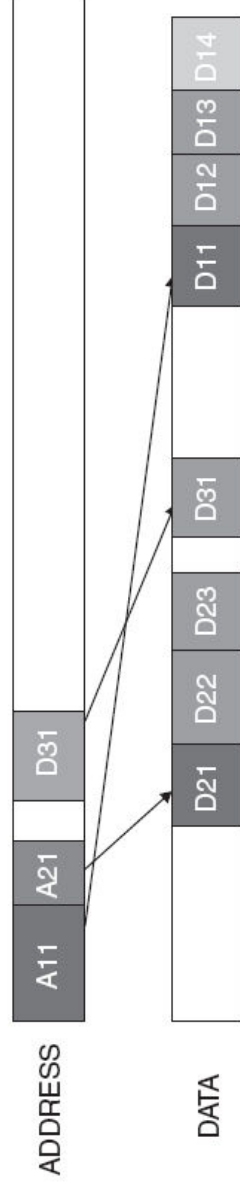


FIGURE 3.18(b)

AXI bus

IBM CoreConnect

- Általános célú SoC buszrendszer
- Három szintű
 - PLB Processor Local Bus
 - OPB On-Chip Peripheral Bus
 - DCR Device Control Register (bus)
- Szolgáltatásai, komplexitása az AMBA-hoz hasonló

IBM CoreConnect

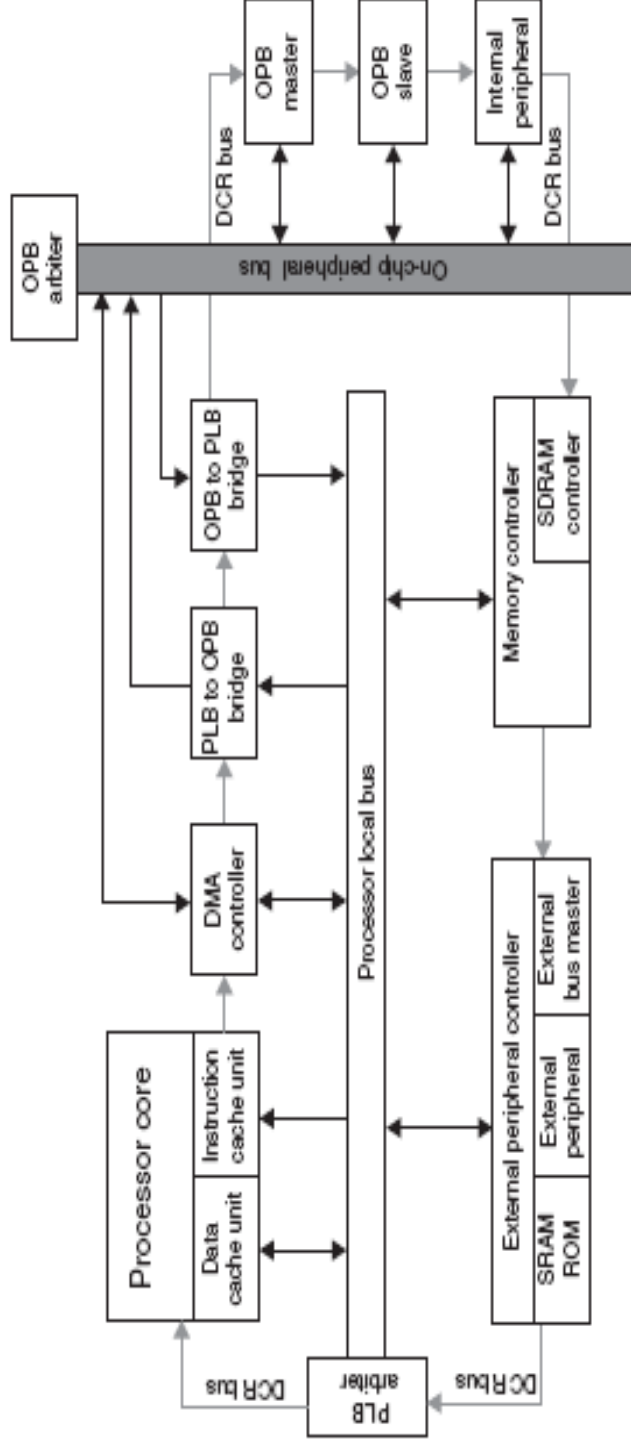


FIGURE 3.19

An example of CoreConnect-based SoC design [3]

Reprint Courtesy of International Business Machines Corporation copyright (2001) © International Business Machines Corporation

IBM CoreConnect – Xilinx EDK

- Xilinx EDK rendszerhez ajánlott buszrendszer
- Választás oka:
 - Önmagában sem rossz, de..
 - Virtex-IIPro sorozat: Beépített Power PC 405 PLB interfésszel kiegészítve
 - Sok létező IP elérhető (nem feltétlenül FPGA-ra, de könnyen portolhatók)
- Bemutató a Xilinx EDK fóliákról

További SoC buszok

- OpenCores WishBone
 - Egyszintű, nagysebességű szinkron kapcsolat
 - Közepes, vagy kis teljesítményű rendszerekhez
 - Nincs átlapolt működés, nincs megszakított átvitel
 - Multi master, 64 bit címzés, 8-64 bit adat
 - Egyedi vagy blokkos átvitelek
 - RMW ciklus a szemafor típusú műveletekhez
 - Ninc fix topológia, illetve többféle rendszer kialakítható

OpenCores Wishbone Pont-pont és adatfolyam

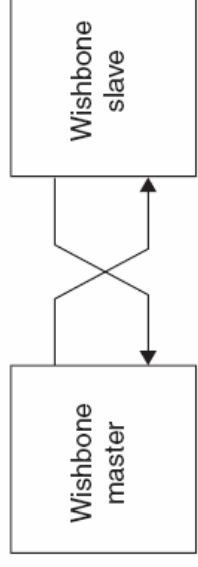


FIGURE 3.30(a)

Different topologies of the Wishbone bus architecture: point-to-point

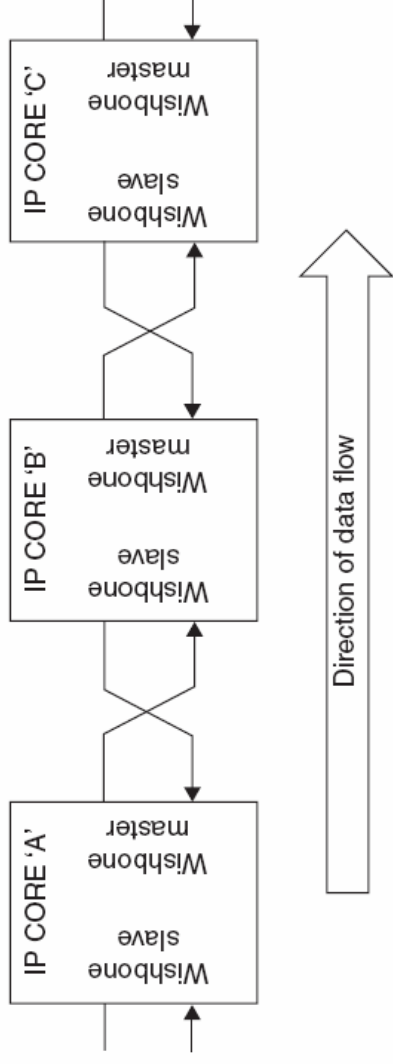


FIGURE 3.30(b)

Data flow

OpenCores Wishbone normál busz

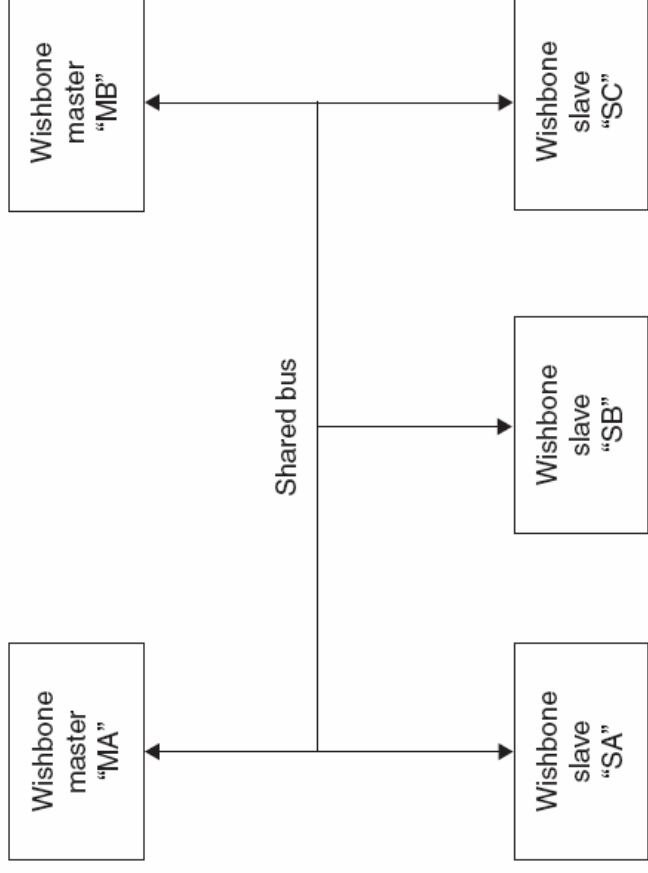


FIGURE 3.30(c)

Shared bus

OpenCores Wishbone hálózat

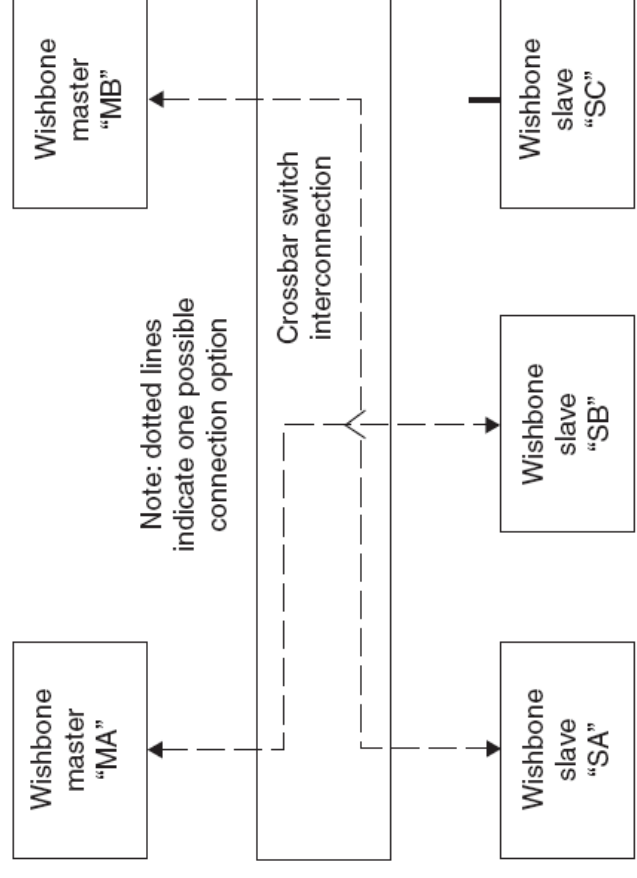


FIGURE 3.30(d)

Full or partial crossbar [6]

OpenCores Wishbone problémák

- Egyszerű, könnyen használható
- Flexibilis, az igényekhez igazítható
- Speciális tulajdonságok kezelése: TAGi jelek
- Nincs előírás az arbitrációra, a hibajelzésre,

OpenCores Wishbone alapkapcsolat

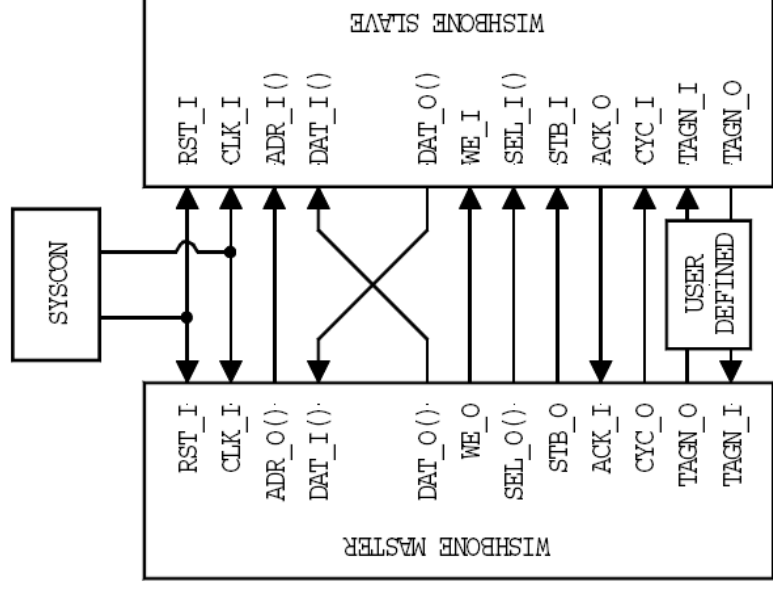


Figure 1-2. Standard connection for timing diagrams.

Wishbone busciklusok

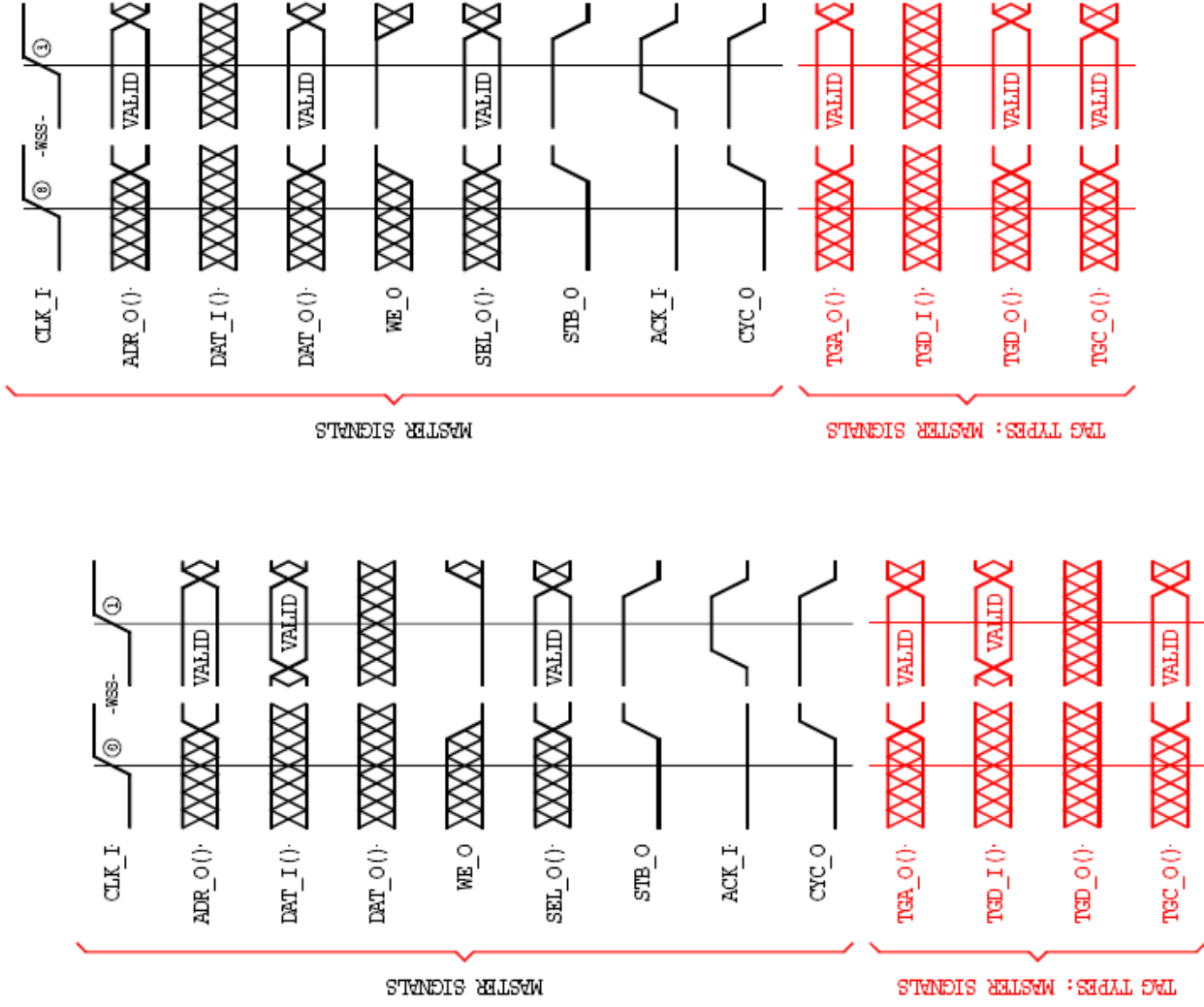


Figure 3-3. SINGLE READ cycle.

Figure 3-4. SINGLE WRITE cycle.

Wishbone blokkos olvasás

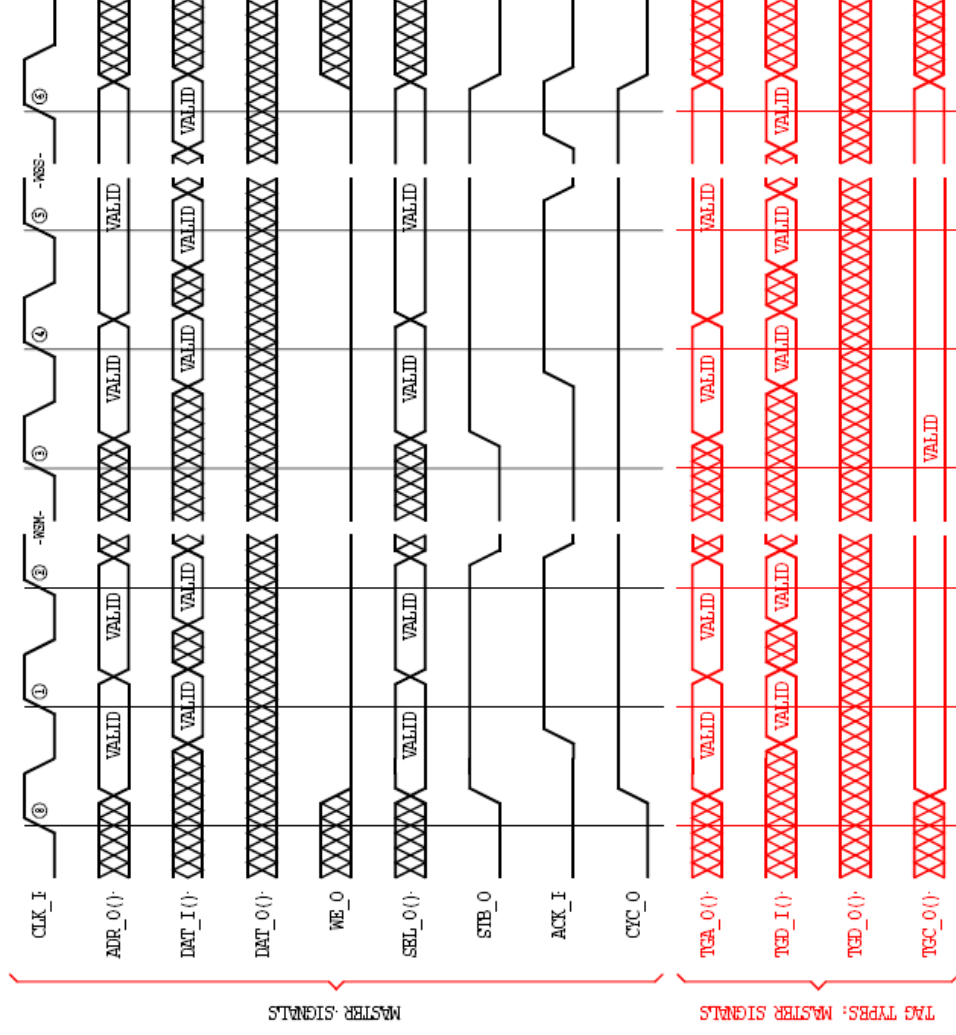


Figure 3-6. BLOCK READ cycle.

Wishbone blokkos írás

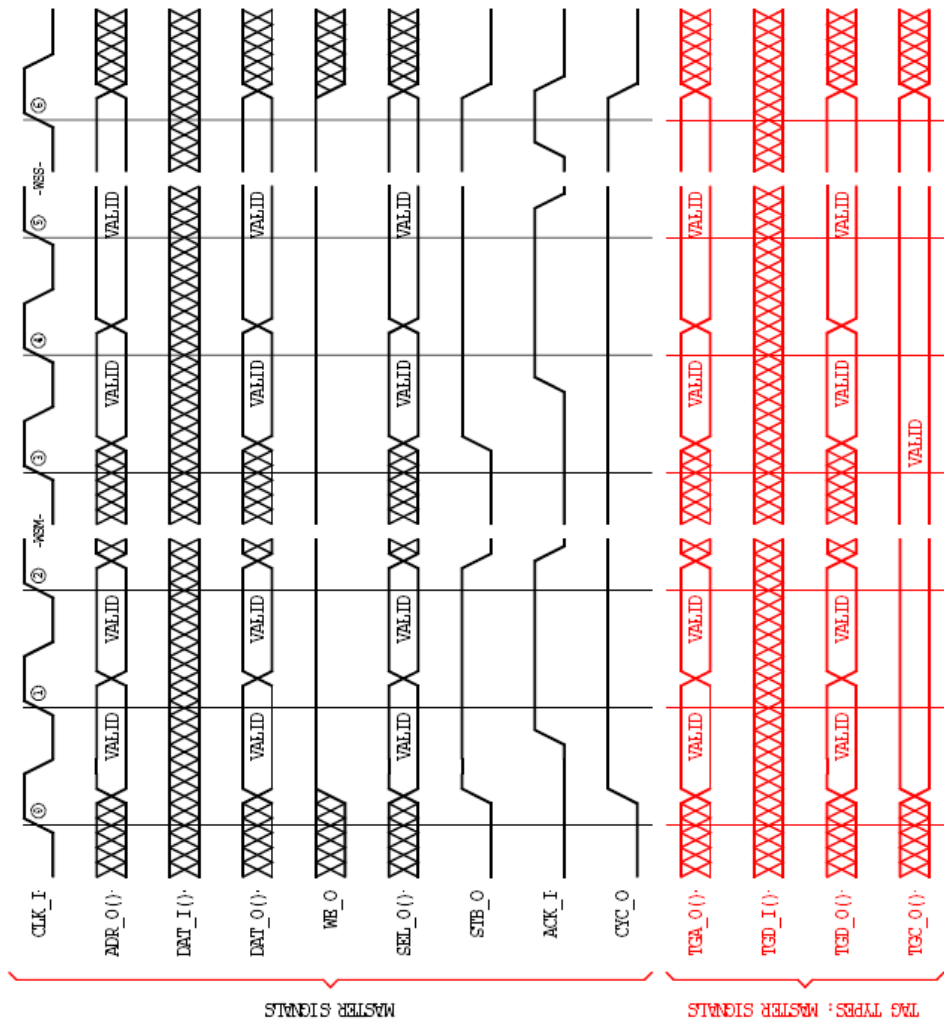


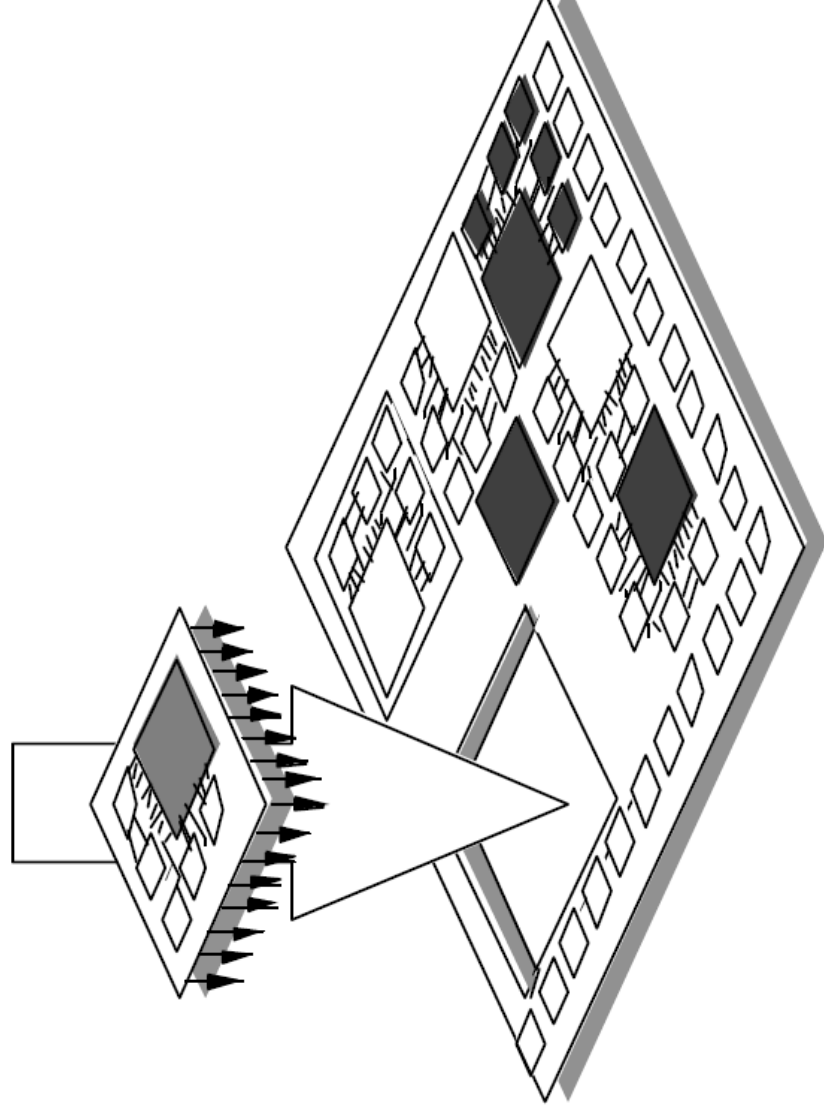
Figure 3-7. BLOCK WRITE cycle.

Socket alapú kommunikáció

- Busz alapú kommunikáció
 - Busz definíció, busz interfész
 - Busz-komponens interfész
- Jelentős befolyás a interfész kialakításában
- Nem teljesen hordozható
 - Buszonként egyedi kialakítást igényel
 - Egyedi tervezés, ellenőrzés, verifikáció

A probléma

- Rendszer építés különböző gyártóktól származó IP blokkok felhasználásával
- A kulcsszó az adapter réteg



Socket alapú SoC rendszer

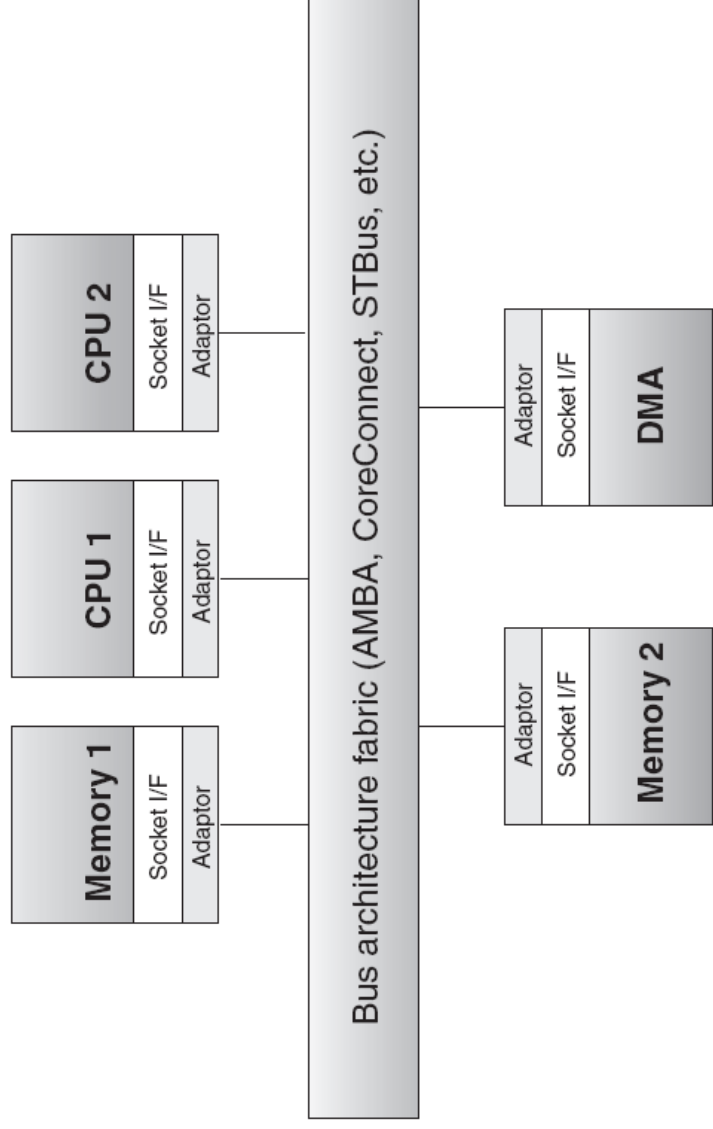


FIGURE 3.33

Example of system implemented with socket-based interface standards

Socket alapú kommunikáció

- OCP Open Core Protocol interfész
 - Szinkron pont-pont kapcsolat
 - Busz független
 - Konfigurálható adatfolyam jelek (cím, adat, vezérlés)
 - Konfigurálható kiegészítő jelek
 - Pipeline és burst átviteli lehetőség
 - Többszörös szálak kialakíthatósága

OCP blokkvázlat

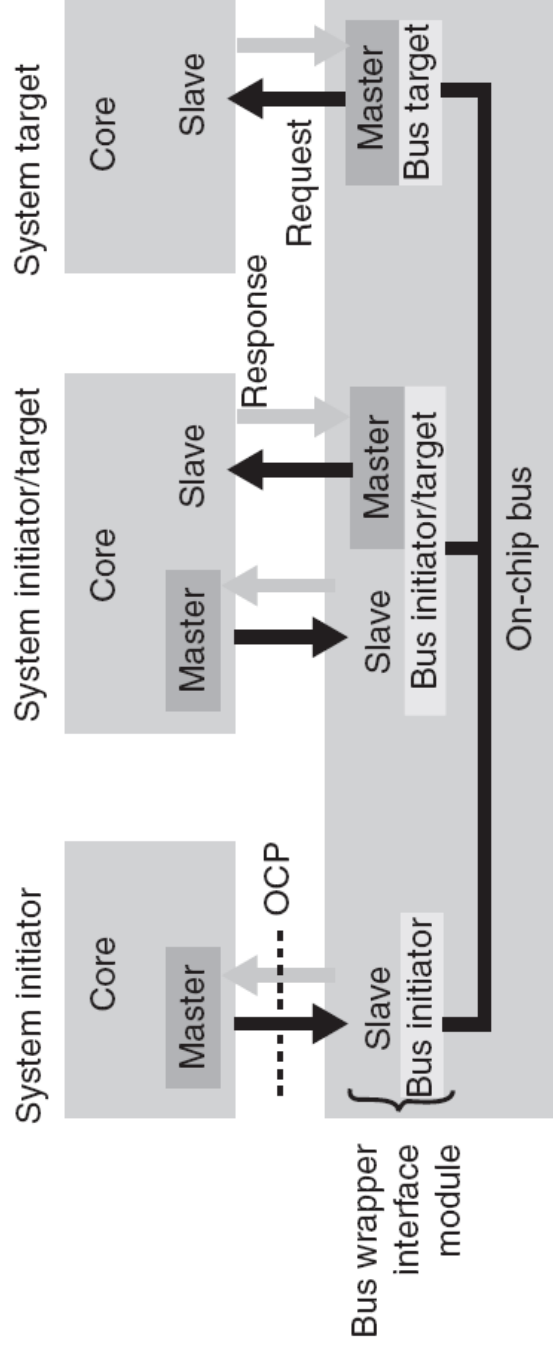


FIGURE 3.34

Example of system implemented with OCP socket-based interface standard [15]

Source: *OCP-IP*

OCP jelek

Data Flow

Clk
EnableClk
MAddrSpace
MAtomicLenght
MBlockHeight
MBlockStride
MBurstPrecise
MBurstSeq
MBurstSingleReq
MByteEn
MCmd
MConnID
MReqInfo
MReqLast
MReqRowLast
MSecure
MTagID
MTagInOrder
MThreadID
SCmdAccept

request

MRespAccept
SData
SDataInfo
SResp
SRespInfo
SRespLast
SRespRowLast
STagID
STagInOrder
SThreadD

response

MDataByteEn
MData
MDataInfo
MDataLast
MDataRowLast
MDataTagID
MDataThreadID
MDataValid
SDataAccept

data handshake

Test

Scanctrl
Scanin
Scanout
ClkByp
TestClk
TCK
TDI
TDO
TMS
TRST_N

Sideband

MThreadBusy
SDataThreadBusy
SThreadBusy

MReset_n
MError
MFlag
SError
SFlag
SInterrupt
SReset_n

Control
ControlWr
ControlBusy
Status
StatusRd
StatusBusy

OCP jelek

- Adatfolyam csoport
 - Alapvető átviteli jelek
 - Óra, cím, adat (Read, Write), típus, szinkronizáció
 - Típus: Read, Write, Idle, Broadcast, Exclusive read, Linked read, Non-posted write, Conditional write
 - Szinkronizáció
 - Egyszerű kiegészítések adatbájt, paritás, FIFO jelzések, cache kezelés,
 - Burst kiegészítések fix/tetszőleges, címkezelés(wrapping/incrementing/streaming)
 - Packing/nonpacking, eltérő szélességű interfészekre
 - Tag/ID kiegészítések
 - Eltérő sorrendű végrehajtásokkezelésére
 - Szál kiegészítések
 - Master és slave egységek számára a szálak azonosítására

OCP protokoll hierarchia

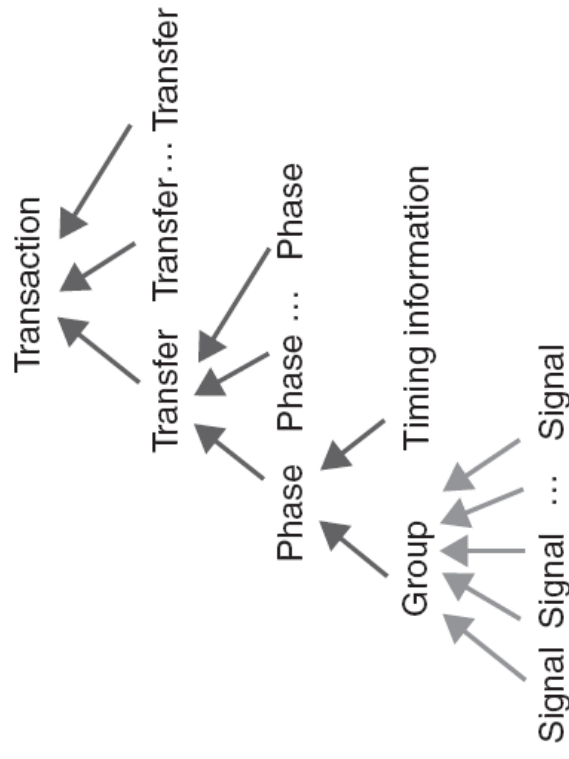


FIGURE 3.35

Hierarchy of elements that compose the OCP [15]

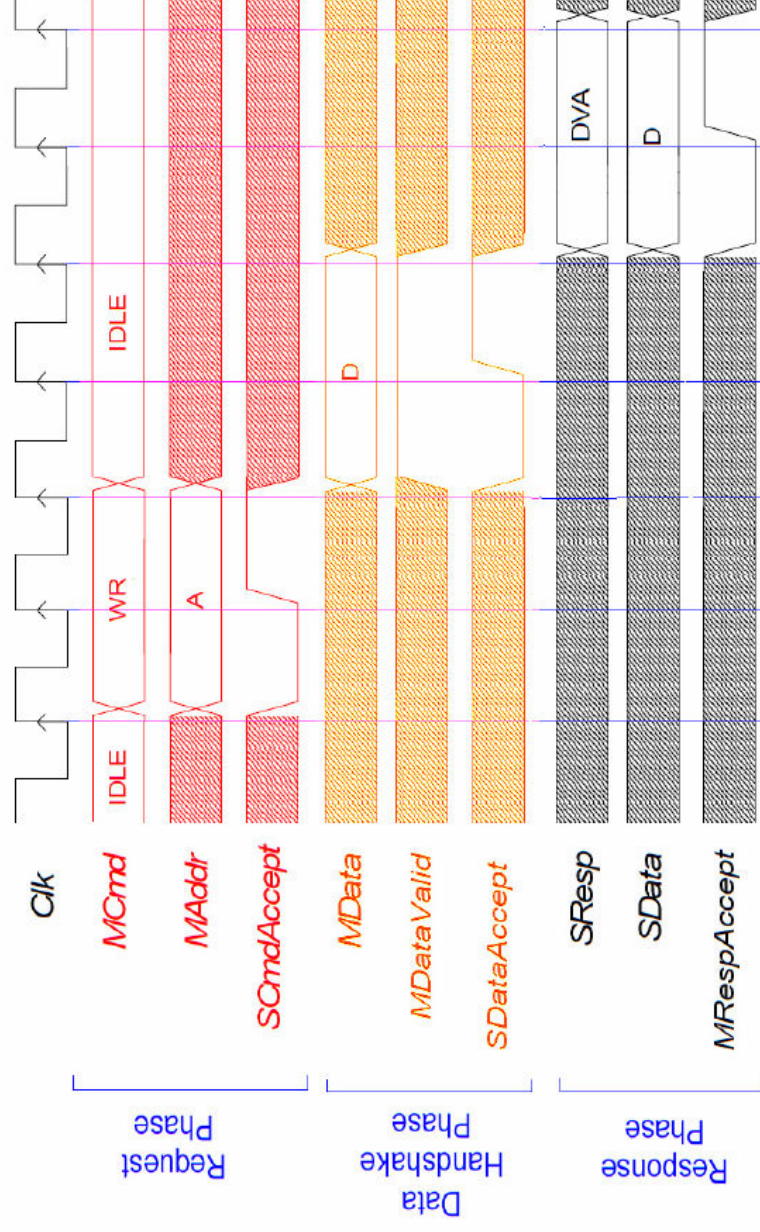
Source: OCP-IP

OCP hierarchia

- Jelek csoportja aktivitás alapján
 - Pl. adatátviteli jelek: kérés, válasz, szinkronizáció
- Fázisok, hasonlóan: kérés, válasz, szinkronizáció
- Átvitel: több fázisból áll, de mindig
 - Kérés szinkronizáció {válasz}
- Tranzakció: néhány, összetartozó átvitel együttese (adott címek, adatok és sorrend)

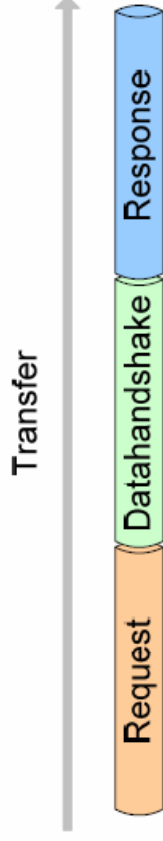
OCP átvitelek

- Egy teljes OCP átvitel

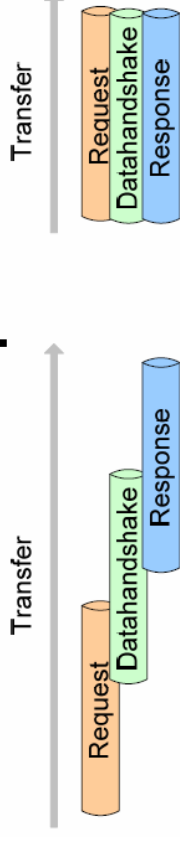


OCP átvitel típusok

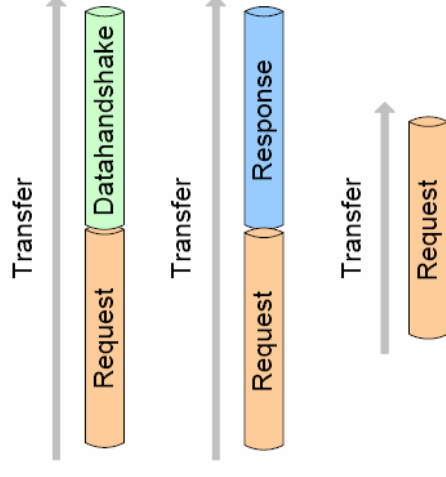
Normál



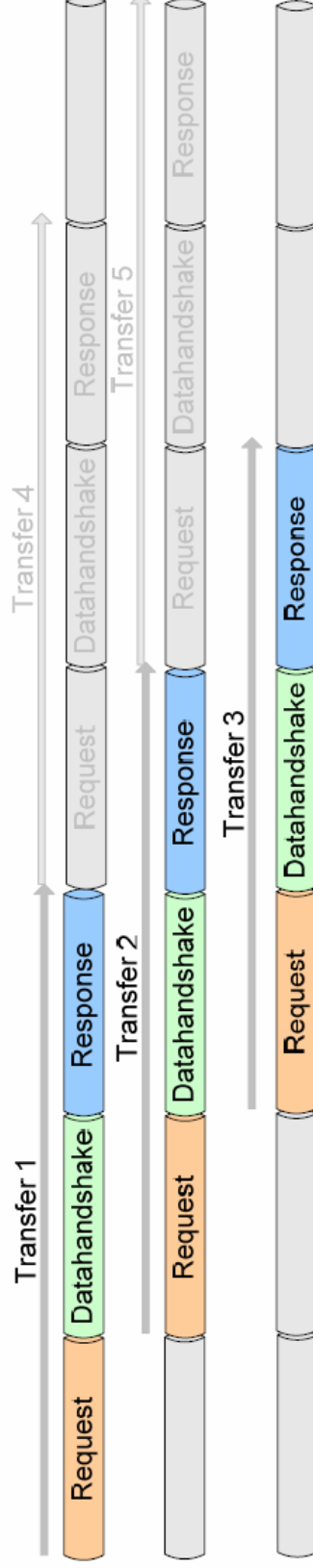
Átlapolt



Hiányos



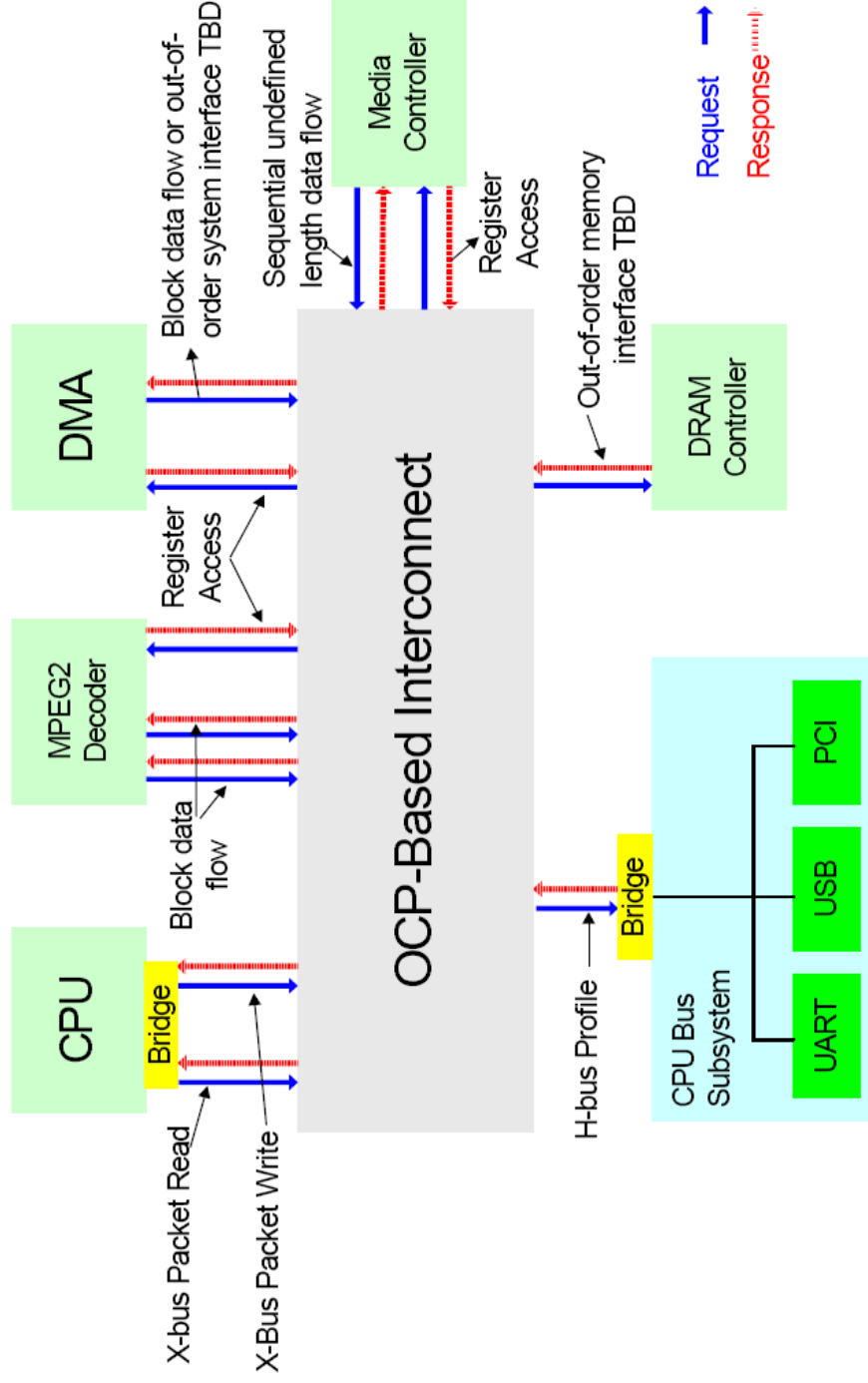
Pipe-line



OCP profilok

- Az OCP egyik erőssége a kialakított szabványos kapcsolatok lehetőségében van
- Előnyök:
 - Csökken az inkompatibilitás veszélye
 - Egyszerűsített illesztő rétegek a más típusú rendszerekhez
 - Könnyebb tesztkörnyezet fejlesztés
- Kétfajta alkalmazási cél
 - Új IP magok használjanak szabványos natív profilokat egymás között
 - Külső eszközök közötti hídkapcsolatokhoz pedig egyedi profilok fejlszthetők

OCP profilok



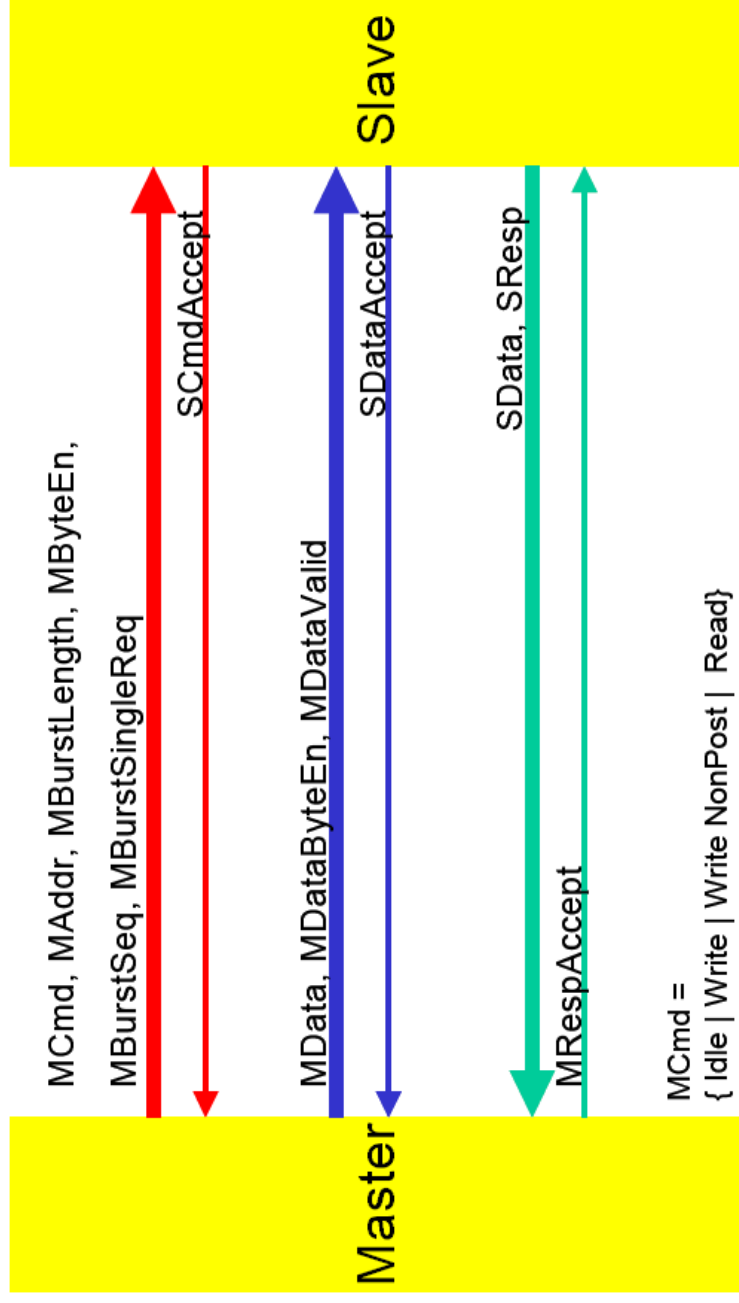
OCP profilok

- Natív OCP profilok
 - Blokkos adatfolyam profil
 - Szekvenciális tetszőleges adatfolyam profil
 - Regiszter hozzáférés profil
- Híd profilok
 - Szimpla H-busz profil
 - X-busz packet írás profil
 - X-busz packet olvasás profil
- Biztonsági profil

Natív OCP profil

- Blokkos adatfolyam profil

Figure 51 Block Data Flow Signal Processing



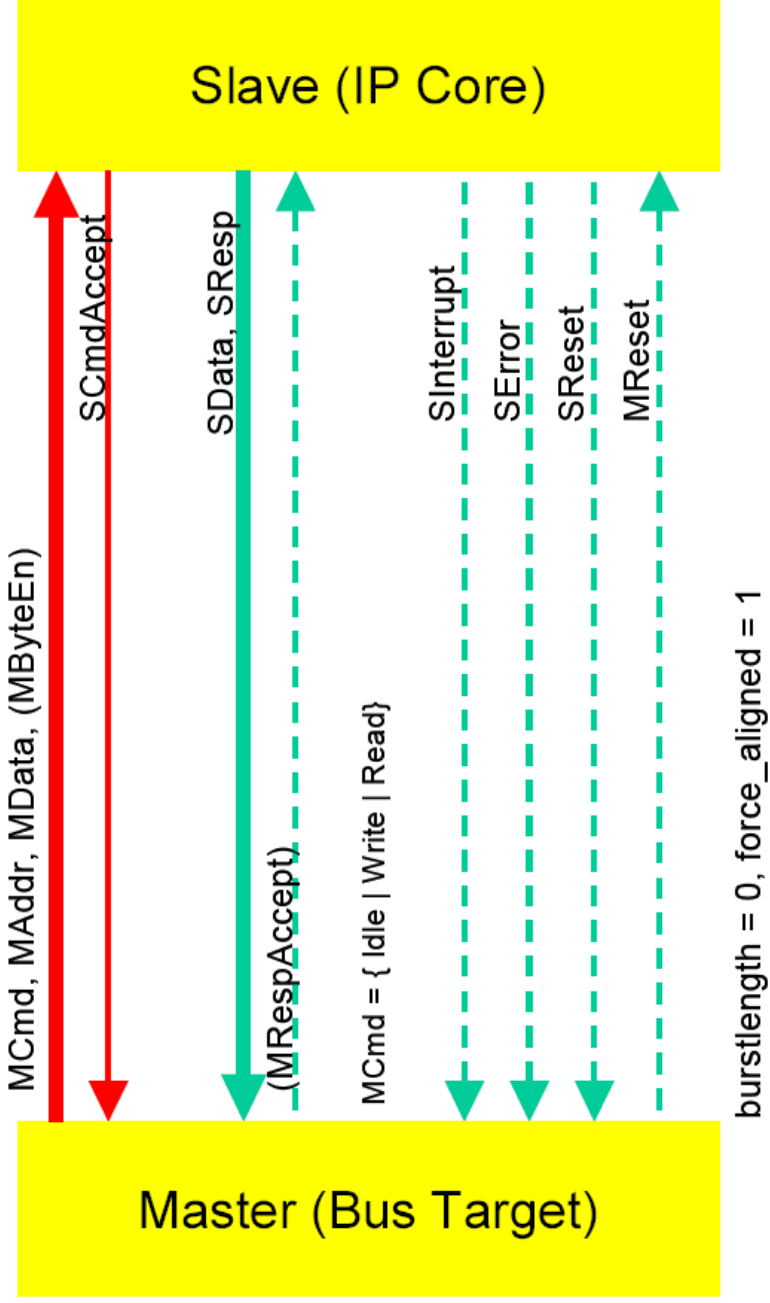
Natív OCP profil

- Blokk alapú (méret = 1..N) átvitel
- 1 kérés, több adat, cím inkrementálás vagy stream modell szerint
- Szétválasztott (pipe-line) parancs és adatfolyam
- 32 bit cím, normál adat és blokkméret
- Termelő /fogyasztó szinkronizálás
- Egyszálú megoldás

Natív OCP profil

- Regiszter hozzáférés profil

Figure 53 Register Access Signals Processing



Natív OCP profil

- Egyedi adatátviteliek, burst nélkül, egyszerű címzéssel
- Címzéseléség igény szerint, adatszéléséség ajánlott: 32 bit (a regiszter szélesség szerint)
- Írás ciklus is nyugtázva
- Segéd jelek, mint RESET, IRQ, ERROR, DMA_RDY, stb.

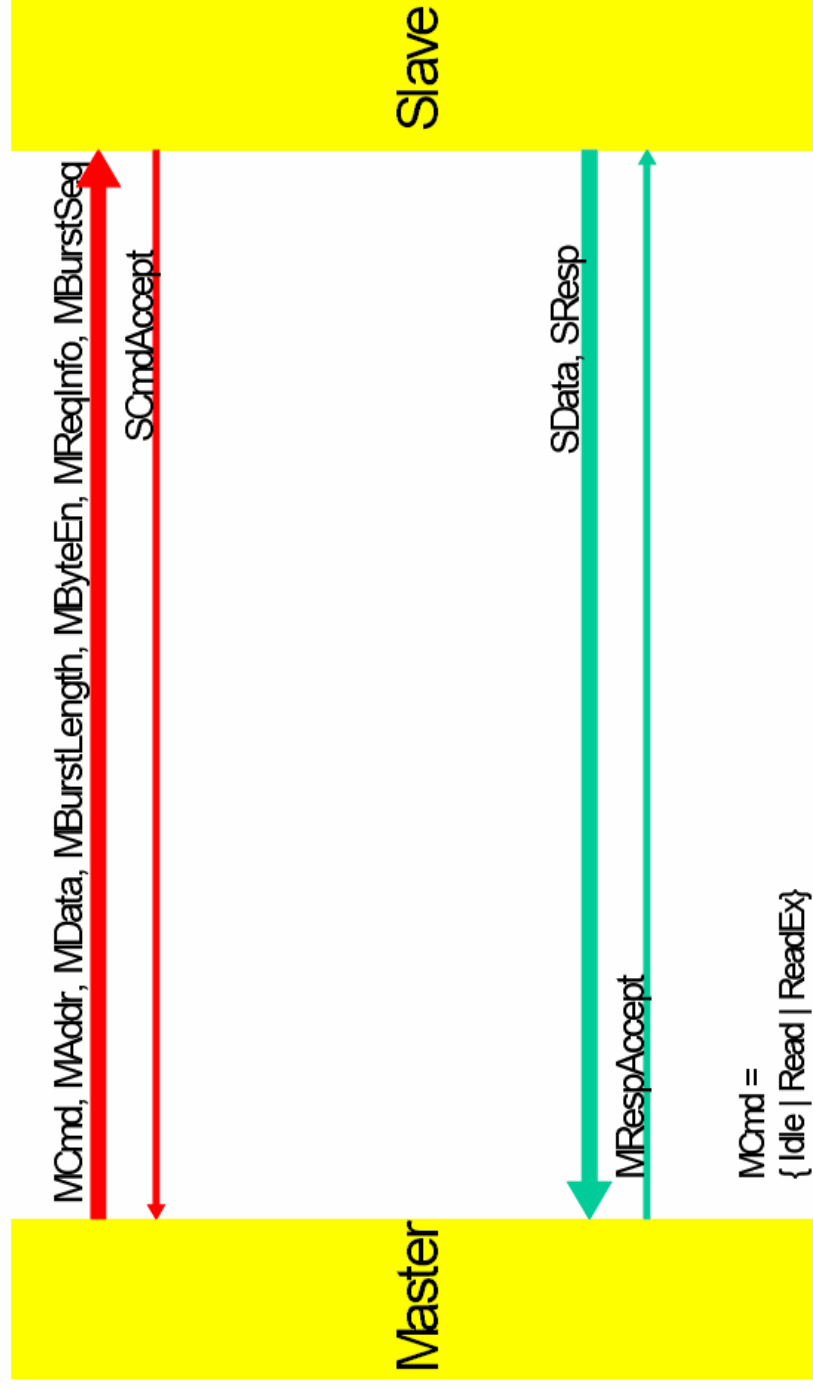
Híd profilok

- A cél az elterjedt, szabványos buszokhoz szükséges kapcsolat realizálása
- Master vagy slave funkció is lehetséges
- Két típus:
 - H-bus (talán az AMBA AHB-re utal..)
 - Normál egyedi és blokkos átvitelek
 - X-bus (talán az AMBA AXI-re utal...)
 - X-bus Packet Write
 - X-bus Packet Read

H-bus híd profil

- Egyszerű busz híd kapcsolat

Figure 54 Simple H-Bus Signal Processing



H-bus híd profil jellemzői

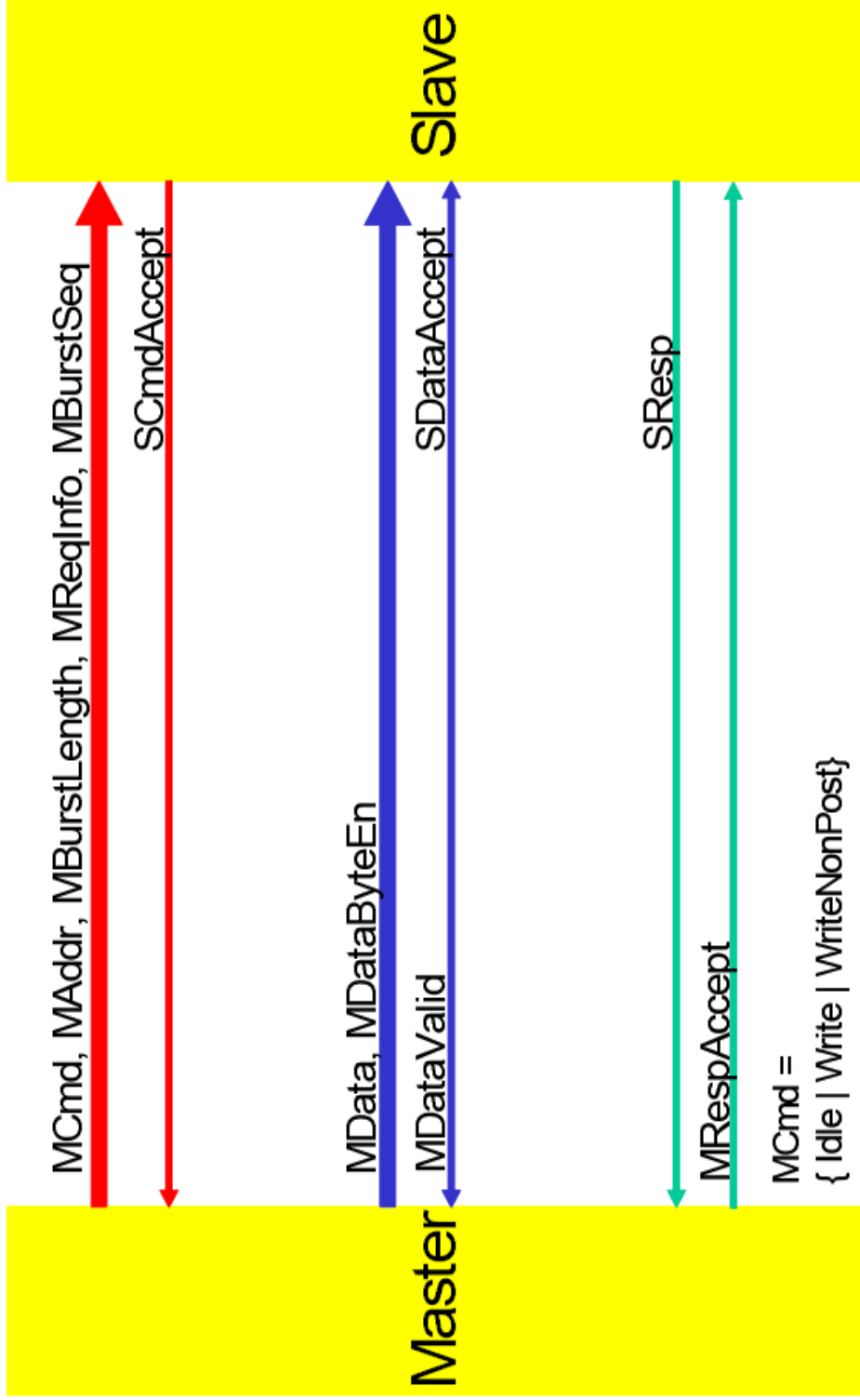
- Egyszerű címzéssel leképezett kommunikáció
- Normál cím és adatméretek, bájt engedélyezéssel
- Fix burst méretek
- Speciális tulajdonságok (pl. cache) az MReqInfo mezőben adhatók meg

X-bus profilok (WRITE, READ külön)

- Kifejeztem olyan mastereknek, amelyek egy-egy kérésre több fázisban szeretnének adatot írni vagy olvasni (de csak egy irányban)
- Csomag alapú kapcsolatok felépítésére szolgál
- Az adat szervezés, érvényesség ellenőrzés a master feladata

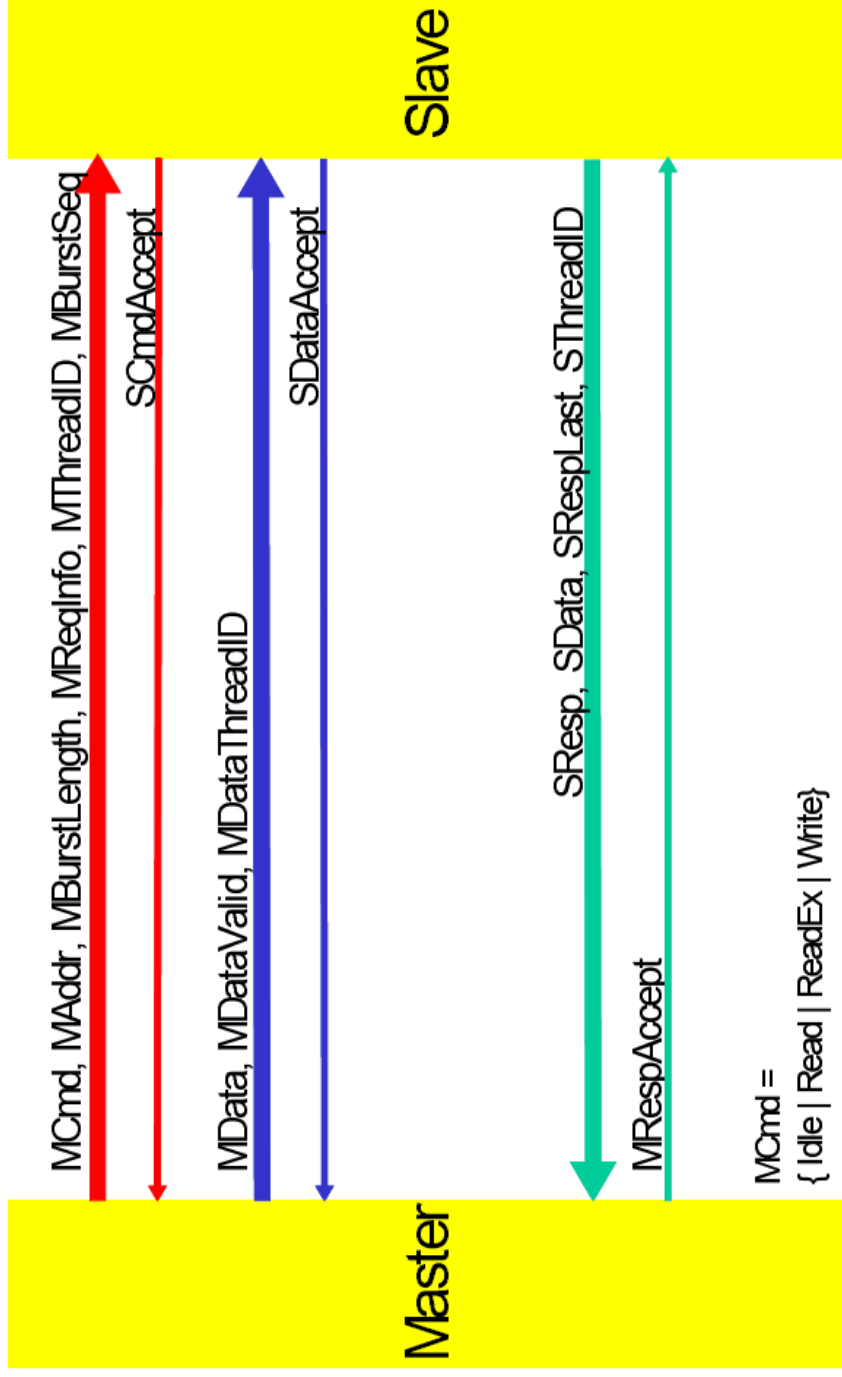
X-bus WRITE

Figure 55 X-bus Packet Write Signal Processing



X-bus READ

Figure 56 X-bus Packet Read Signal Processing



VSA Virtual Component Interface

- Hasonló célú, mint az OCP
- Pont-pont, szinkron, socket alapú protokoll
- Három verzió:
 - PPCI: Periféria VCI
 - BVCI: Basic VCI
 - AVCI: Advanced VCI

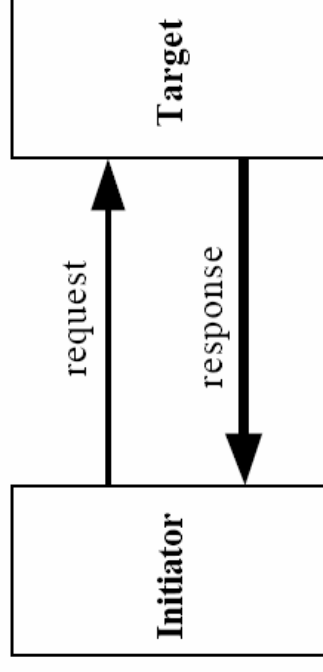


Figure 1: VCI is a Point-to-Point Connection

A VCI használata OCB esetén

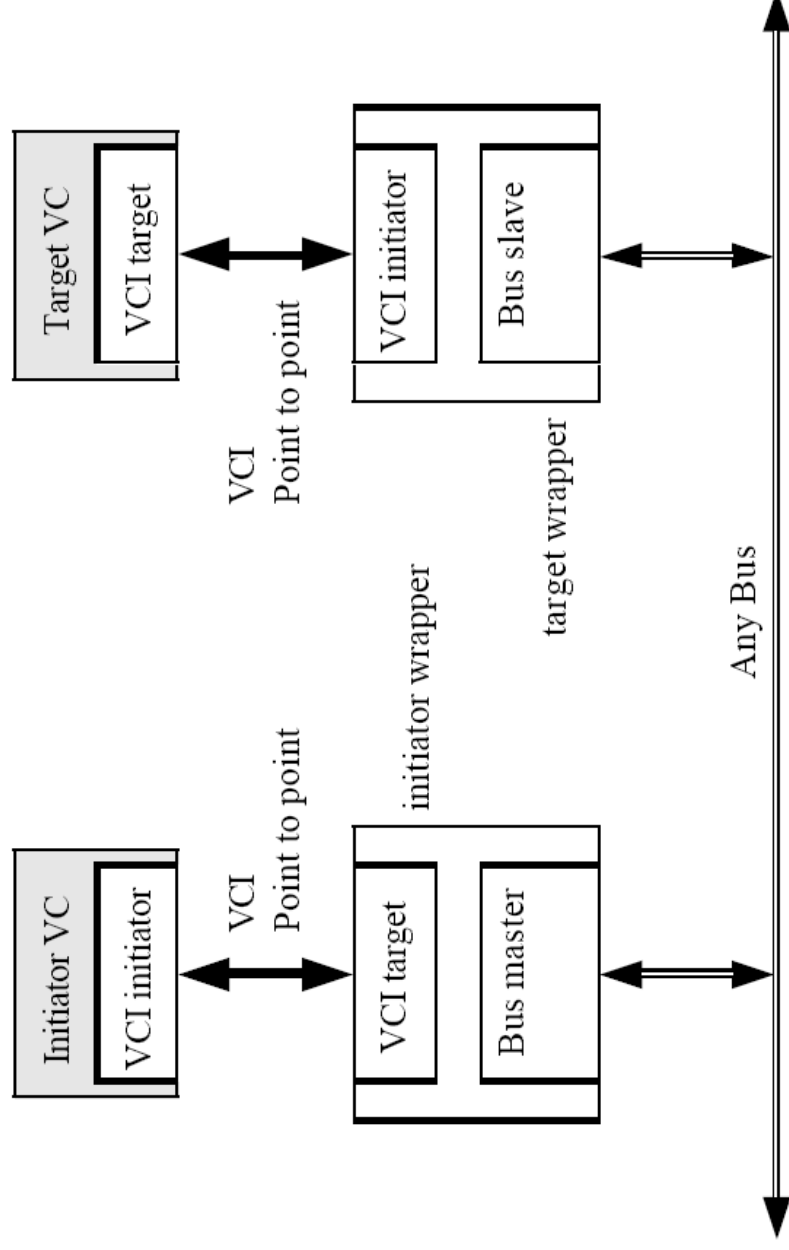


Figure 2: Two VCI Connections Used to Realize a Bus Connection

VCI split protokol

- **PVCI:**
 - Egyszerű, kérés-válasz protokol
- **BVCI és AVCI:**
 - Megosztott protokol, a kérések és válaszok szétválaszthatók
 - Újabb kérések adhatók ki, a válaszok beérkezése előtt
 - A válaszok sorrendje követi a kérésekét
- **AVCI:**
 - Lehetséges Out-of-Order válaszolás is

PVCI protokoll

- Egyszerű kapcsolat, szokásos jellemzők
- Szinkron/aszinkron verzió
 - Mindkettő kézfogásos

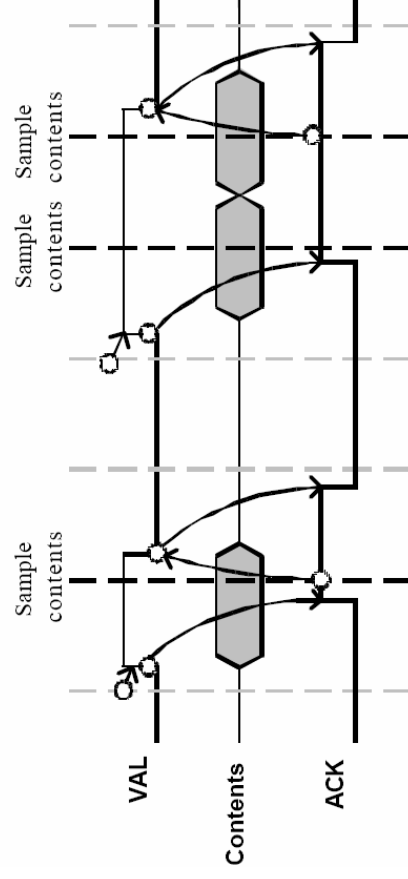


Figure 4: Control Handshake (Asynchronous ACK)

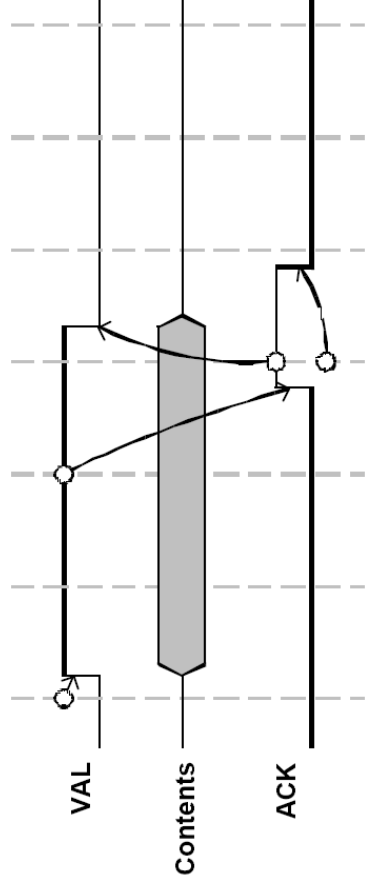


Figure 5: Fully Synchronous Control Handshake (ACK Late by Two Cycles)

BVCI protokol

- Általános interfész, a legtöbb igény kielégítésére
- Adat cellás átvitelek (1, 2, 4, 8, 16 bájtt)
- A cellákat egy csomagba lehet foglalni
 - Ez akár egy burst is lehet
 - A csomagátvitel atomi jellegű, nem megszakítható
 - Használjunk kis csomagokat, mert blokkol!
 - Ezért lehet jó a csomagok láncolata, ami periódikusan felszabadítja az interfészt

BVCI tranzakciók



Figure 15: System Transaction Layer View of Information Transfer over the VCI

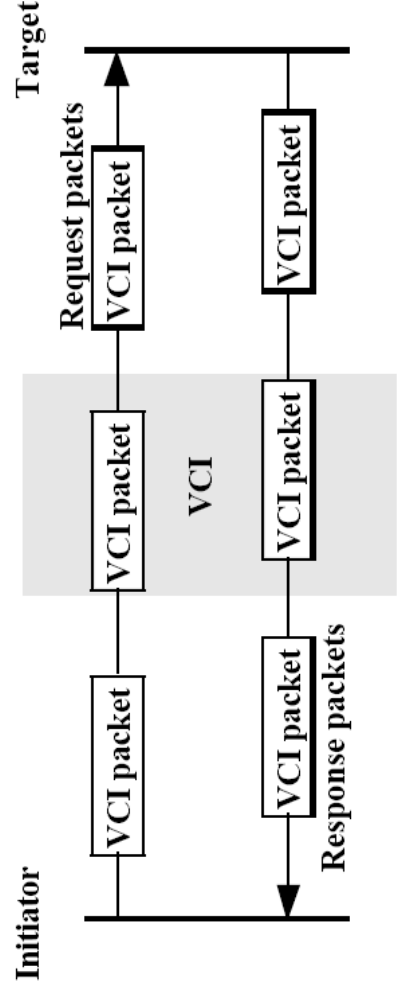


Figure 16: Packet Layer View of Information Transfer over the VCI

BVCI tranzakciók

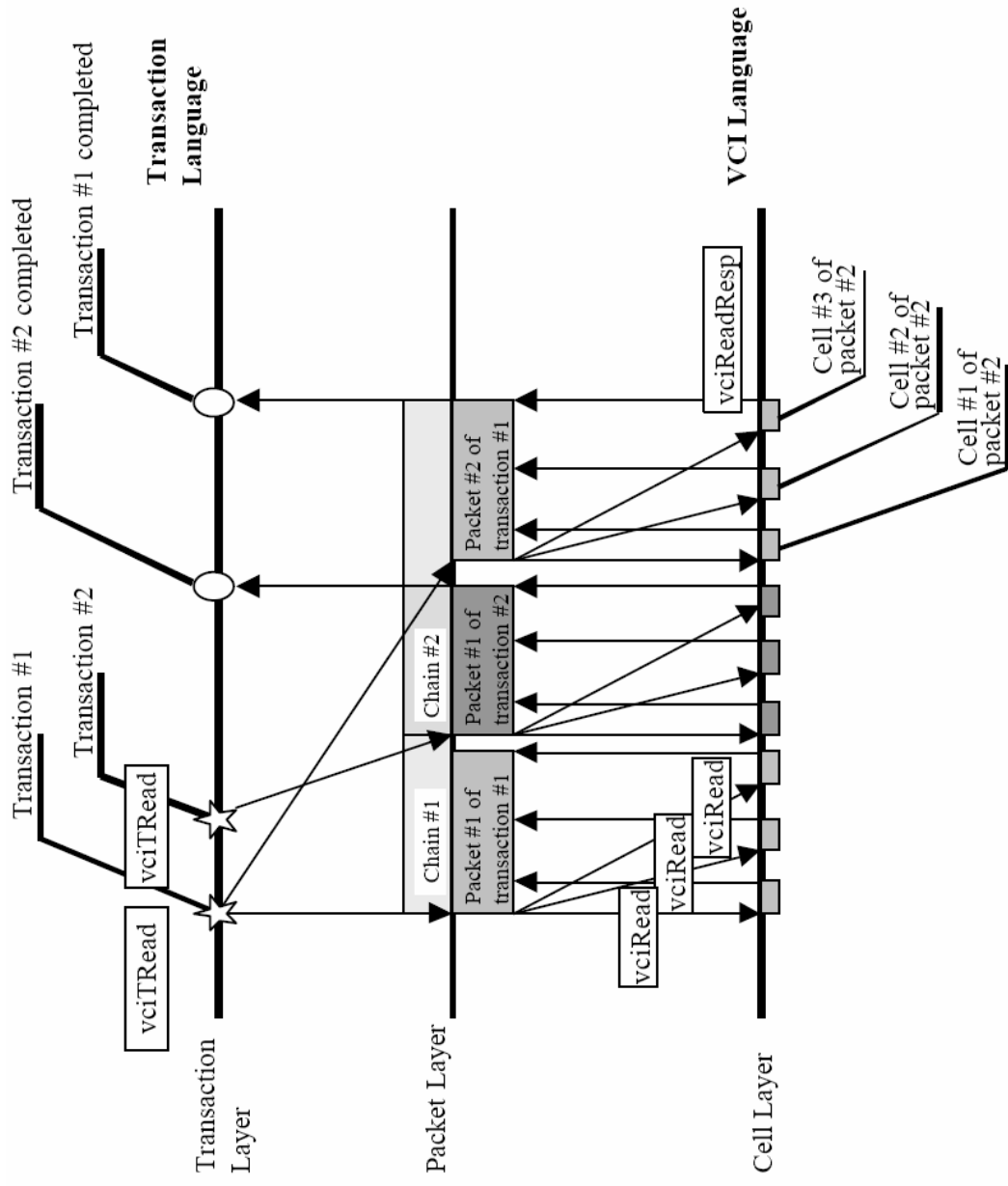


Figure 51: Normal Packet Model

AVCI protokoll

- Mint a BVCI, de a fejlettebb csomagmodell
- Lehet eltérő számú cella a kérés és válasz között
- Támogatás nem sorrendi végrehajtáshoz

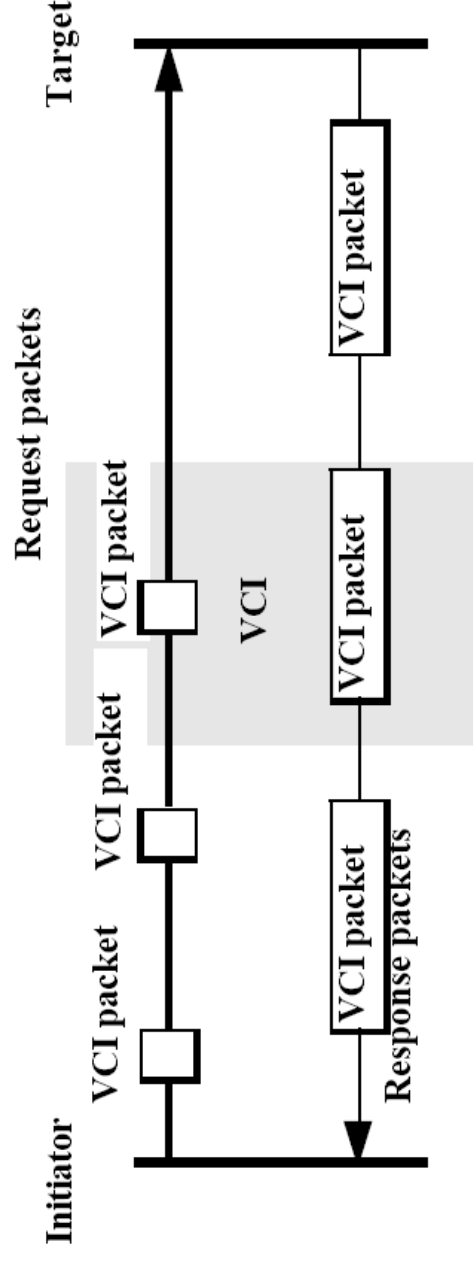


Figure 31: Advanced Packet Model

AVCI tranzakciók

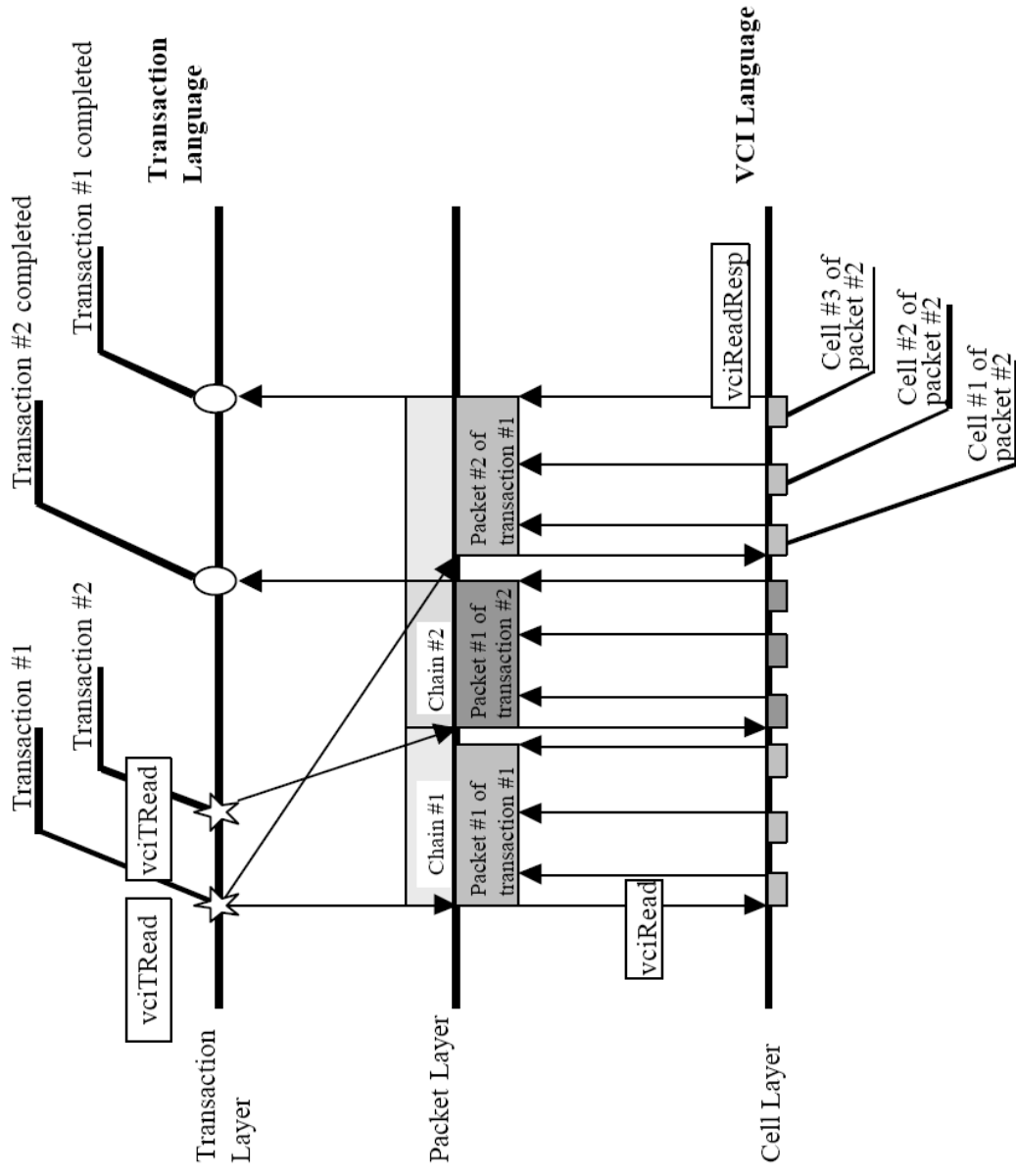


Figure 52: Advanced Packet Model

Tervezési tippek

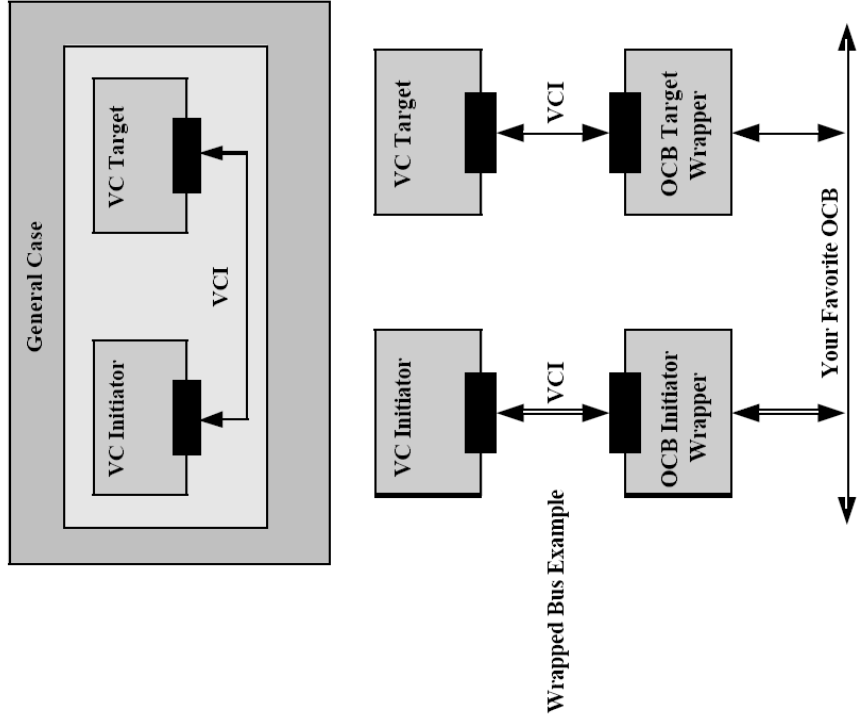


Figure 43: VCI Block Diagram

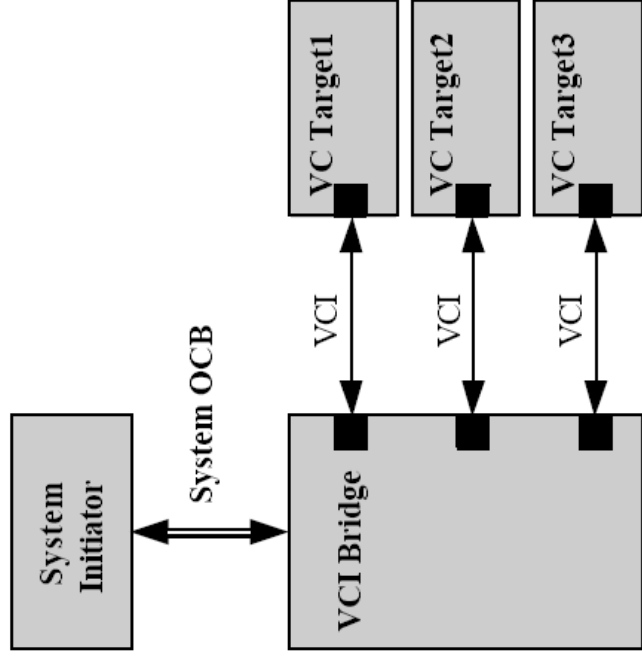


Figure 44: VCI with Star Topology

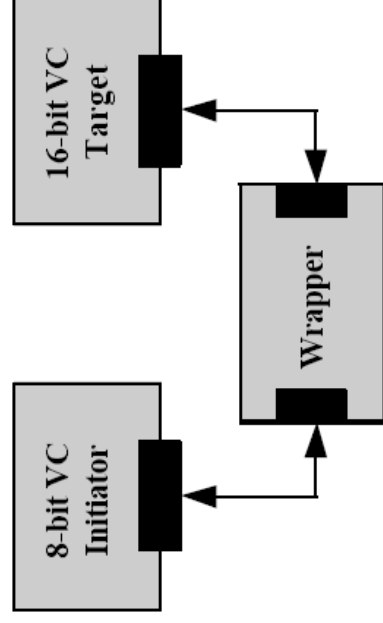


Figure 45: Interconnecting Different-Size VCI Components

Összegzés

- Az áramkörön belüli és áramkörön kívüli buszok jellemzői, követelményei eltérőek
- A komplex rendszerek összetett, nem feltétlenül egyetlen (akár hierarchikus) megosztott buszt kívánnak
- Nő a pont-pont kapcsolatok jelentősége
- Léteznek busz szabványok, illetve a buszok feletti, socket alapú interfész megoldások is