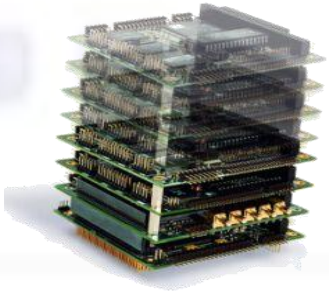
Embedded system hardware architecture

Embedded system hardware

System level

Board level



Component level

System level

- **Elements:**
  - Central Unit
  - Peripherals
- **Interconnection:**
  - Peripheral busses

Board level

Power supply

CPU

Peripherals

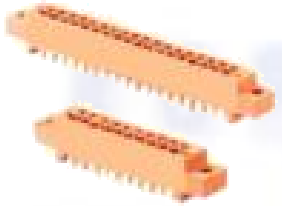
- **Elements:**
  - Boards
- **Interconnection:**
  - Internal busses

- *Communication*
- *Data Acquision*

Component level

Chips

Passive & active components



Connectors

- **Elements:**
  - Components
- **Interconnection:**
  - Wires

Embedded system architecture

CPU

North - bridge

Memory

PC-104 + (PCI)

PC-104 +

South - bridge

PC-104

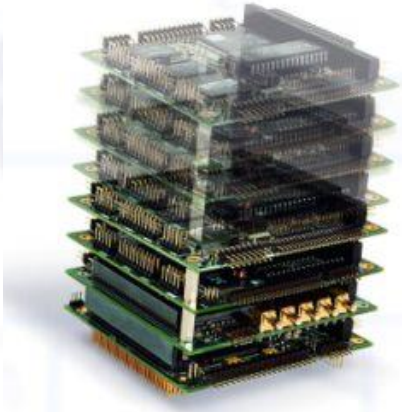
PC 104 (ISA)

Buses

- System Bus
- Peripheral bus
- Inter chip communication

Grouping of buses

- **System Bus**
  - ISA
  - PCI
  - PCI Express



- **Inter-chip communication**
  - I2C
  - JTAG

- **Low speed peripheral**
  - RS 232/422/485
  - CAN
- **High-speed peripheral**
  - USB
  - Firewire
  - Ethernet

- **System bus extension**
  - PCMCIA
  - Cardbus

Bus types

System Bus

- Parallel
  - ISA
  - PCI
- Serial
  - PCI-Express

System bus extension

- Parallel
  - IDE
  - PCMCIA (PCCARD)
  - Cardbus
- Serial
  - ExpressCard

Low speed peripheral

- Parallel
  - Parallel port
- Serial
  - RS 232/422/485
  - CAN

High-speed peripheral

- USB
- Firewire
- Ethernet

Inter-chip communication

- I2C, SPI, Microwire
- JTAG

What is a System Bus for?

- System Bus
- Peripheral bus
- Inter chip communication

A communication pathway connecting two or more cards

Data Bus

- Carries Data
- No difference between "Data" and "Instruction
- Width determinates performance

Address Bus

- Identify the source or destination of data
- Width determines maximum memory capacity of system

Control Bus

- Control and timing information
- Memory read/write signal
- Interrupt request
- Clock signals

Power lines

- Distributes power

Typical control line signals

- **Data transfer control**
  - Memory Write
  - Memory Read
  - I/O Write
  - I/O Read
- **Bus Arbitration**
  - Bus Request or DMA Request
  - Bus Grant or DMA Ack

- **Interrupt related**
  - Interrupt Request
  - Interrupt Ack
- **Other signals**
  - Clock
  - Reset

System Bus Types

- **Are the signals synchronized?**
  - **Yes** - Synchronous
  - **No** - Asynchronous
- **How many address data lines?**
  - **1** – Serial
  - **Multiply -** Parallel
- **Are the address data buses separated?**
  - **Yes** – Normal
  - **No -** Multiplexed

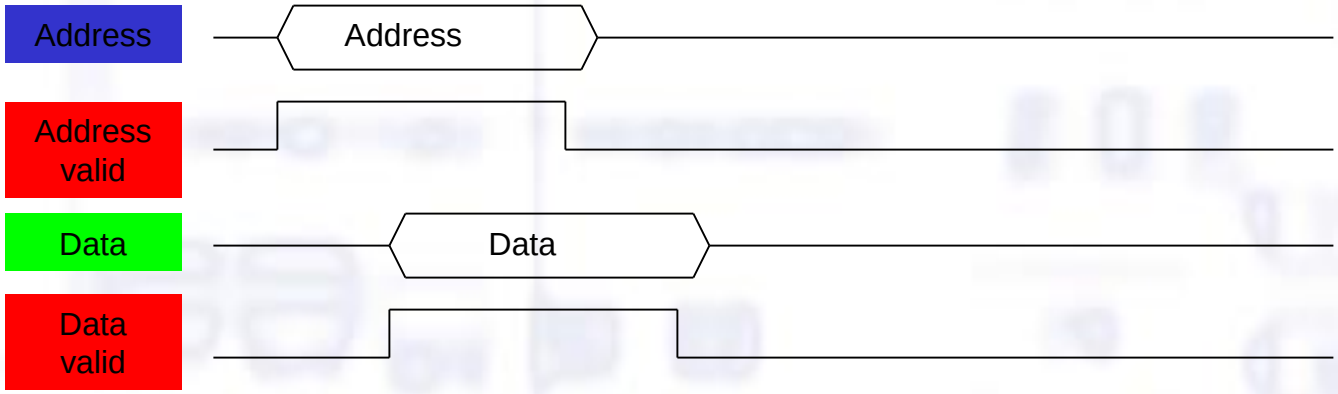
PC 104 (ISA) is a **Asynchronous, Parallel**

PC 104 Plus (PCI) is a **Synchronous, Parallel, Multiplexed**

System Bus Types I.

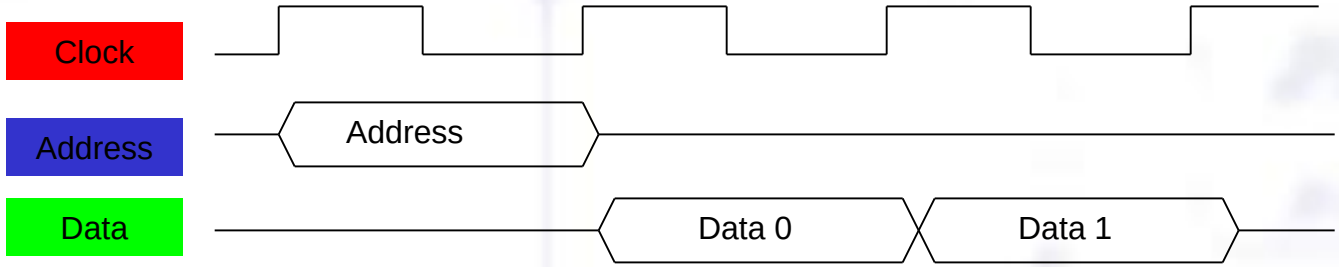
Parallel Asynchronous

ISA



Parallel Synchronous

PCI

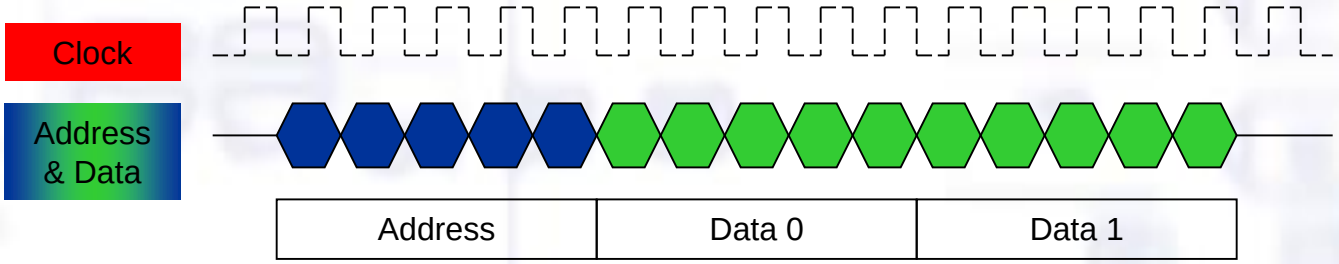


System Bus Types II.

Serial

Synchronous

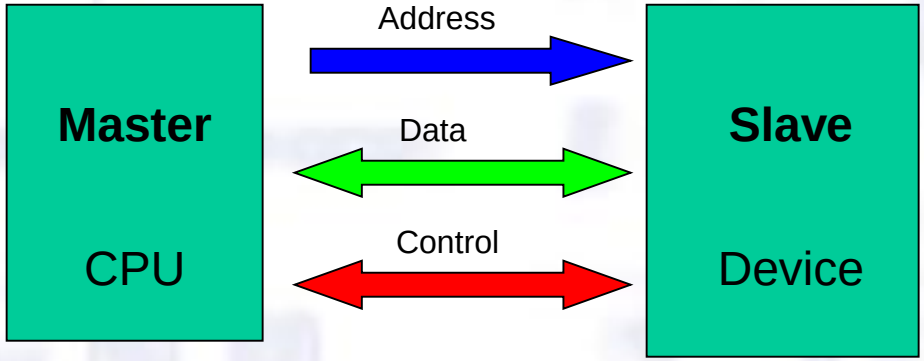
PCI-Express

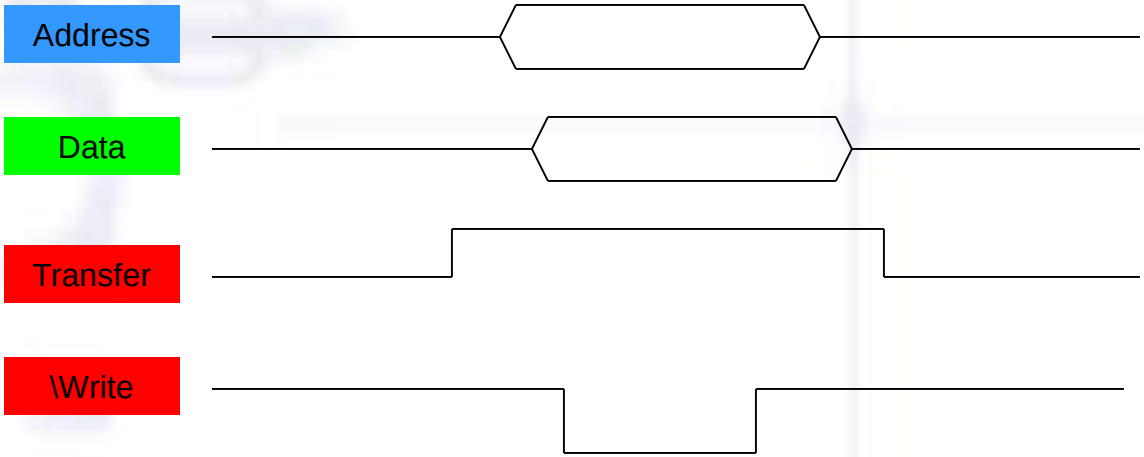


System Bus Types III.

Separate data address lines

ISA

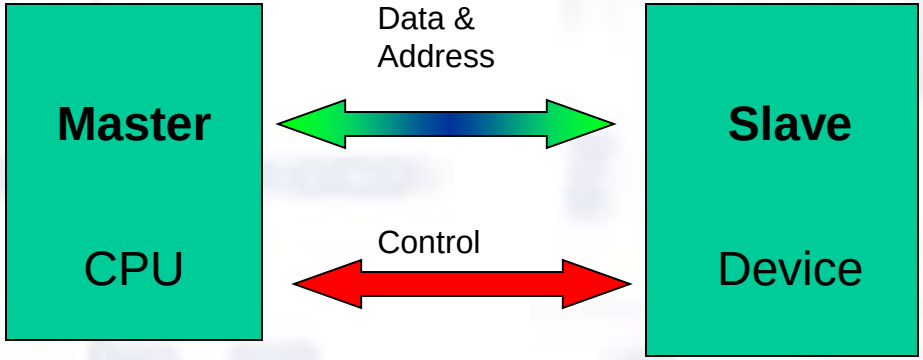


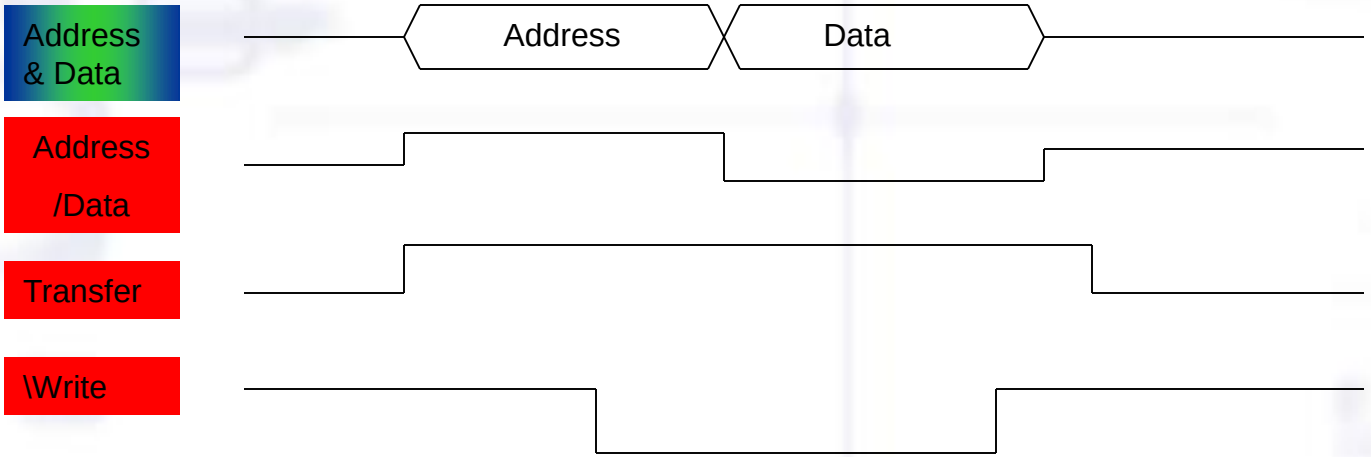


System Bus Types IV.

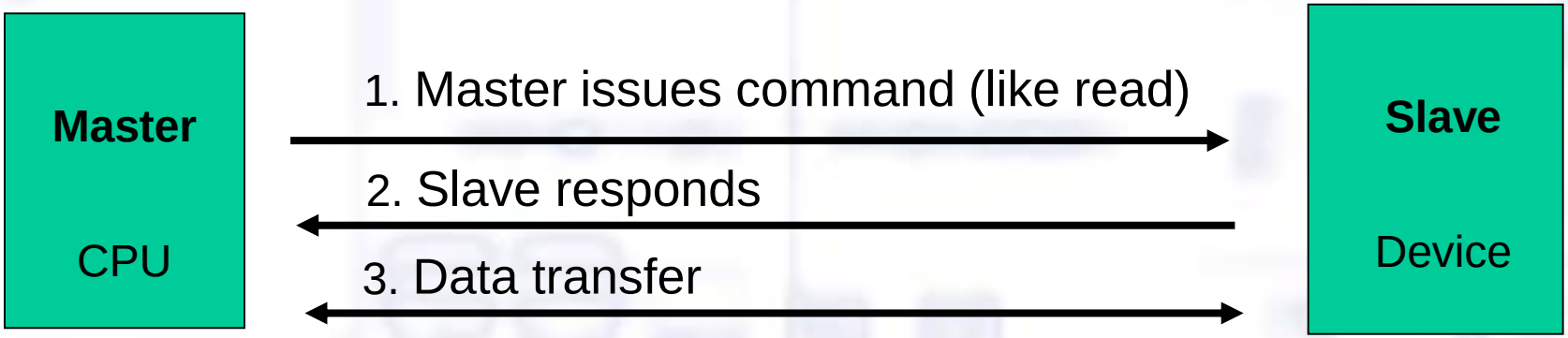
Multiplexed data & address lines

PCI





Bus Arbitration I. Master vs. Slave



- **Master** is the one who **starts** the bus transaction by:
  - issuing the command & address
- **Slave** is the one who **responds** to the address by:
  - *Sending* data to the master if the *master ask* for data
  - *Receiving* data from the master if the *master wants* to send data

Bus Arbitration I.
Who can be the master ?

- **How many potential bus masters?**
  - **1** – No arbitration needed
  - **More** – We need arbitration
- **Who makes the arbitration?**
  - 1 dedicated module (e.g.: CPU)
  - Distributed
- **Does any module have advantage?**
  - **Yes** – Fixed priority
  - **No** – Fairless - Round robin

PC 104 (ISA) is uses Fixed priority arbitration (DMA)

PC 104 Plus (PCI) uses fairless arbitration

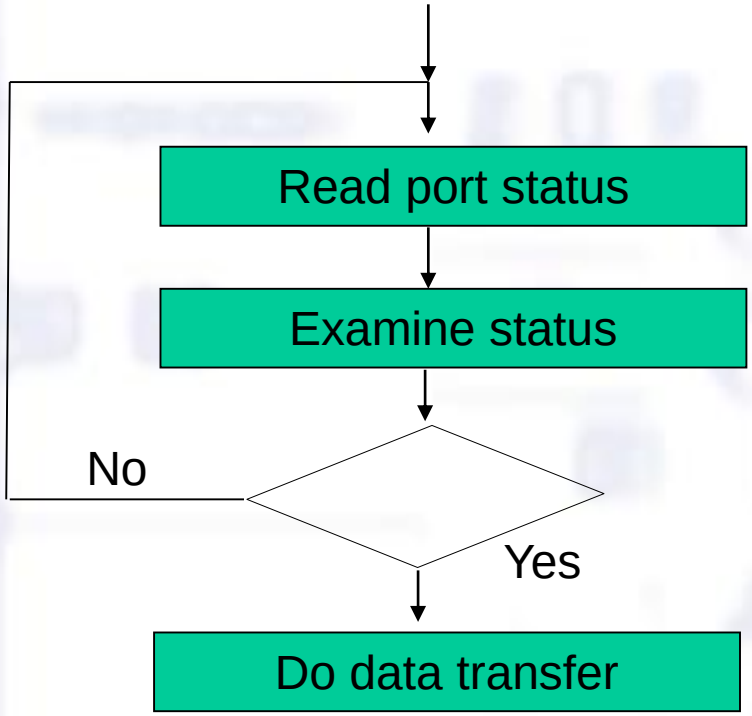
Techniques for I/O interfacing

	Data transfer type	Who drives the bus ?	CPU action	Signaling media
Polling	Simple	CPU	Normal	N/A
Interrupt	Quick	CPU	ISR	Dedicated
DMA	Multiple & quick	CPU / Device	Hold	Dedicated

Programmed I/O (Polling)

- Processor is in total control of the transfers

- Software Polling Loop

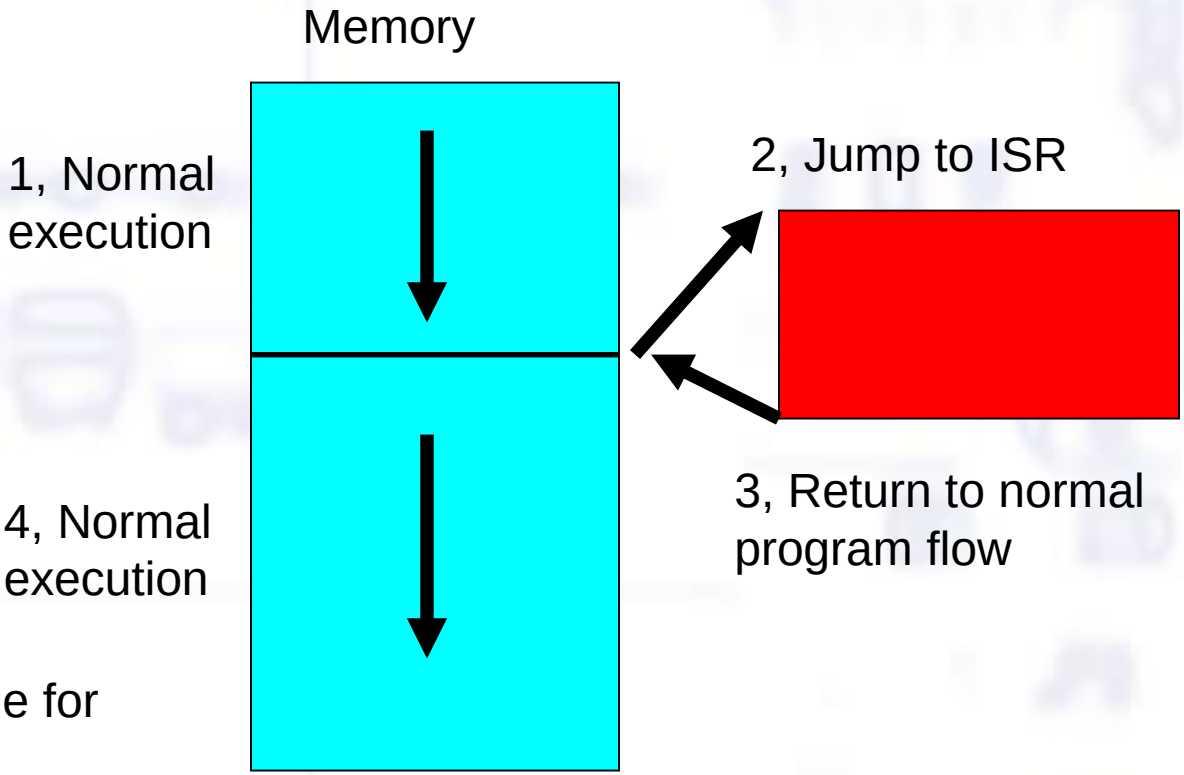


Problems:

- How can we determine when the Status will be read?
- Wasted Processor time while waiting for device to become Ready

Interrupt

- Device signals processor when transfer required
- Priority can be defined

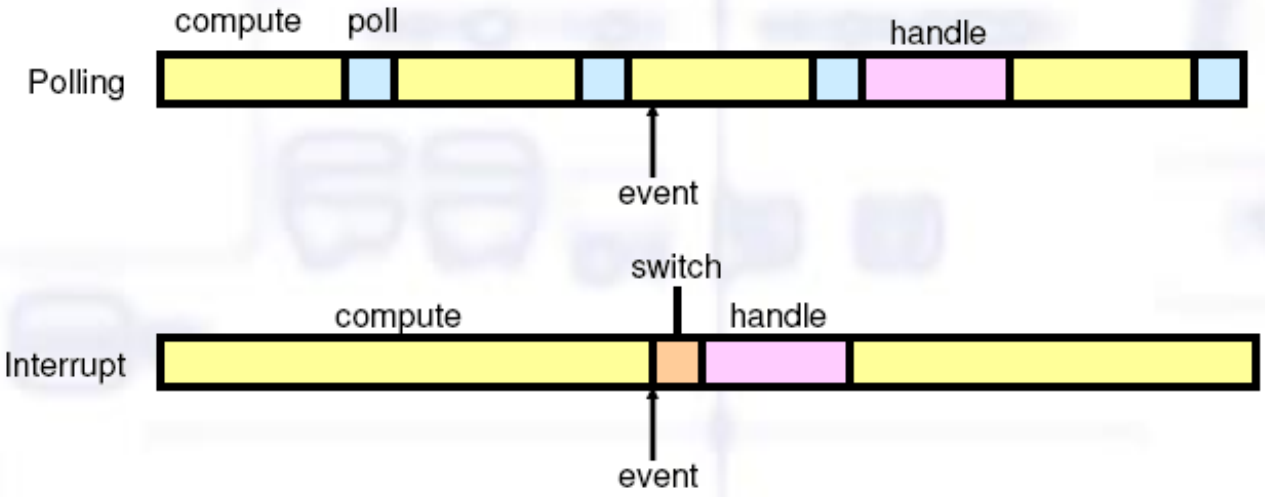


Problems:

- Dedicated Interrupt line for each device
- Wasted Processor time: Context switching

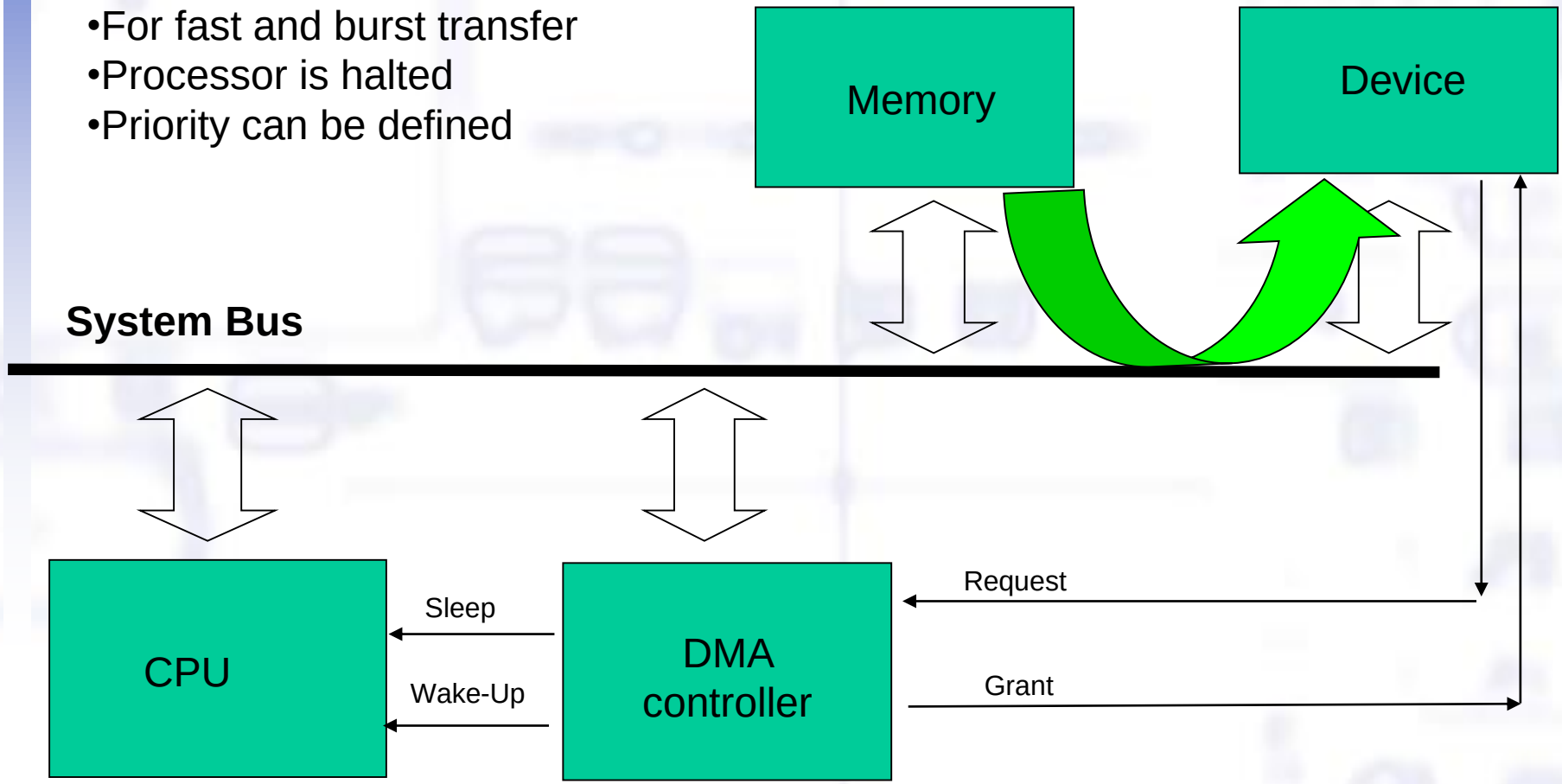
- System Bus
  - Background
- Peripheral bus
- Inter chip communication

Polling vs. Interrupt



Direct Memory Access

- For fast and burst transfer
- Processor is halted
- Priority can be defined

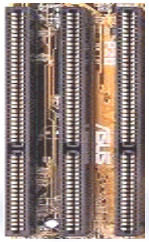


- System Bus
  - ISA Bus
- Peripheral bus
- Inter chip communication

Commonly used system buses in embedded systems

System Bus Owerview

- System Bus
- Peripheral bus
- Inter chip communication



System Bus	**ISA**	**PCI**
Embedded version	**PC/104**	**PC/104 *Plus***
System bus extension	**PCCARD**	**CARDBUS**

Comparing ISA and PCI

- System Bus
- Peripheral bus
- Inter chip communication

	ISA (PC/104)	PCI (PC/104+)
Bus width	8/16 bit	32 bit
Clock speed	8 Mhz	33 Mhz
Speed	8 MByte/s	132 Mbyte/s
Simplest interface	22 lines	54 lines
Complexity	Simple	Far more complicated
DMA and IRQ	Shared amond slots	Unique for each slot
Interrupt sharing	Not supported (P'n'p)	Automatic
Resource assignment	With jumpers	Automatic
Error detection	No	Yes
Voltage level	5V	5V / 3.3V

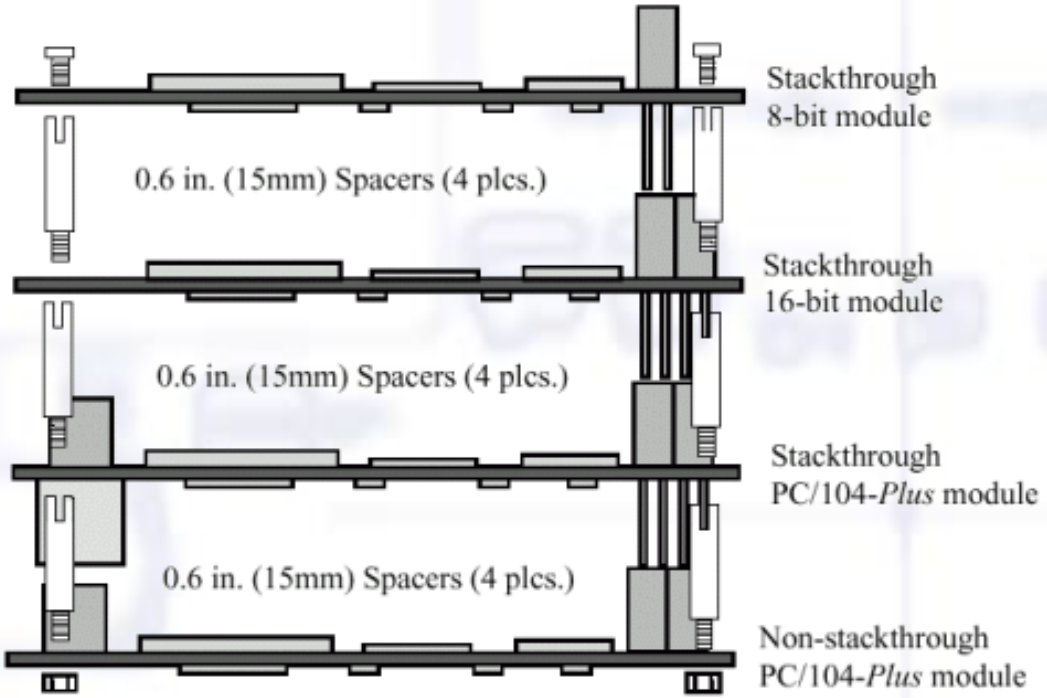
Comparing bus standards

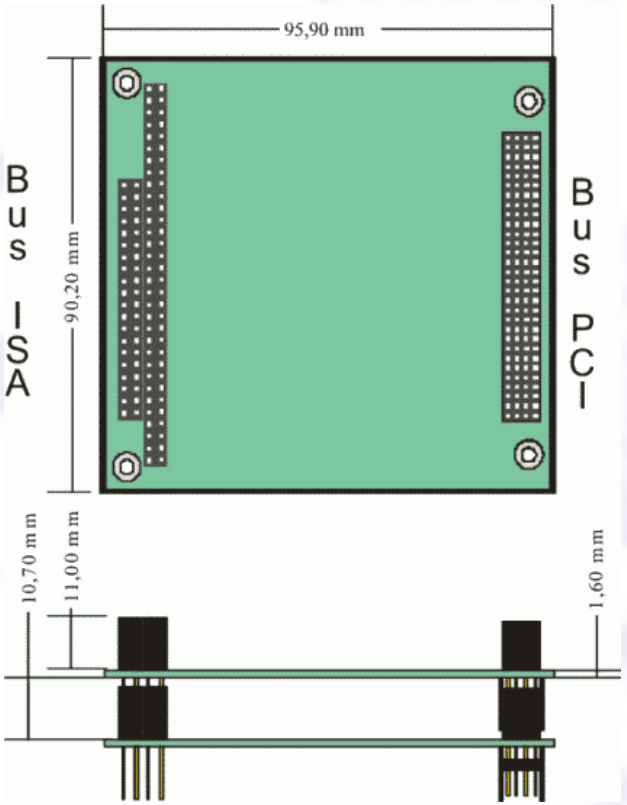
- System Bus
- Peripheral bus
- Inter chip communication

Bus	Year	With (bit)	Speed (Mhz)	Transfer (Mbyte/s)
PC/XT	1980	8	4.7/6	2.3 / 3
ISA	1984	16	8	8
EISA	1988	32	8	32
PCI	1993	**32**/64	**33**/66	**132** / 508
AGP	1997	32/64	66	264 / 4224
PCI-X	1998	64	Max. 133	1066

The PC/104 (*Plus*) standard

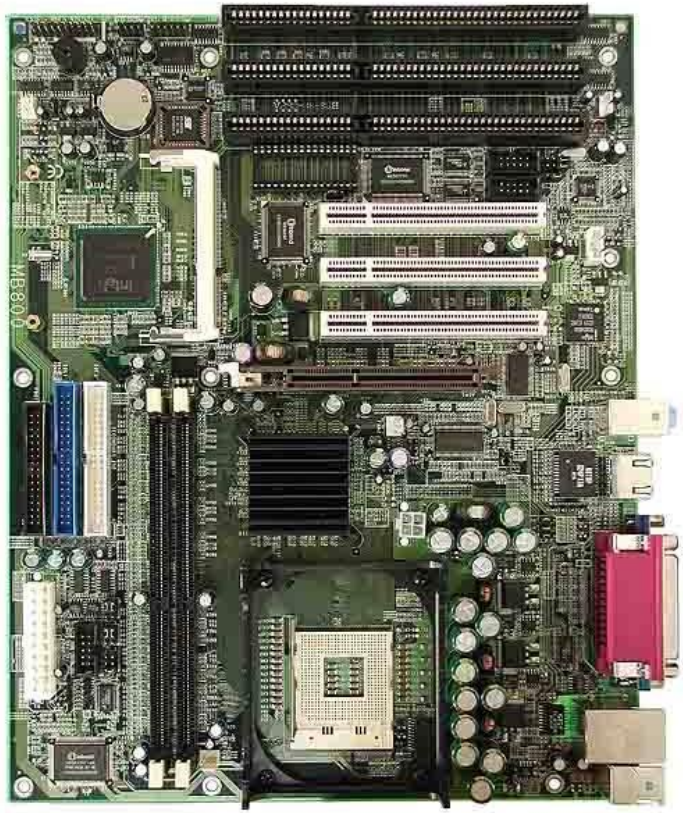
- **System Bus**
- **Peripheral bus**
- **Inter chip communication**





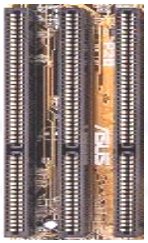
Comparing PC / 104 with standard PC

- **System Bus**
- **Peripheral bus**
- **Inter chip communication**



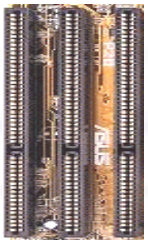
- System Bus
  - ISA Bus
- Peripheral bus
- Inter chip communication

The ISA Bus



ISA Address domain

- Memory Address Space
  - 24-bit address : FFFFFF : 16MB
  - 8/16-bit data
  - Access with: Memory Instructions
- I/O Address Space
  - 16-bit address space : FFFF: 64KB
  - 8/16-bit data
  - Access with: I/O instructions



ISA Signals I.

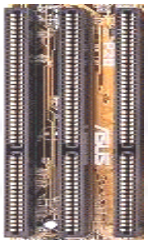
Address bus:

SA19 to SA0 – ***System Address***
Address memory and I/O devices 64K
With LA23 to LA17 16 Mbyte

LA23 to LA17 - ***Unlatched Address* bits 23:17**
With SA19 to SA0

AEN - ***Address Enable***
CPU Passive
DMA controller Active

BALE - ***Buffered Address Latch Enable***
Latch the LA23 to LA17 - falling edge of BALE.
With AEN, microprocessor or DMA address.



ISA Signals II.

Data bus:

SD15 to SD0 - ***System Data***

SD7 to SD0 8-bit transfer, SD15 to SD0 16-bit transfer.

Control bus:

CLK - ***System Clock***

Freerunning clock typ. 8MHz - 10MHz.

RESET DRV - ***Reset Drive***

High upon power up or subsequent / system reset.

-SMEMR, SMEMW ***System Memory Read / Write***

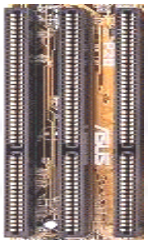
-MEMR, -MEMW **Memory Read / Write**

If address < 1 Mbyte

-IOR, -IOW *I/O Read* , *I/O Write*

-MEM CS16 , -I/O CS16 ***Memory Chip Select 16, I/O Chip Select 16***

Driven low by a I/O slave if capable 16-bit I/O data transfer.

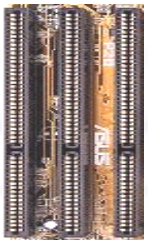


ISA bus cycles

- 8-Bit I/O Bus Cycles
- 16 Bit I/O Bus Cycles

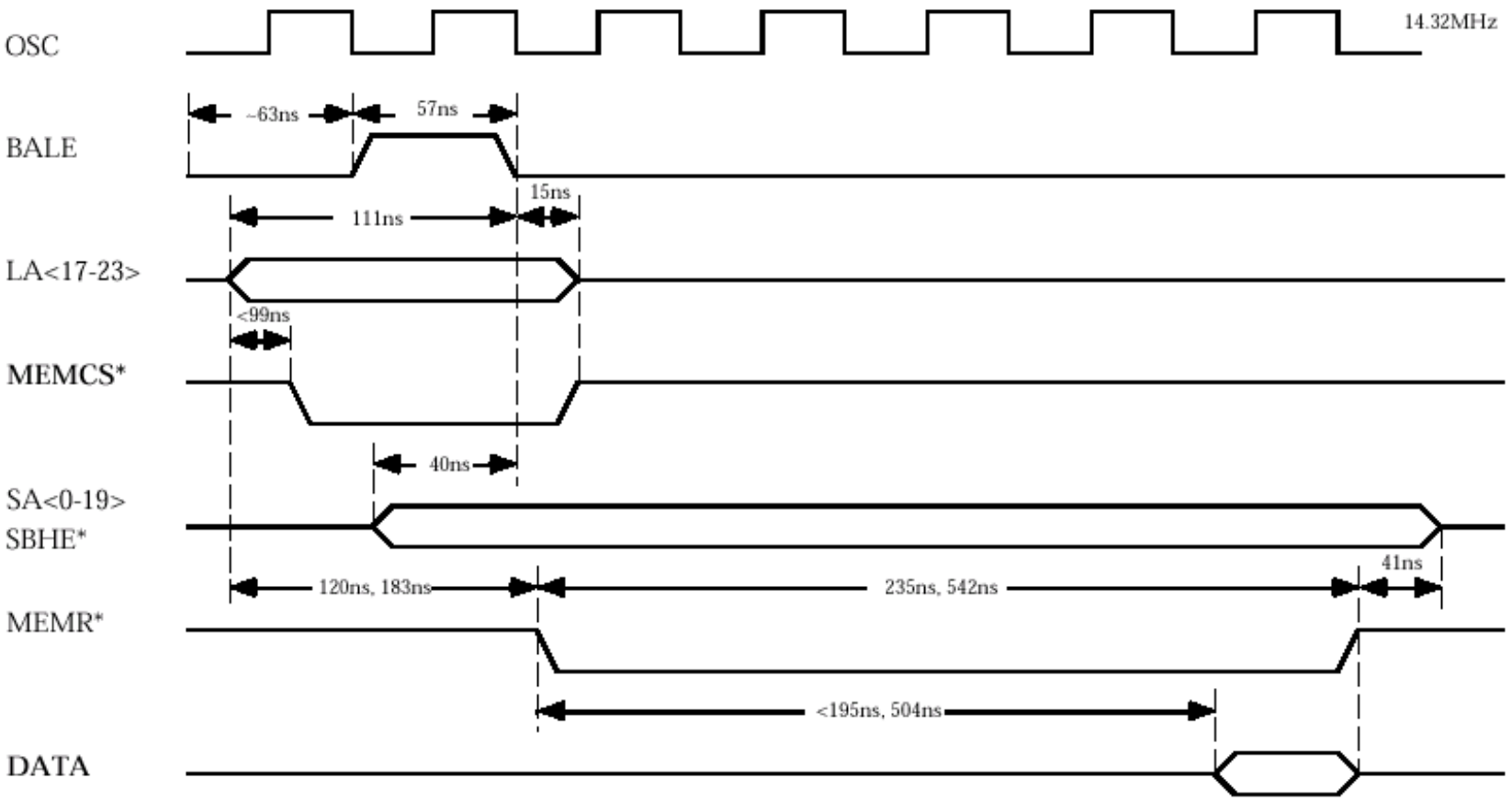
- 8-Bit Memory Bus Cycles
- 16-Bit Memory Bus Cycles (1 or more Wait States)
- 16-Bit Memory Bus Cycles (0 Wait States)
- Memory Refresh Cycles

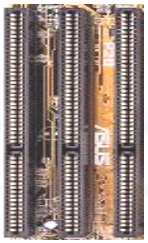
- DMA



ISA bus Timings : CPU MemRead

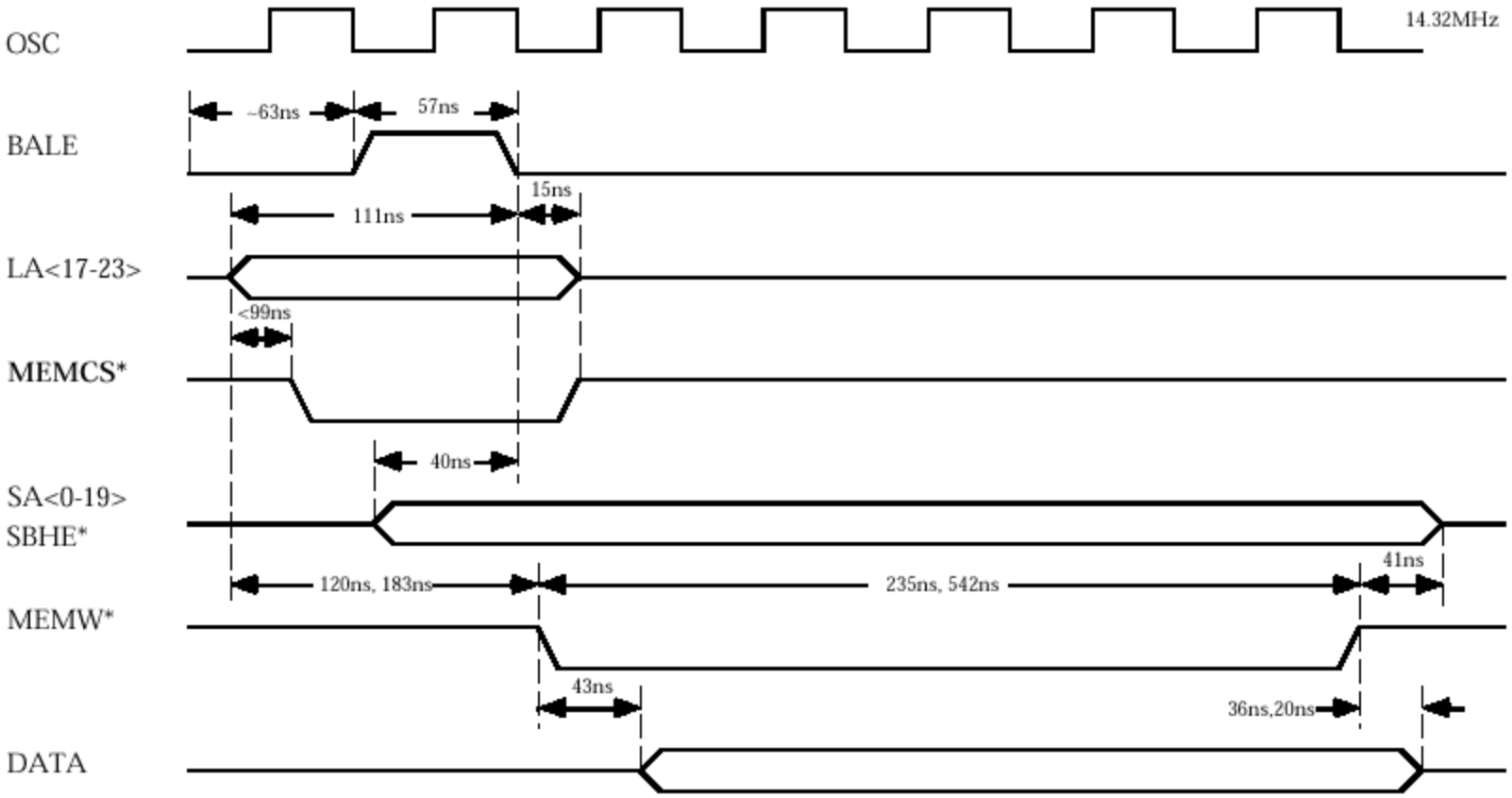
- System Bus
  - ISA Bus
- Peripheral bus
- Inter chip communication

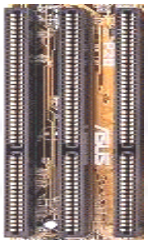




ISA bus Timings : CPU MemWrite

- System Bus
  - ISA Bus
- Peripheral bus
- Inter chip communication

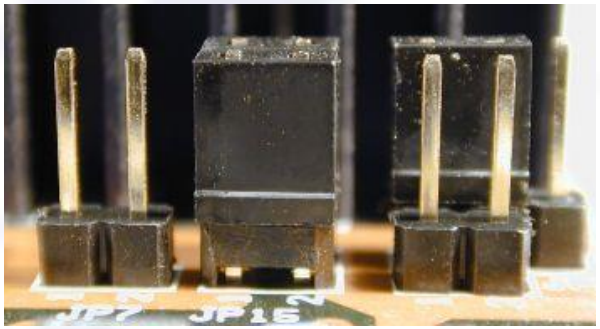


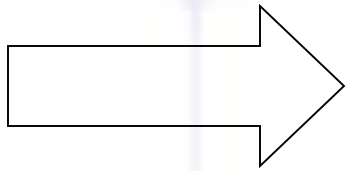


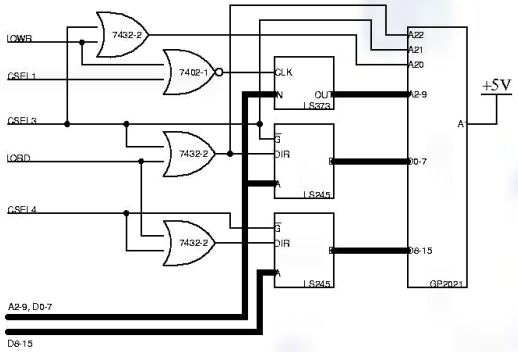
- System Bus
  - ISA Bus
- Peripheral bus
- Inter chip communication

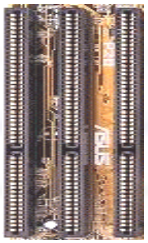
ISA Plug and Play (P'n'P)

- Settings can be the following
  - Interrupt requests (IRQ)
  - Direct memory access (DMA)
  - Memory addresses.
  - Input/Output (I/O) configuration





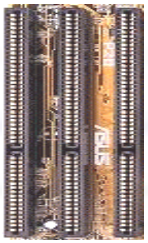




- System Bus
  - ISA Bus
- Peripheral bus
- Inter chip communication

I/O address map

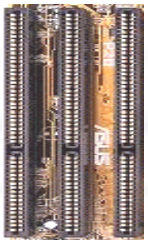
```
000-10F    DMA Controller, Interrupt Controller
           Real Time Clock, Keyoard, PS/2, Timer
080        POST code

170-17F    Hard Drive 1 (AT)
1F0-1FF    Hard Drive 0 (AT)
270-27F    Parallel Port 3
2E8-2EF    Serial Port COM4
2F8-2FF    Serial Port COM2
370-377    Floppy Disk
378-37F    Parallel Port 2
3BC-3BF    Parallel Port 1
3E0-3EF    Serial Port COM3
3F0-3F7    Floppy Disk 3F8-3FF
Serial Port COM1

110-16F    AVAILABLE
180-1EF    AVAILABLE
220-26F    AVAILABLE
280-2A1    AVAILABLE
320-32F    AVAILABLE
340-35F    AVAILABLE

210-217    Prototype Adapter
300-31F    Prototype Adapter
```



- System Bus
  - ISA Bus
- Peripheral bus
- Inter chip communication

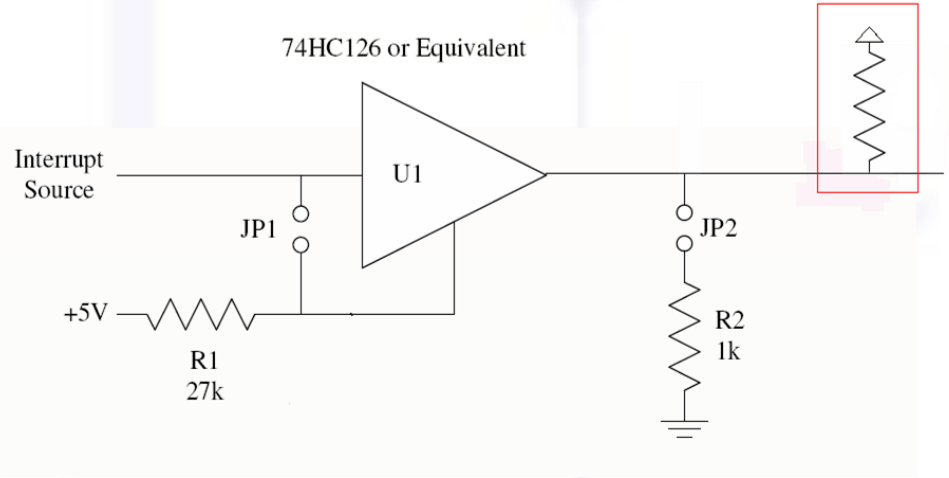
Interrupt map

NMI	Parity Error, Mem Refresh
IRQ0	8253 Channel 0 (System Timer)
IRQ1	Keyboard
IRQ2	Cascade from slave PIC
IRQ3	COM2
IRQ4	COM1
IRQ5	LPT2
IRQ6	Floppy Drive Controller
IRQ7	LPT1
IRQ8	Real Time Clock
IRQ9	Redirection to IRQ2
IRQ10	Reserved
IRQ11	Reserved
IRQ12	Mouse Interface (PS/2)
IRQ13	Coprocessor
IRQ14	Hard Drive Controller
IRQ15	Reserved

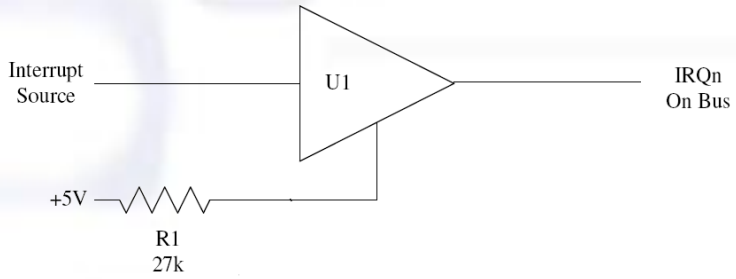


- System Bus
  - ISA Bus
- Peripheral bus
- Inter chip communication

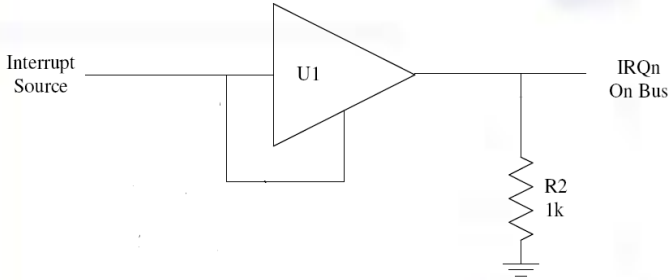
Interrupt sharing

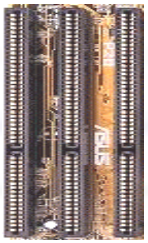


Without interrupt sharing



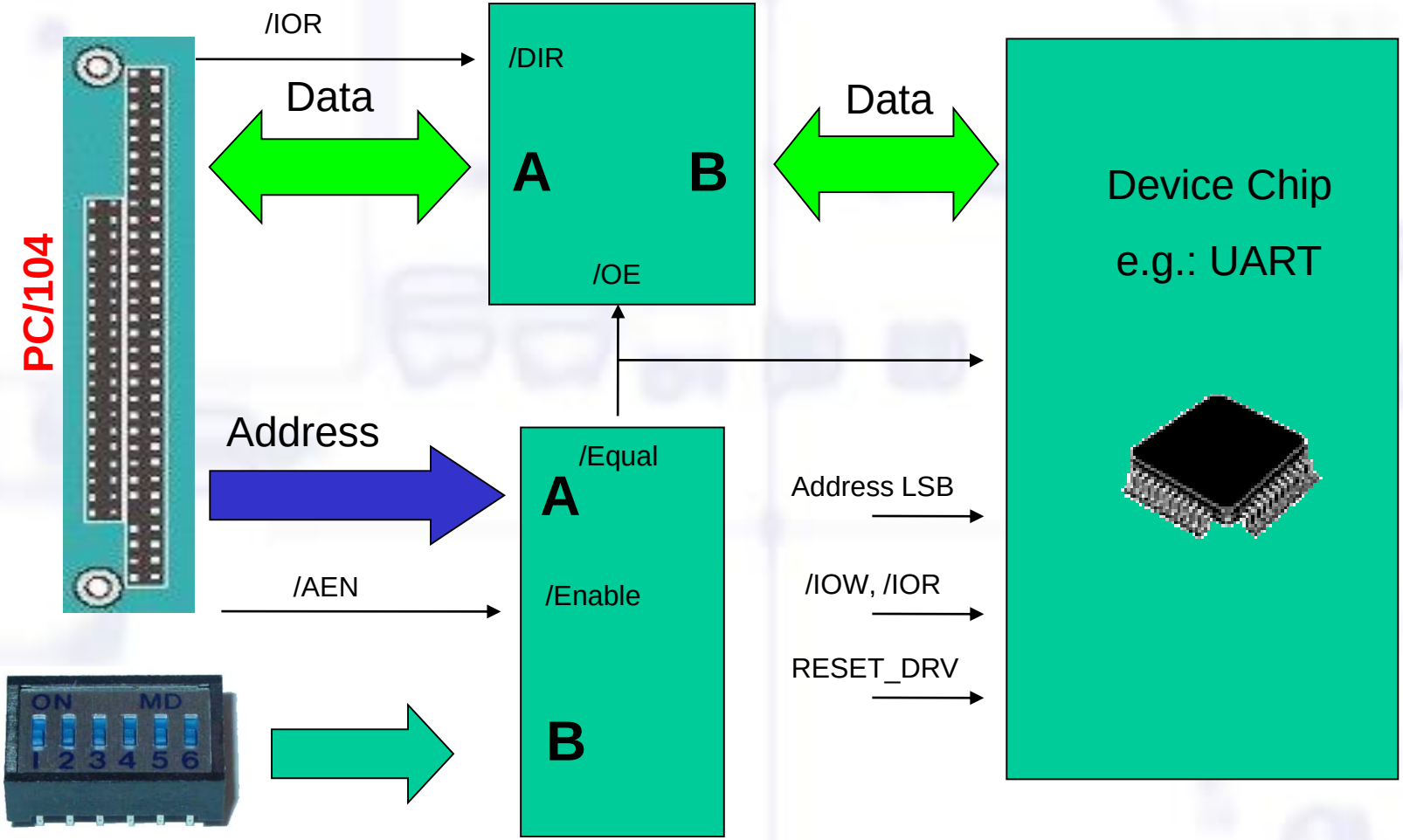
Interrupt sharing enabled

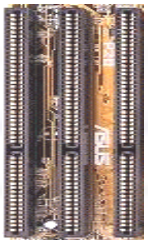




Case study: Serial port interface I.

- System Bus
  - ISA Bus
- Peripheral bus
- Inter chip communication





Case study: Serial port interface II.

The same in Verilog

```
module SGPS(sd, aenn, iown, iorn, sa, uartdata,
uartcs, jumpers);

        input [9:0] jumpers;
        input aenn, iown, iorn;
        input [9:0] sa;
        inout [7:0] uartdata;
        inout [7:0] sd;
        output uartcs;

assign uartcs   = ( ( sa== jumpers)& ~aenn);
assign sd = (uartcs & ~iorn) ? uartdata : 8'bz;
assign uartdata = (uartcs & ~iown) ? sd : 8'bz;

endmodule
```

The PCI Bus

PCI Overview

- System Bus
  - PCI Bus
- Peripheral bus
- Inter chip communication

CPU

North-bridge

Memory

PCI 0

PCI – PCI bridge

PCI 1

PCI devices

PCI – ISA bridge

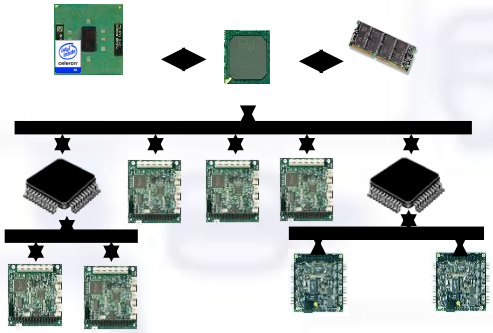
ISA

ISA devices

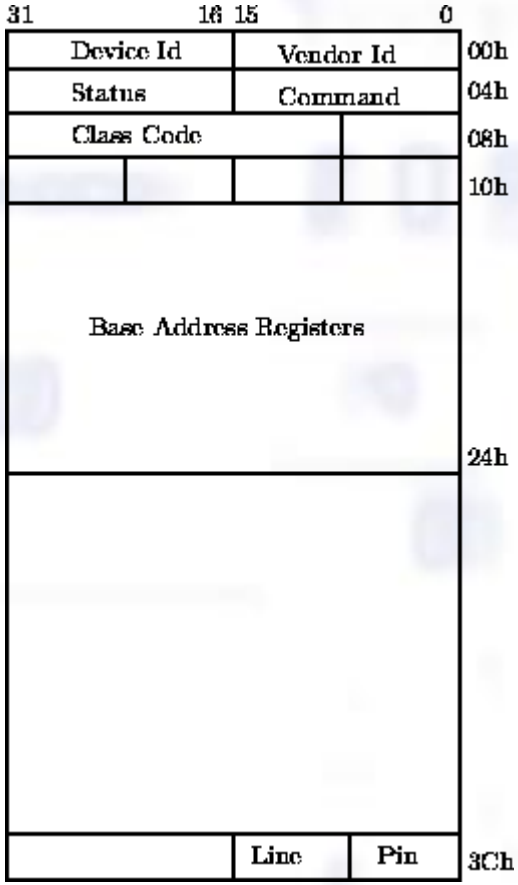
Configuration Address Space

- System Bus
  - PCI Bus
- Peripheral bus
- Inter chip communication

- PCI devices locations defined by
  - PCI bus number (hierarchy)
  - Device (slot) number



- Each device contains configuration registers
  - Card vendor and type
  - Memory and I/O Address
  - Interrupt line



PCI Address domain

- Memory Address Space
  - Supports 32-bit and 64-bit addresses
  - Host assigns section of main memory address space to each PCI device during
- I/O Address Space
  - 32-bit address space
  - Usage dependent on processor implementation
    - Special I/O instructions (x86)
    - Direct memory map

Required PCI signals

- Systems lines
  Including clock and reset
- Address & Data
  32 time mux lines for address/data
  Interrupt & validate lines
- Interface Control
- Arbitration
  Not shared
  Direct connection to PCI bus arbiter
- Error lines

Optional PCI signals

- Interrupt lines
  - Not shared
- Cache support
- 64-bit Bus Extension
  - Additional 32 lines
  - Time multiplexed
  - 2 lines to enable devices to agree to use 64-bit transfer
- JTAG/Boundary Scan
  - For testing procedures

PCI Operation sequence

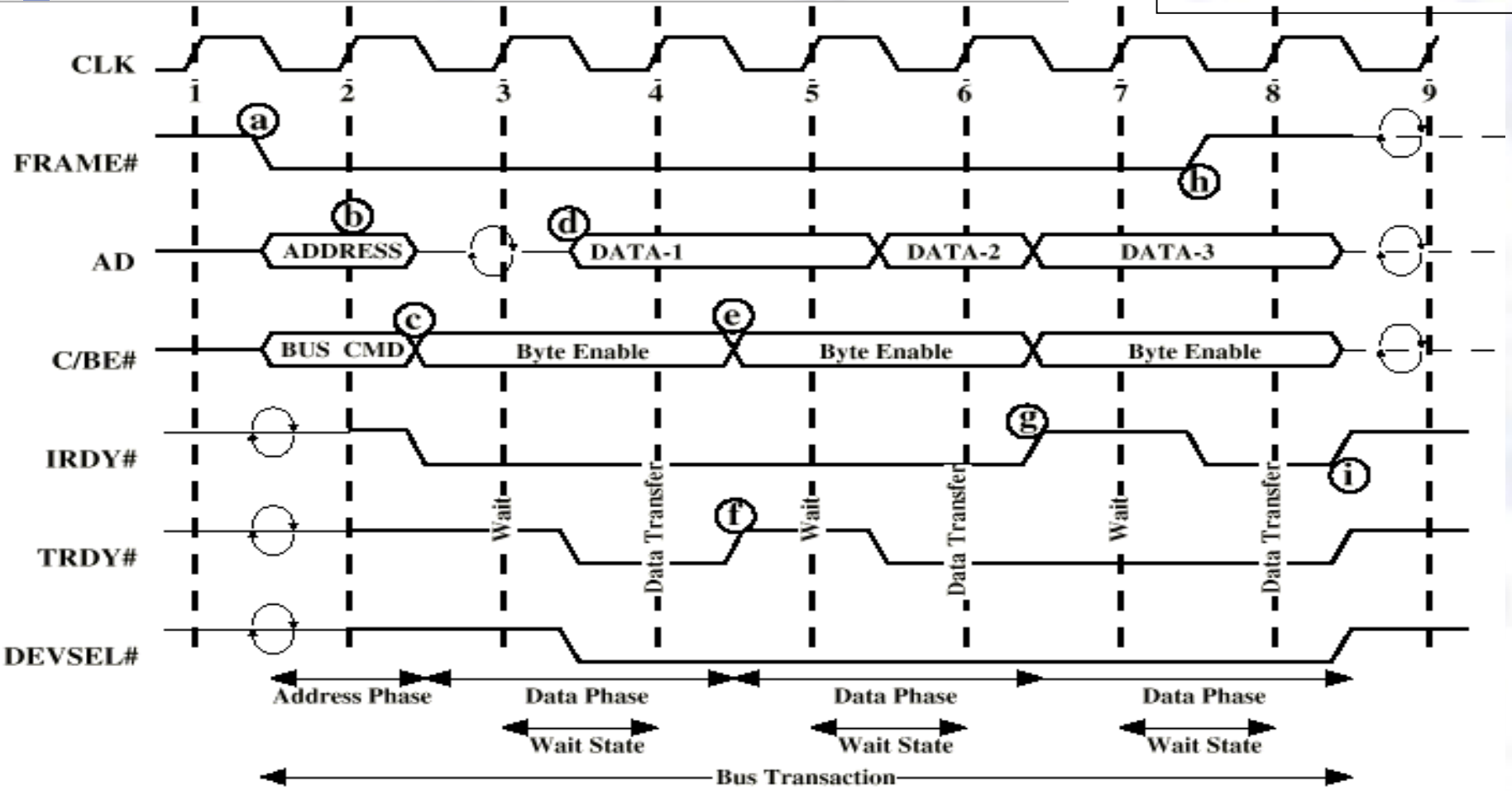
- Transaction between initiator (master) and target
- Master claims bus
- Determine type of transaction
  - e.g. I/O read/write
- Address phase
- One or more data phases

PCI Commands

- Interrupt Ack
- Special Cycle
- I/O Read and Write
- Memory Read and Write (many types)
- Configuration Read and Write
- Dual Address Cycle (i.e.: 64bit add.)

- System Bus
  - PCI Bus
- Peripheral bus
- Inter chip communication

PCI bus cycle

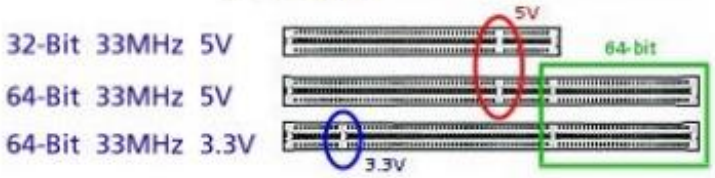


- System Bus
  - PCI Bus
- Peripheral bus
- Inter chip communication

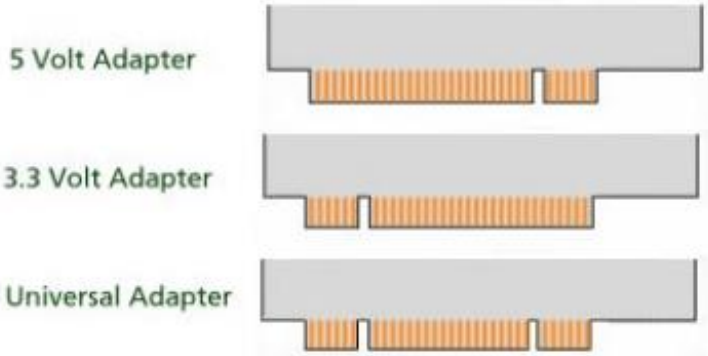
PCI Versions

Key Position

PCI Slots

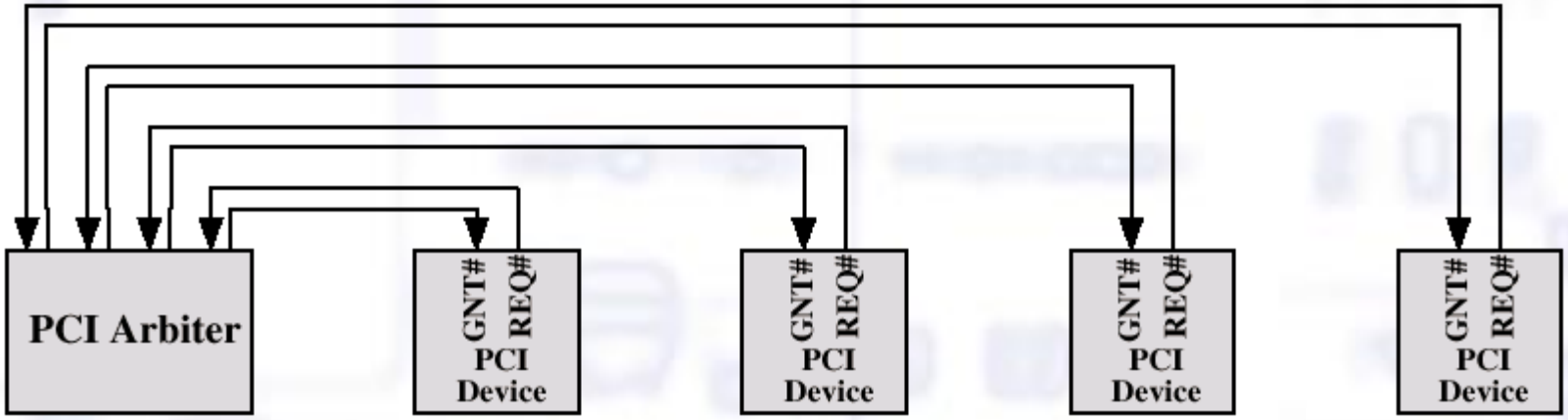


PCI Adapters



- System Bus
  - PCI Bus
- Peripheral bus
- Inter chip communication

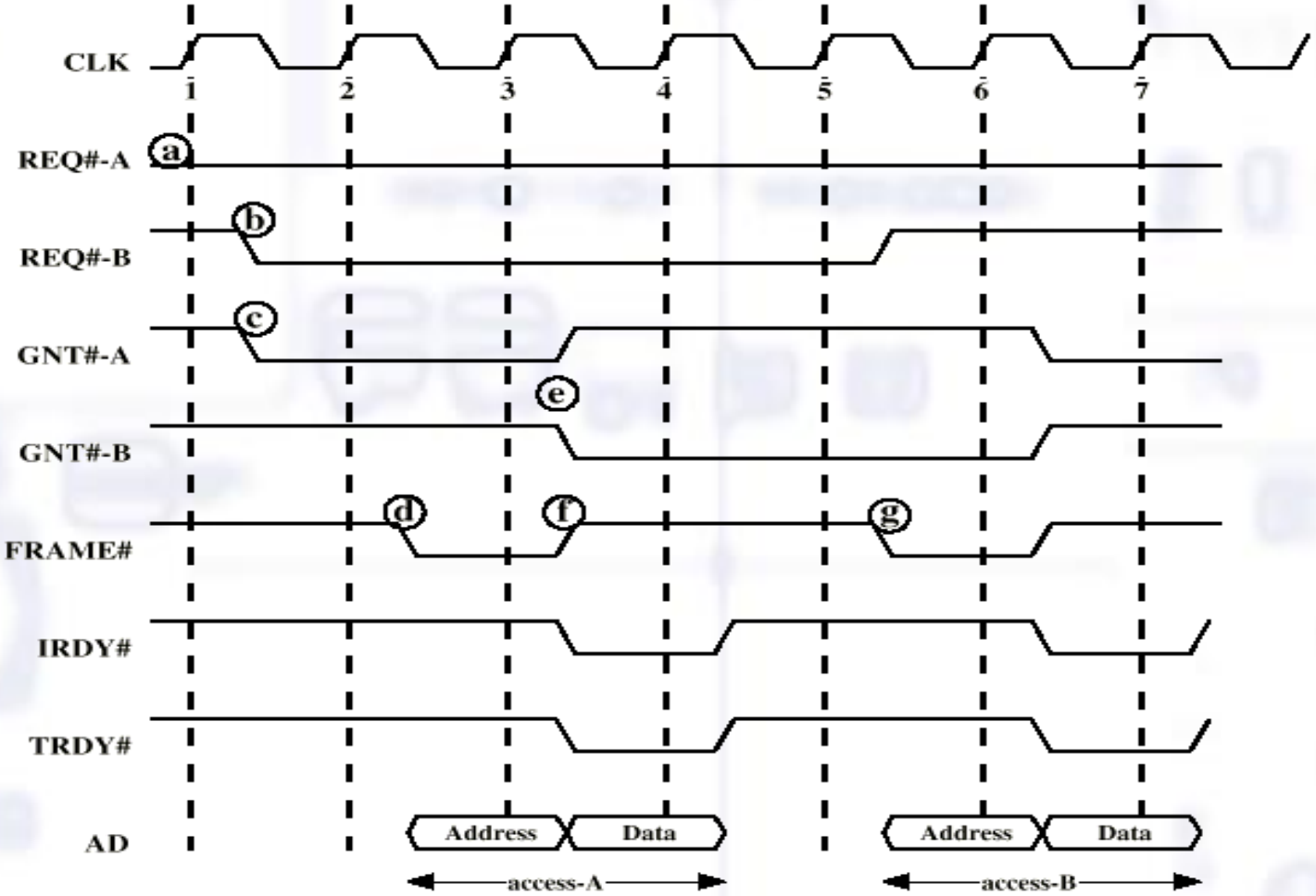
PCI Arbitartion



- Support for up to 6 masters, host and I/O bridge chip
- Round-robin scheduling
- Transaction timer

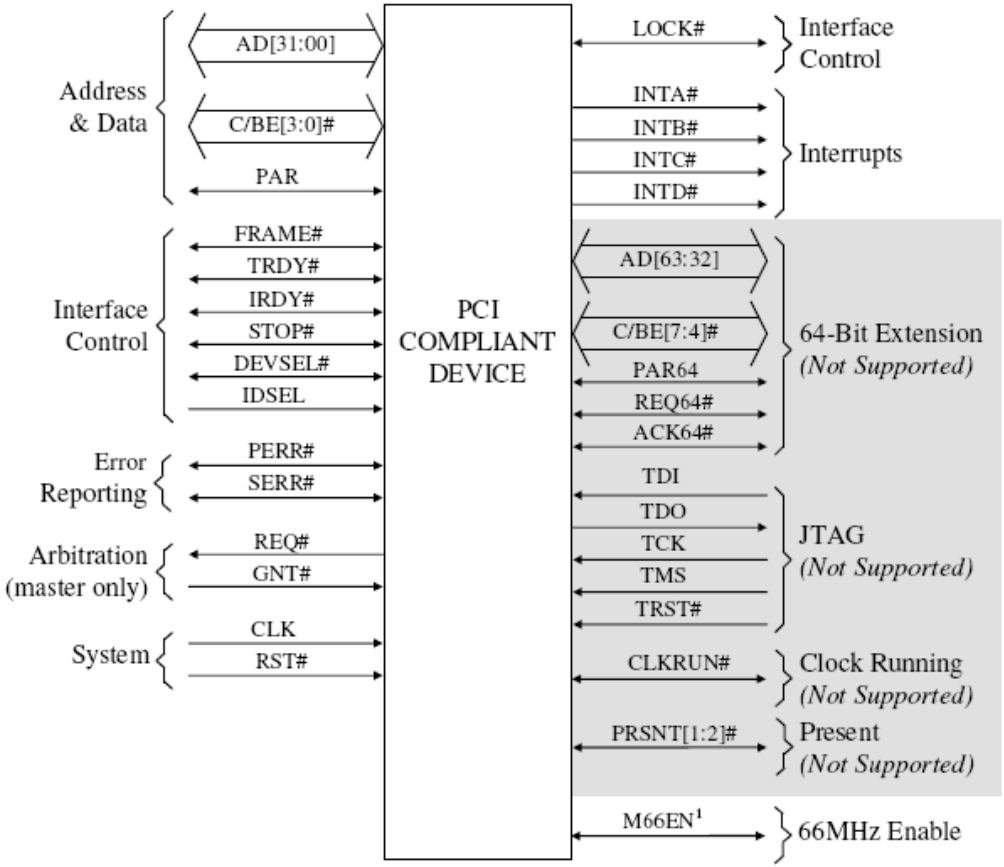
- System Bus
  - PCI Bus
- Peripheral bus
- Inter chip communication

PCI Arbitartion



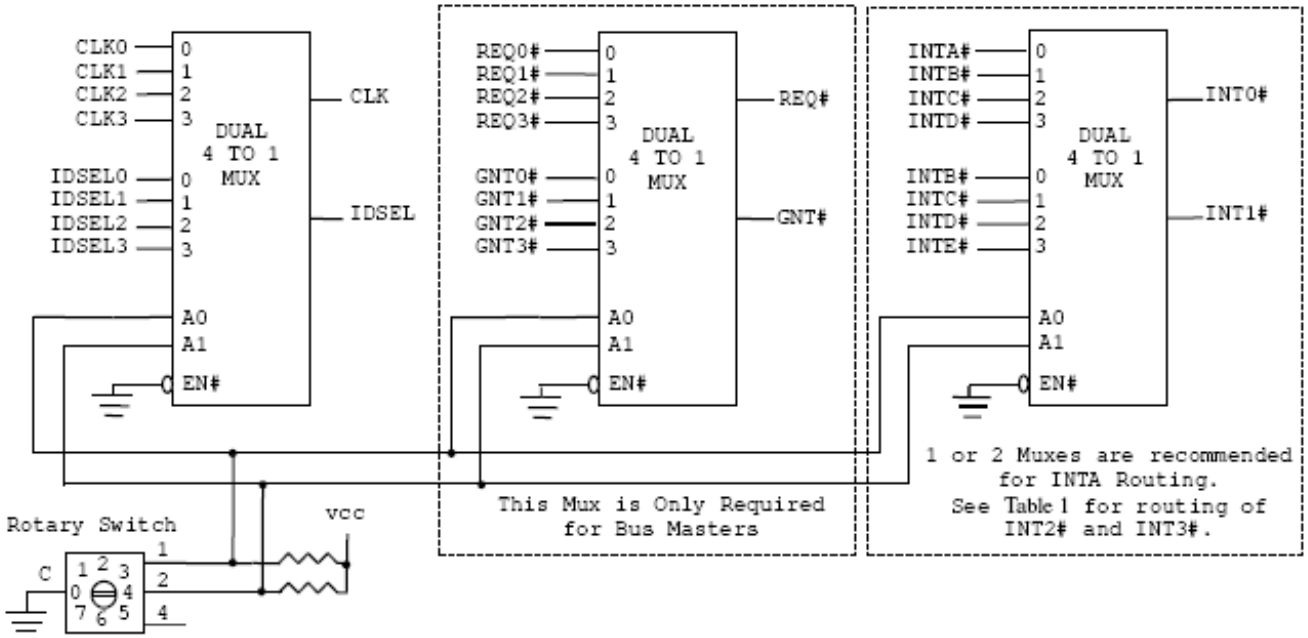
- System Bus
  - PC 104 Plus
- Peripheral bus
- Inter chip communication

PC/104 Plus & PCI pinout



PC/104 Device Selection

- System Bus
  - PCI Bus
- Peripheral bus
- Inter chip communication



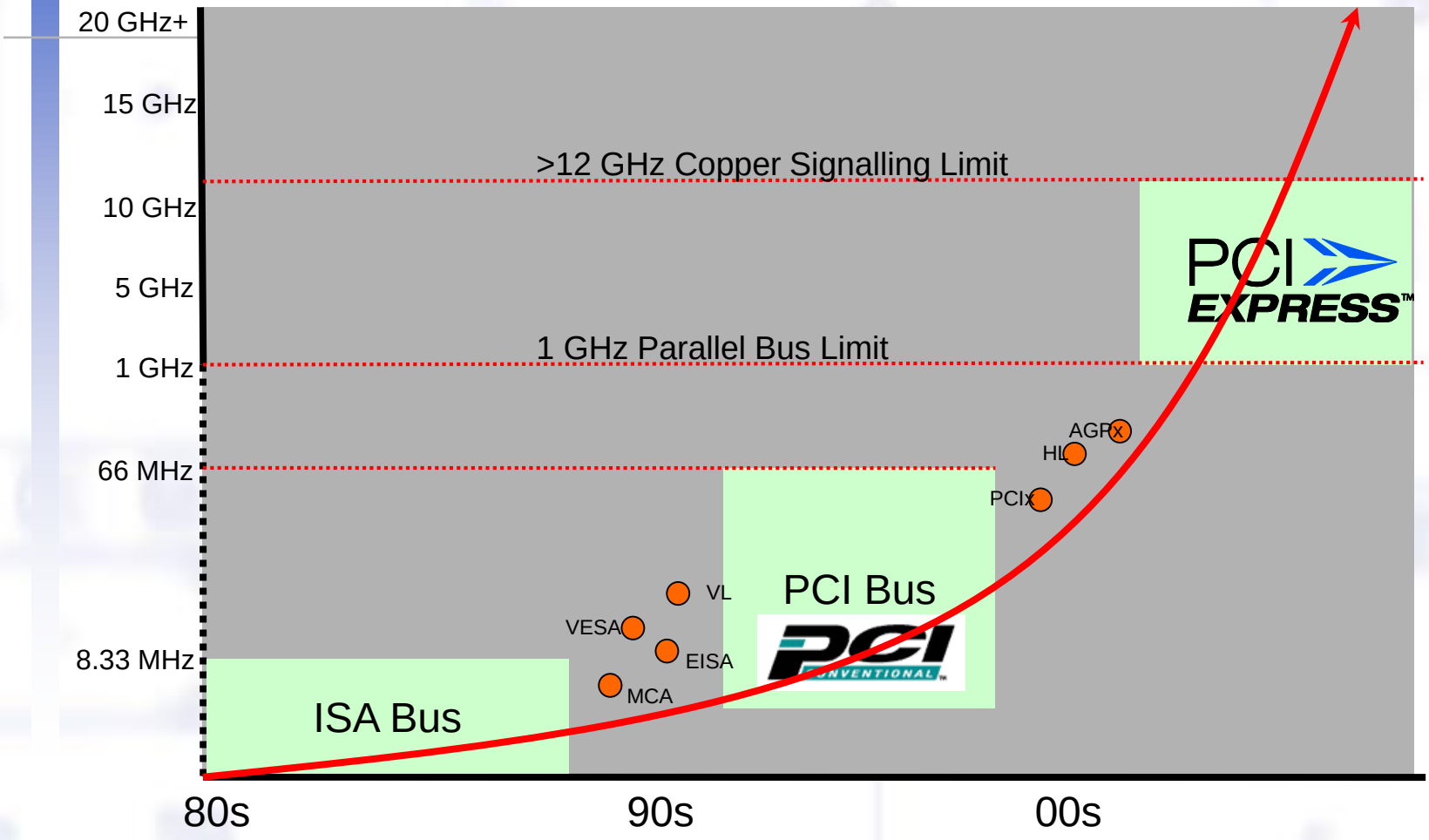
Switch Position	Module Slot	REQ#	GNT#	CLK	INT0#	INT1#	INT2#	INT3#
0 or 4	1	REQ0#	GNT0#	CLK0	INTA#	INTB#	INTC#	INTD#
1 or 5	2	REQ1#	GNT1#	CLK1	INTB#	INTC#	INTD#	INTA#
2 or 6	3	REQ2#	GNT2#	CLK2	INTC#	INTD#	INTA#	INTB#
3 or 7	4	REQ3#	GNT3#	CLK3	INTD#	INTA#	INTB#	INTC#

Overview of PCI Express

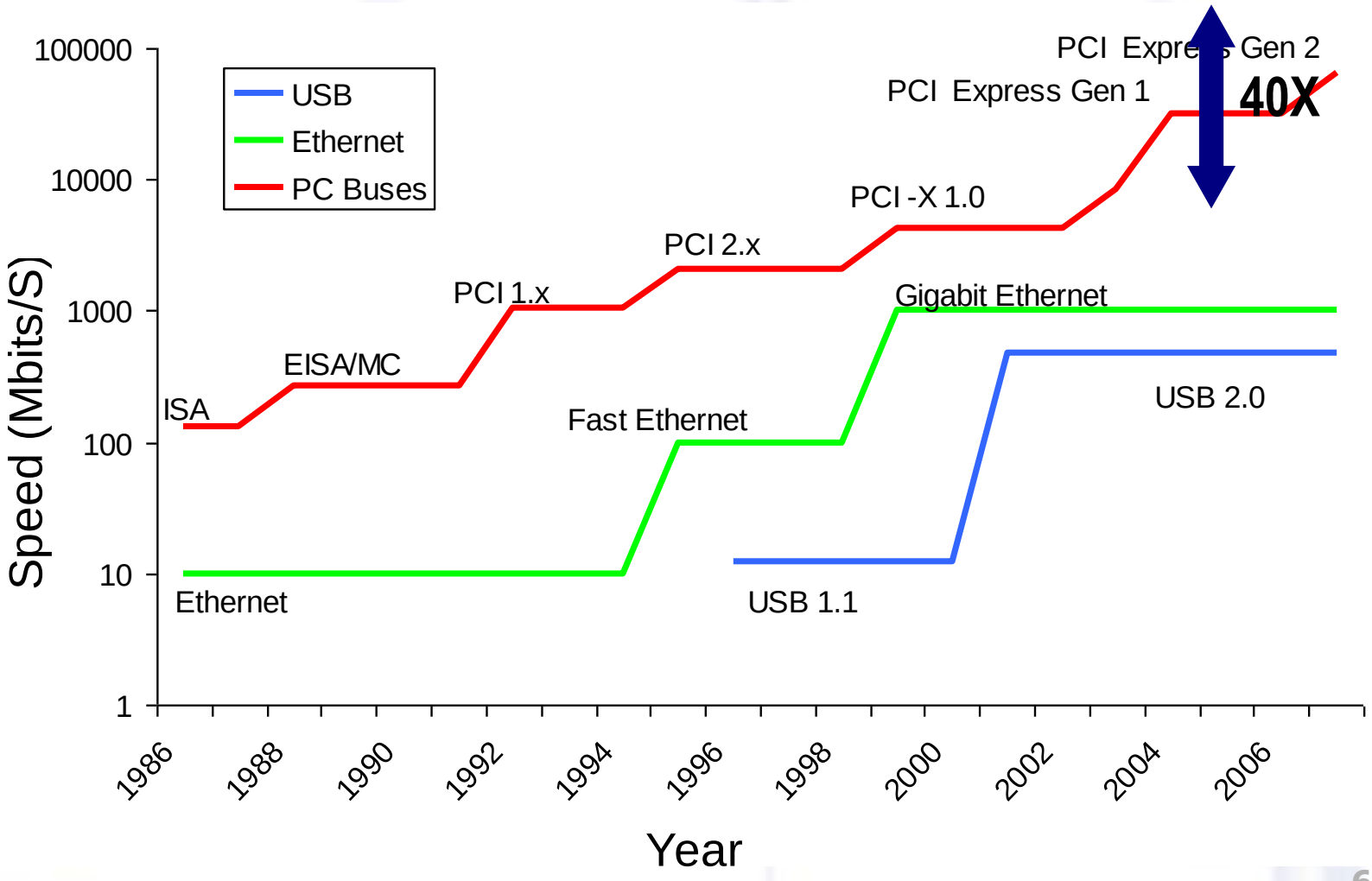
PCI Challenges

- Limited Bandwidth
- Bandwidth shared between all devices
- Limited host pin-count
- Lack of support for real time data transfer
- Stringent routing rules
- Lack of scaling with frequency and voltage
- Absence of power management

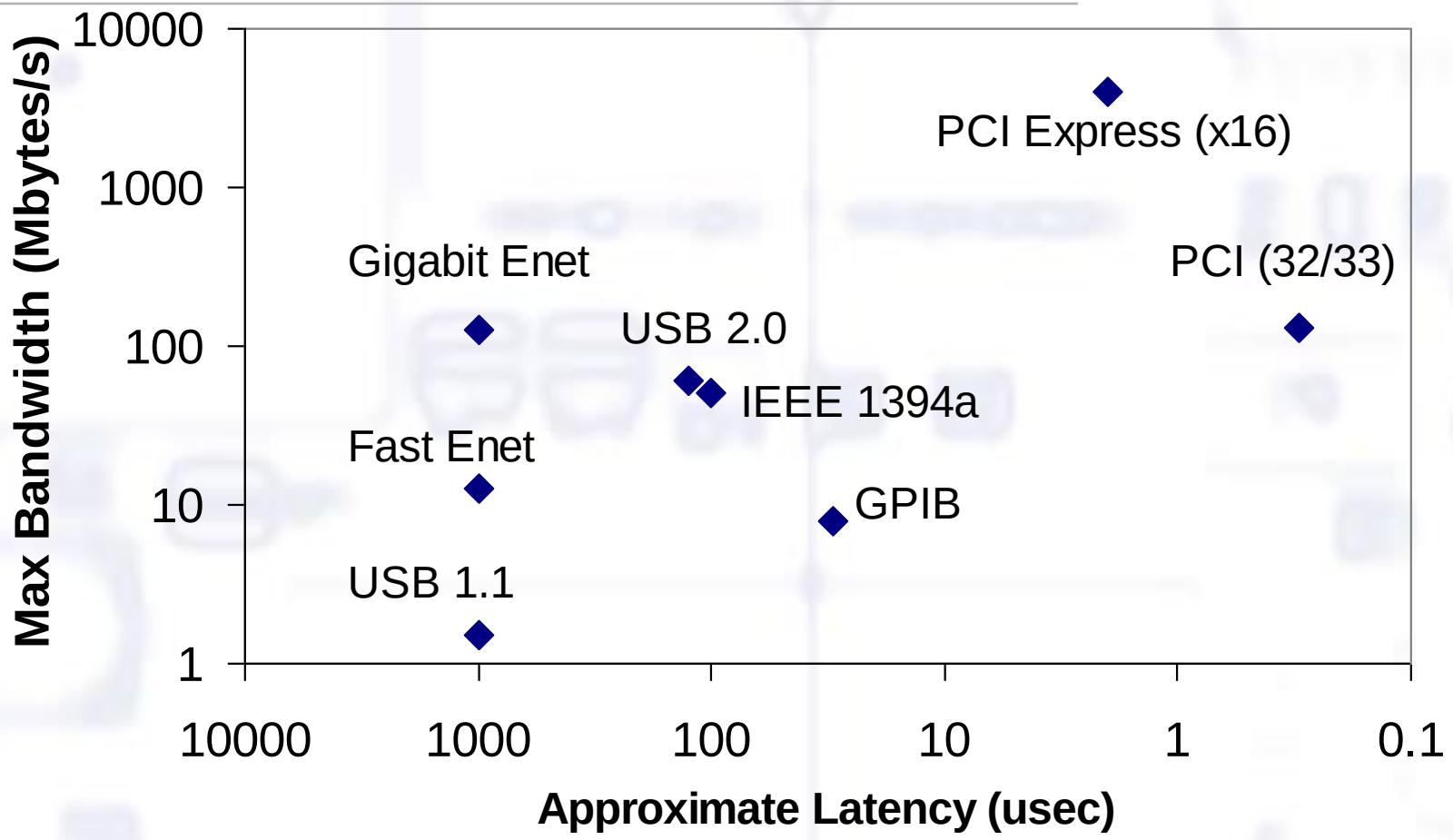
Evolution of PC Buses



Evolution of Industry Standard Bus Throughput



Industry Bus Performance

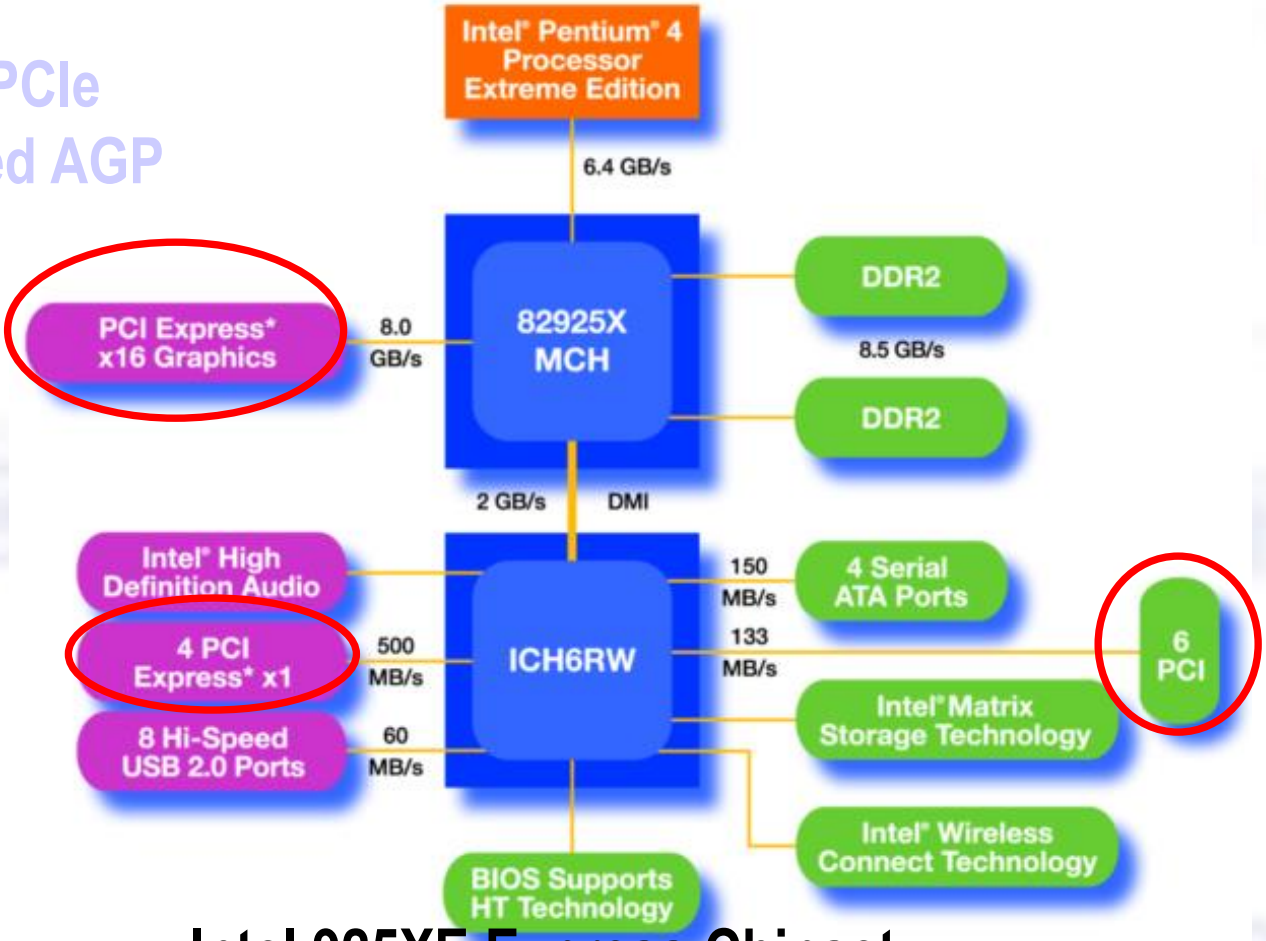


PCI Express: Evolutionary Model of PCI

- Serial interconnect @ 2.5Gb/S
  - PCI transactions are packetized and then serialized
  - LVDS Signalling, point-to-point, 8B/10B encoded
  - Bandwidth is available PER slot, and in BOTH directions
  - X1 gives real-world performance of 200MByts/S/direction
  - X16 gives real-world performance of 3.2GBytes/S/direction

PCI Express in Desktops

x16 PCIe
Replaced AGP



Four
x1 PCIe

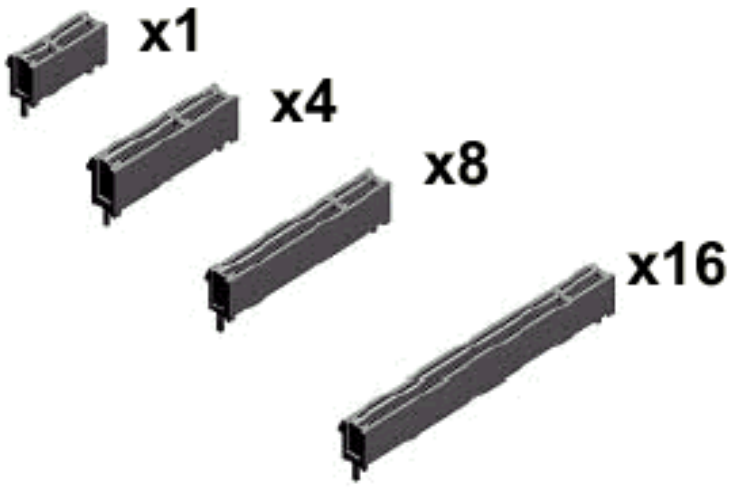
Six
PCI

Intel 925XE Express Chipset

Benefits of PCI Express

Improved Bandwidth

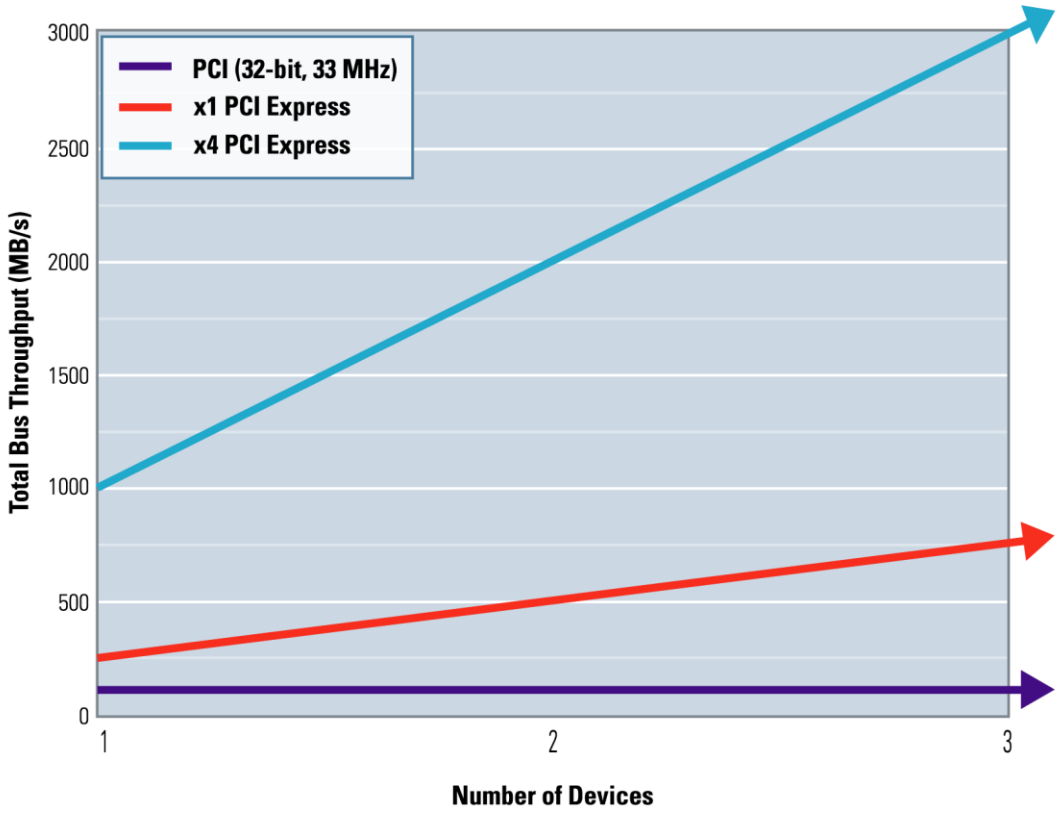
- Up to 30X bandwidth increase over PCI
- Scalable bandwidth by increasing lane width



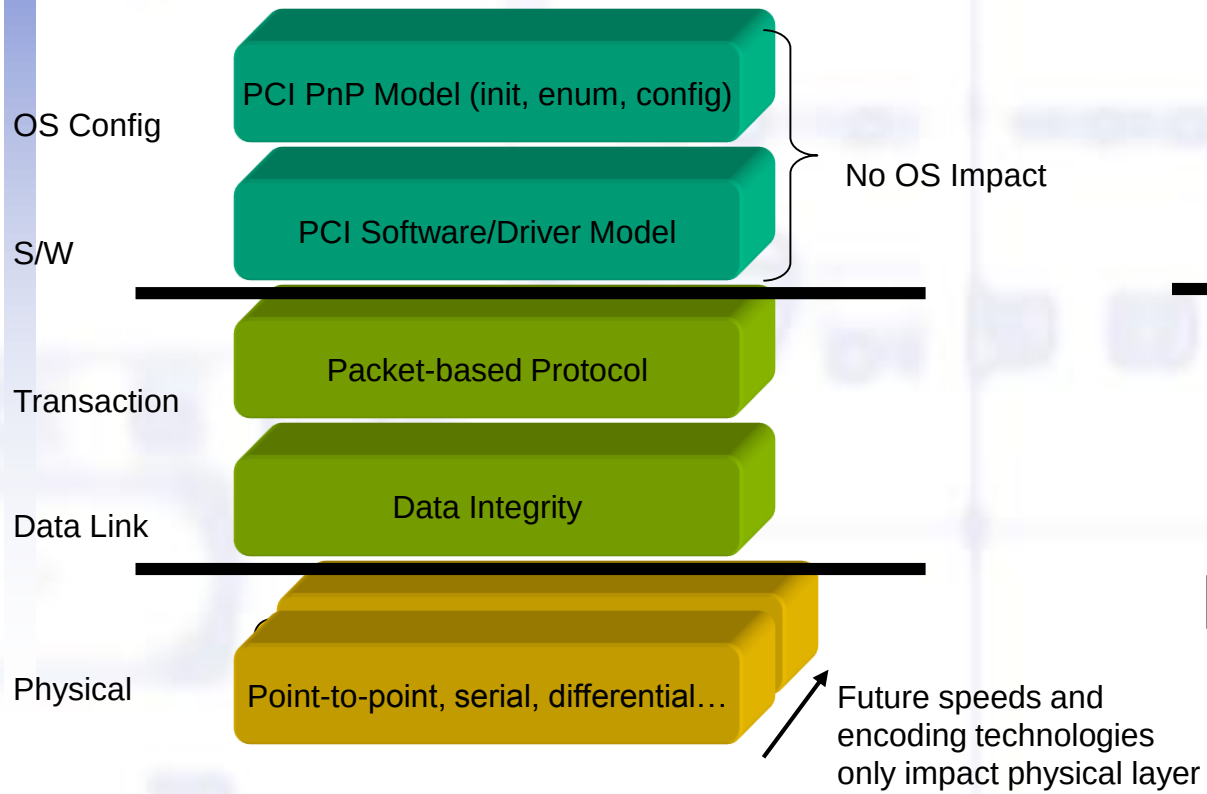
Bus	Bandwidth (MBytes/s)
PCI (32-bit, 33 MHz)	132
x1 PCI Express	250
x4 PCI Express	1000
x16 PCI Express	4000

Dedicated Bandwidth

- PCI Express dedicates bandwidth per device



Software Backward Compatibility



PCI and PCI Express use same software model

New hardware layers deliver 30X performance of PCI

Card Interoperability

Slot / Card	x1	x4	x8	x16
x1	Required	Required	Required	Required
x4	No	Required	Allowed	Allowed
x8	No	No	Required	Allowed
x16	No	No	No	Required

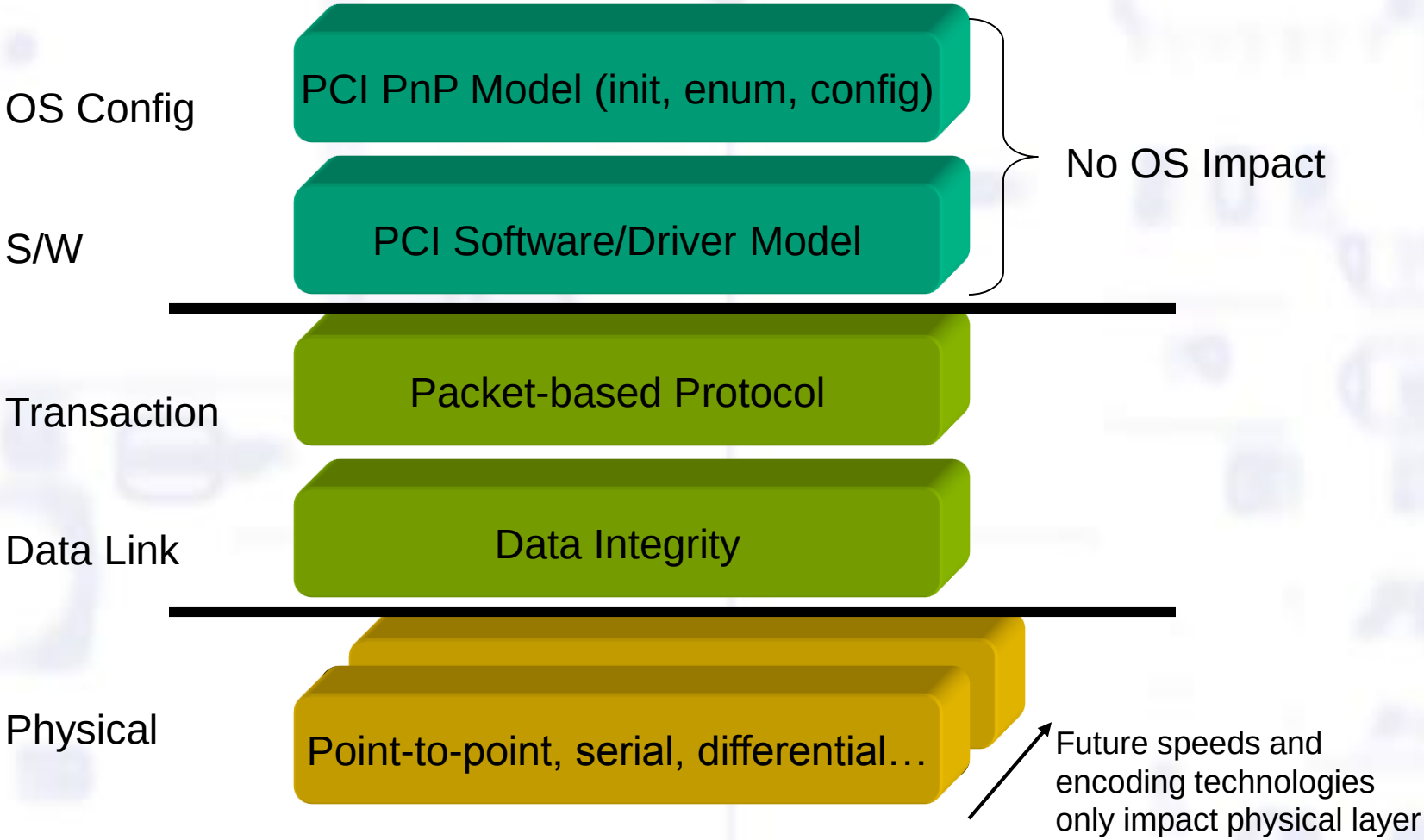
The specification only guarantees a x1 connection when down plugging (i.e. a x4 card in a x16 slot may only negotiate to a x1)

PCI Express Architecture

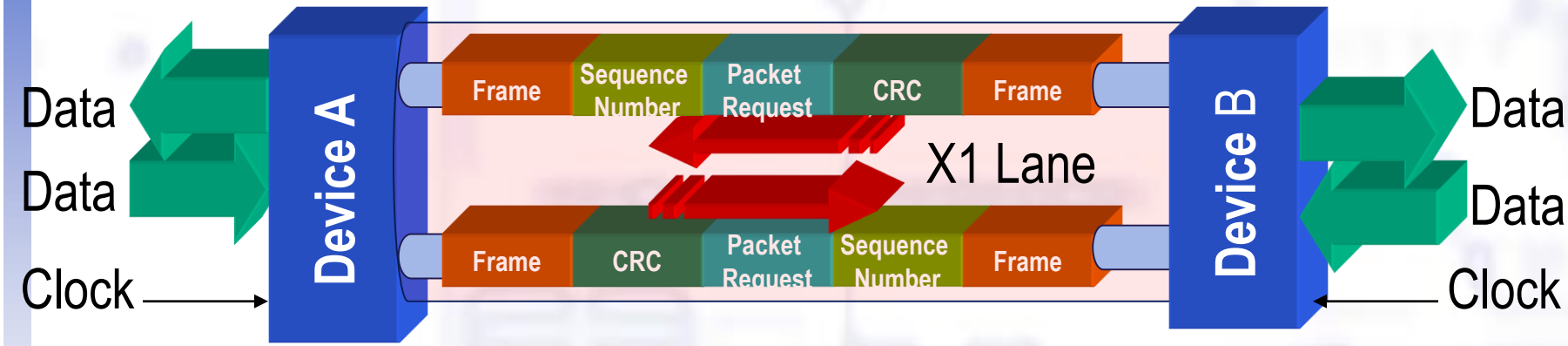
PCI Express Architecture

- PCI Express Layers
- 8B/10B Encoding
- Scalable Lane Widths
- Isochronous Transfer
- PCI Express Flow Control

PCI Express Layers



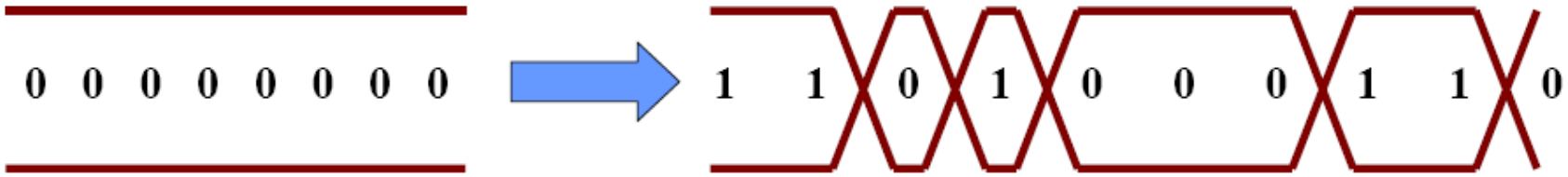
PCI Express Physical Layer



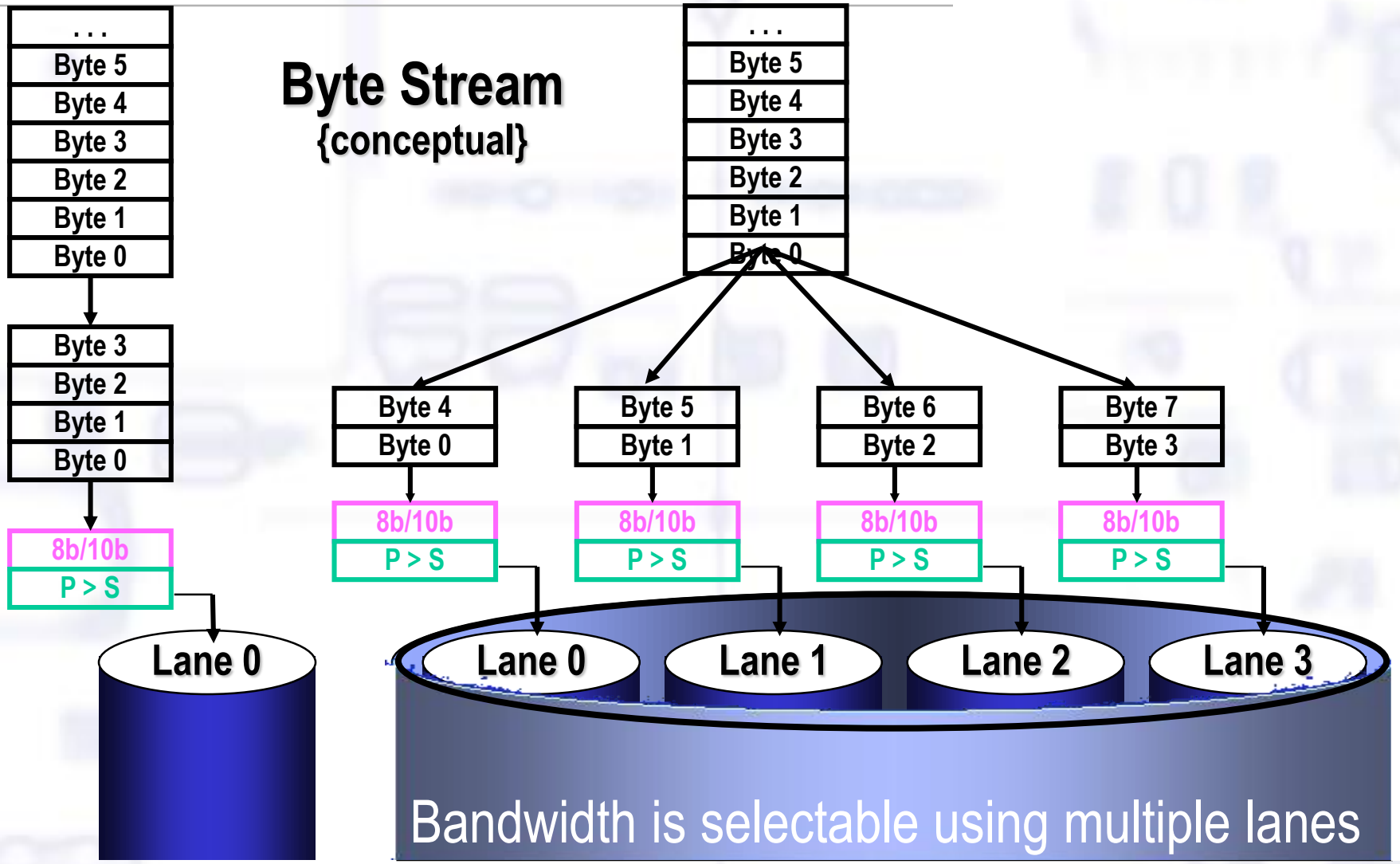
- Point to point differential interconnect with 2 endpoints
- Two unidirectional links, No Sideband Signals
- Bit rate: ≥ 2.5Gb/s/pin/dir and beyond
- Clocking: Embedded clock signaling using 8b/10b encoding
- Interface Width: Per direction: x1, x2, x4, x8, x12, x16, x32
- Gen-2 (5GB/s) speed increase in mid '06

8B/10B Encoding

- Consumes 20% of available bandwidth
  - Incorporated into 250 Mbytes/s bandwidth numbers already
- Guarantees minimum transition density
- Allows for clock recovery on each transition
- **Routing Flexibility**
  - No need to address clock/data skew



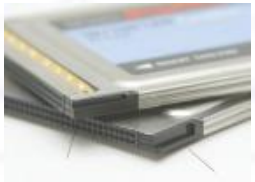
PCI Express Lane Width



Flow Control in PCI Express

- Credit-based flow control
  - Can only transmit if receiver has buffer space
  - No dropped packets
  - No retries (Doesn't send if packet cannot be received)
  - Congestion spreads "up the chain"
- Split transaction
  - Request packets requiring a response are split
- Handled entirely within the data link layer
- Frees up bandwidth

System bus extension



System bus extension PCCARDs

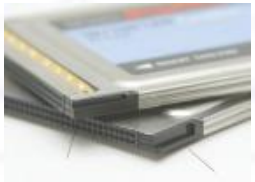
- System Bus
  - Extension
- Peripheral bus
- Inter chip communication

ISA =PCMCIA

- Low speed
- Easy interface
  - Fax modems
  - CF card reader

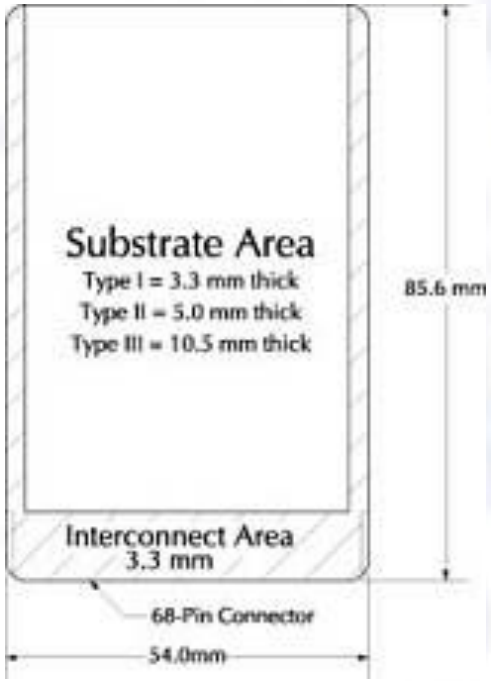
PCI = CARDBUS

- High speed
- More complex design
  - USB 2.0 card
  - WLAN card



Mechanical data

- Small form factor: 85.6mm x 54.0mm
- 68-pin connector
- Three card thicknesses supported



- System Bus
  - Extension
- Peripheral bus
- Inter chip communication

Details

PCMCIA

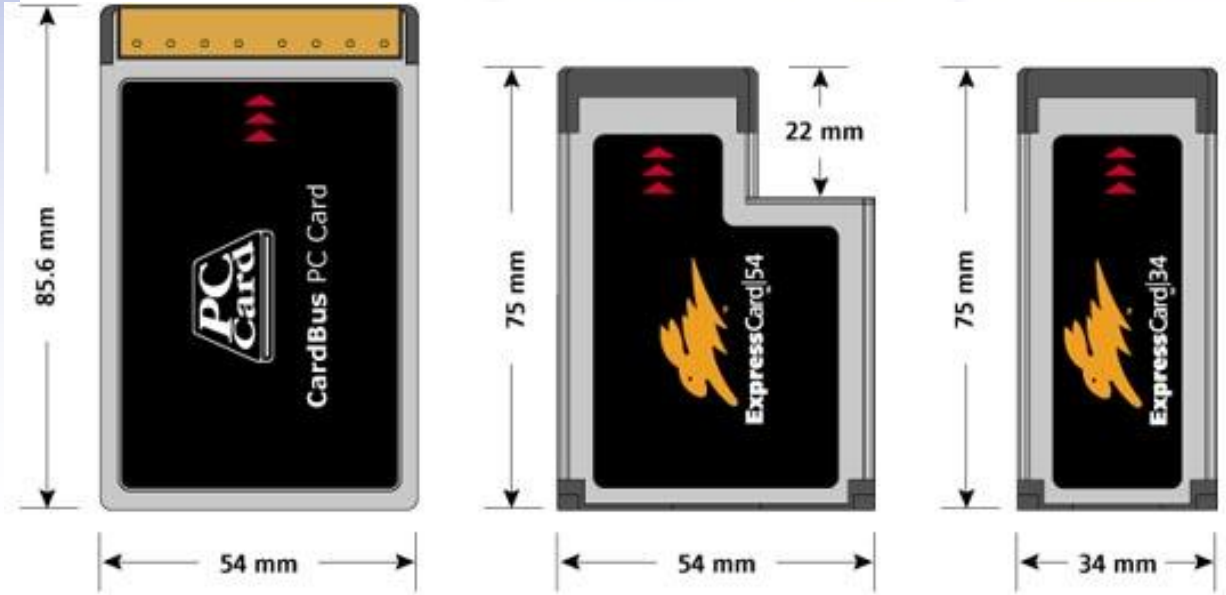
- 8-/16-bit interface at ISA bus speeds (8MHz) using ISA-like protocol

CardBus

- 32-bit multiplexed address/data
- Operates at PCI local-bus speeds
- Up to 33 MHz
- Peak bandwidth of 132MB/sec

ExpressCard : PCI Express for Laptops

- 34mm and 54 mm form-factors



Both USB 2.0 And PCIe signaling on host

In System

- System Bus
  - Extension
- Peripheral bus
- Inter chip communication

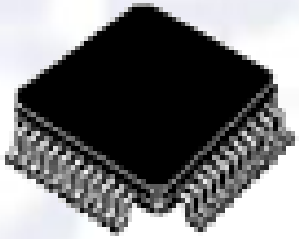
CPU

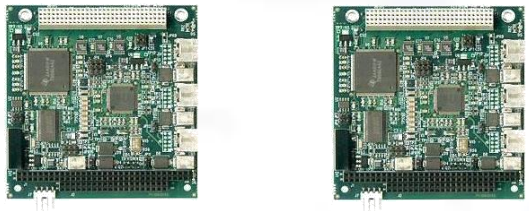
North-bridge

Memory

PCI 0

PCI – Cardbus Bridge





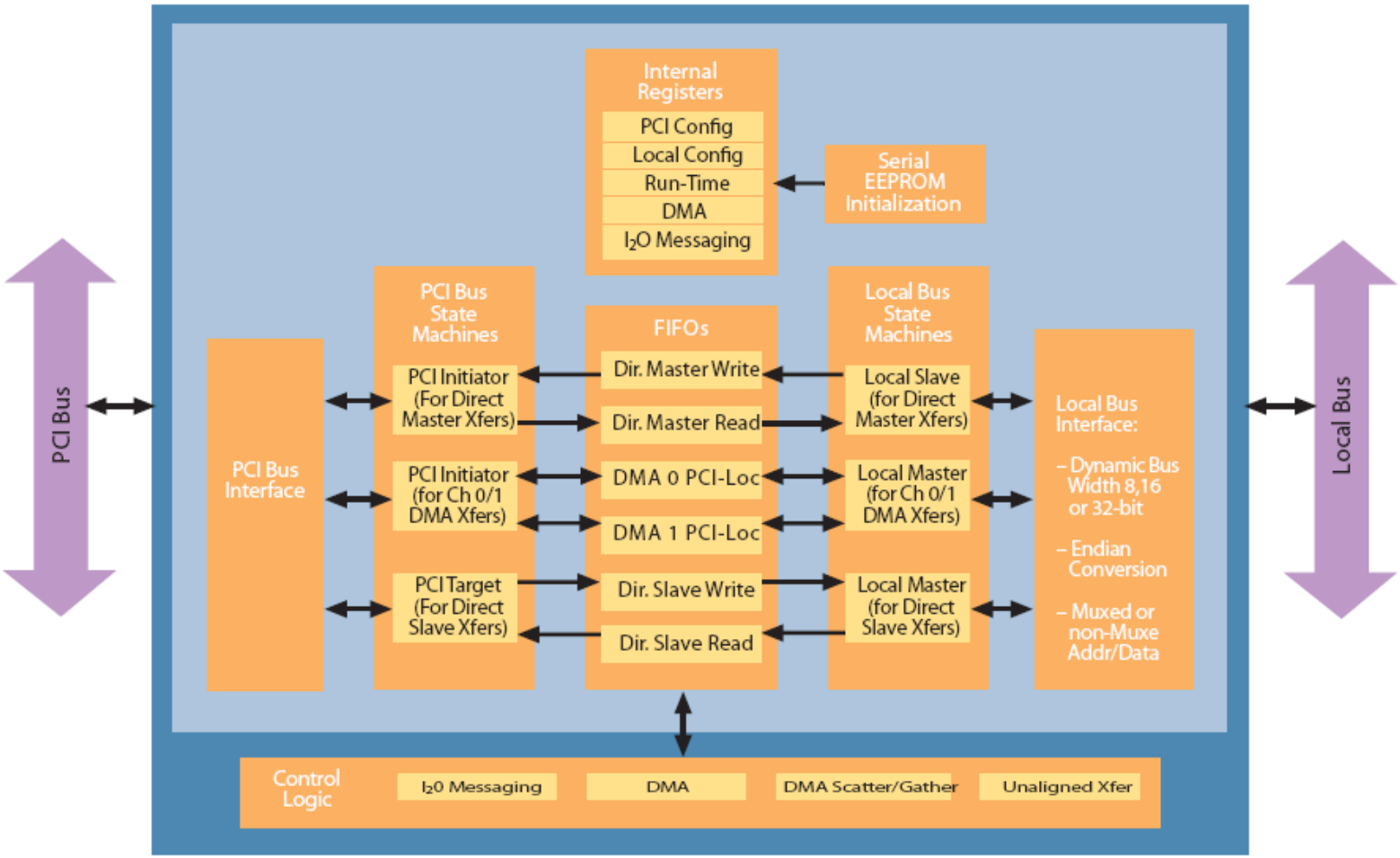
PCI devices

Interfacing to embedded systems

PLX9054 PCI to Localbus

- PCI v2.2 compliant 32 bit / 33Mhz
- 3.3 V & 5V tolerant IO
- Target & Initiator function
- Built in DMA engines
- Different local bus configuration:
  - 32 Bit Multiplexed
  - 8,16,32 Non multiplexed
- Programmable interrupt
- Industrial temp range.

PLX9054


Internal Registers
PCI Config
Local Config
Run-Time
DMA
I2O Messaging
Serial EEPROM Initialization
PCI Bus
PCI Bus Interface
PCI Bus State Machines
PCI Initiator (For Direct Master Xfers)
PCI Initiator (for Ch 0/1 DMA Xfers)
PCI Target (For Direct Slave Xfers)
FIFOs
Dir. Master Write
Dir. Master Read
DMA 0 PCI-Loc
DMA 1 PCI-Loc
Dir. Slave Write
Dir. Slave Read
Local Bus State Machines
Local Slave (for Direct Master Xfers)
Local Master (for Ch 0/1 DMA Xfers)
Local Master (for Direct Slave Xfers)
Local Bus Interface:
– Dynamic Bus Width 8,16 or 32-bit
– Endian Conversion
– Muxed or non-Muxe Addr/Data
Local Bus
Control Logic
I20 Messaging
DMA
DMA Scatter/Gather
Unaligned Xfer

PLX local bus to processor interface

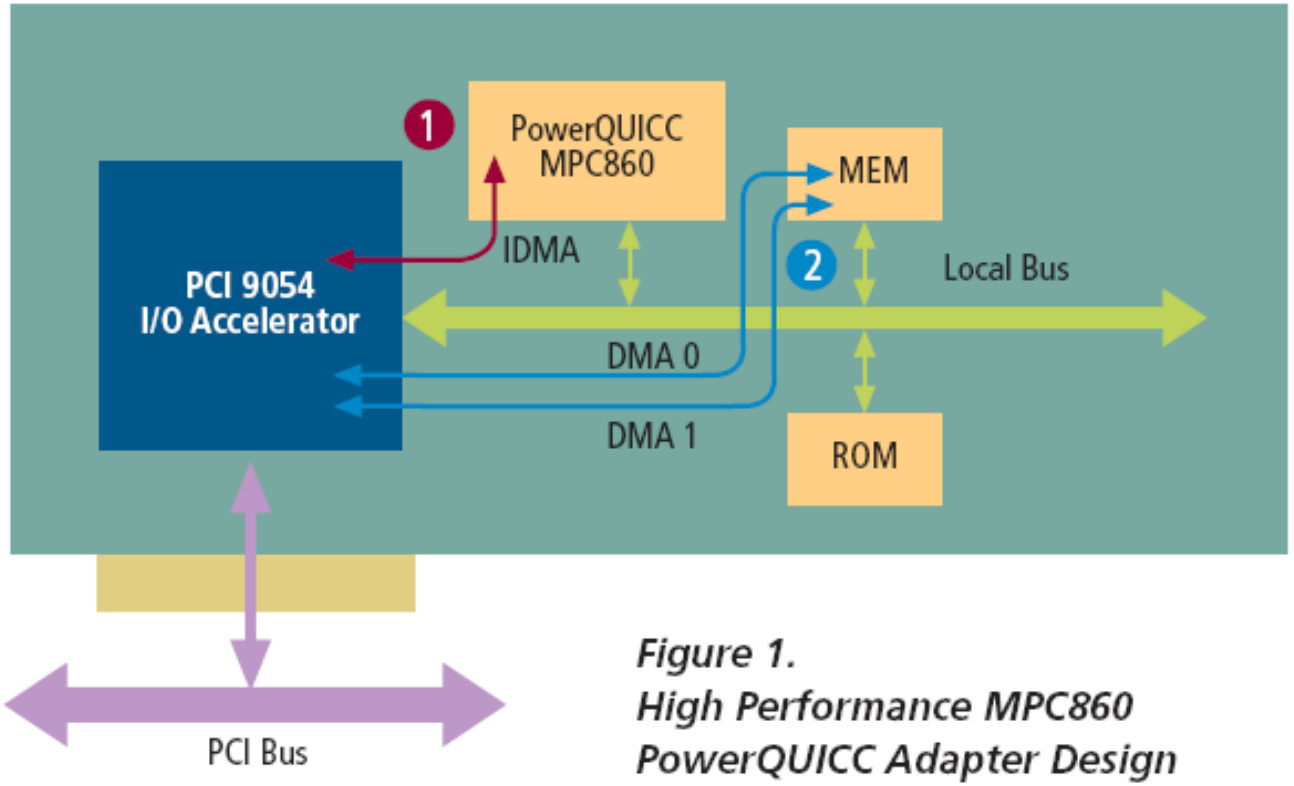
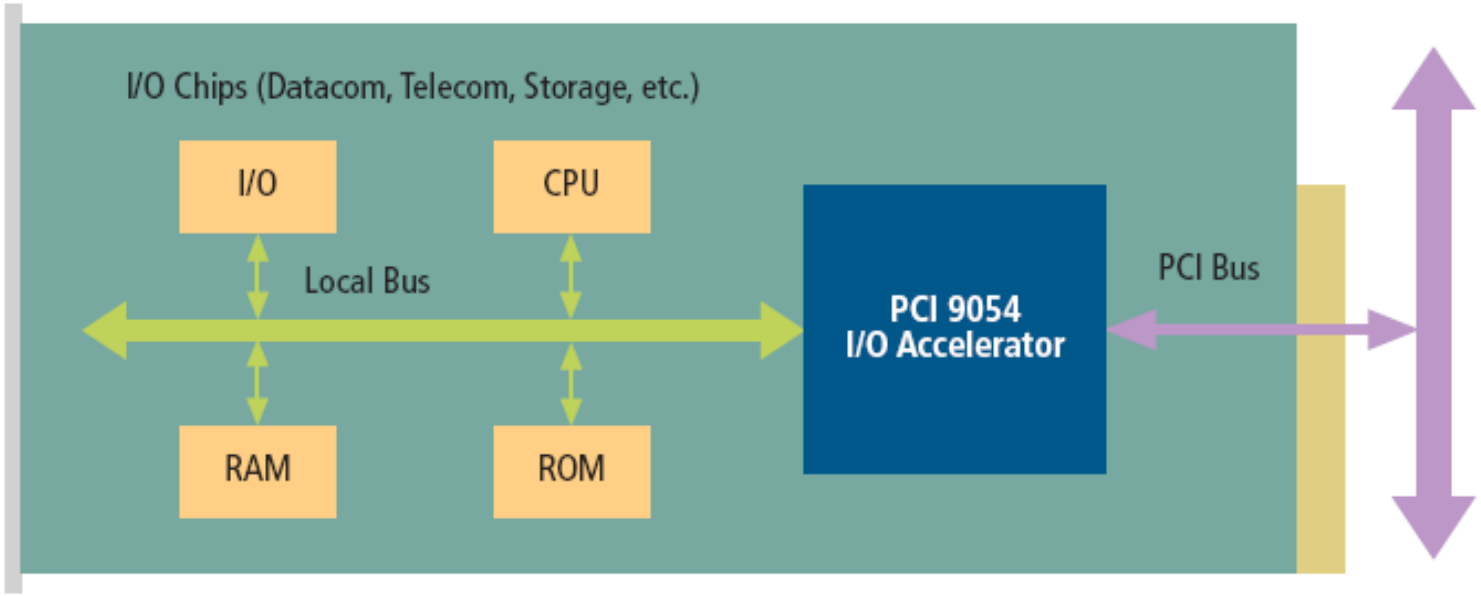


Figure 1.
High Performance MPC860
PowerQUICC Adapter Design

PLX local bus



PEX 8311 – PCI Express to Localbus

- X1 Link @ 2.5 Gbit/s
- Max 264 MB/s burst transfer
- 2 DMA channles
- Multiplexed and Non-multiplexed mode
- Root complex & Endpoint operation
- <1 W
- 337 pin BGA

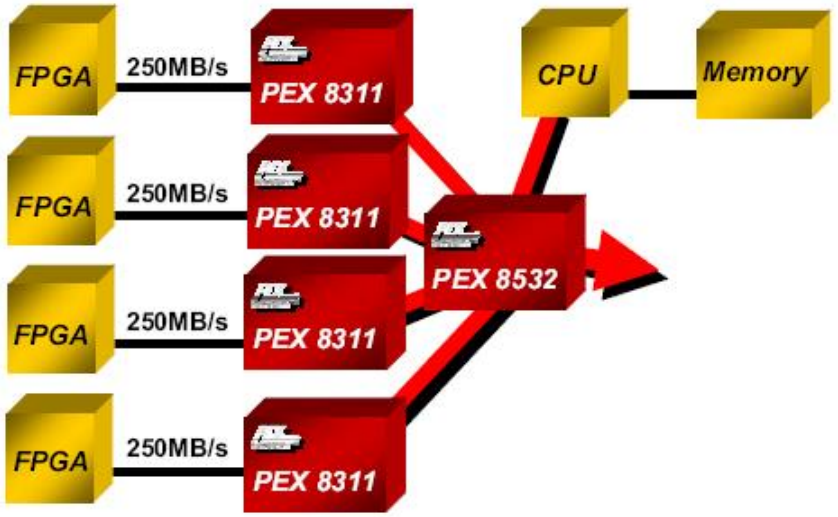


Figure 1

Peripherals

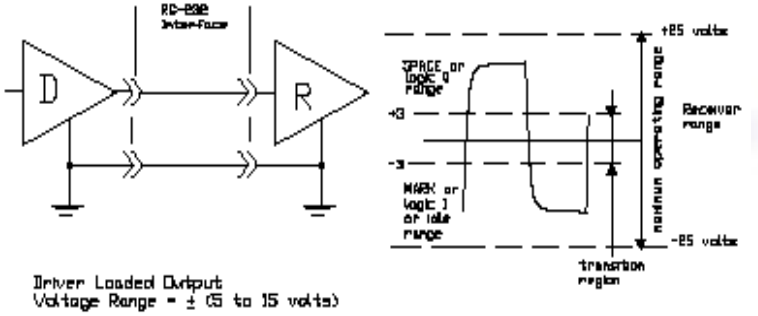
- System Bus
- Peripheral bus
- Inter chip communication

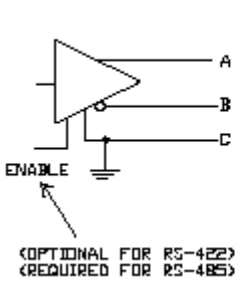
Grouping of peripherals

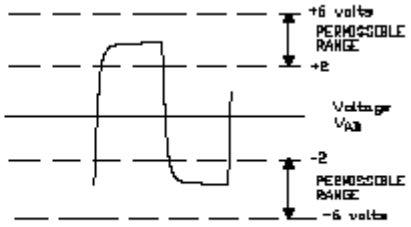
- By bus width
  - Serial
  - Parallel
- By physical transmission
  - Single ended
  - Differential
- By existence of clock signal
  - Synchronous
  - Asynchronous
- By number of devices
  - Point –to- point
  - Point-to-multipoint
  - Multipoint-to- multipoint

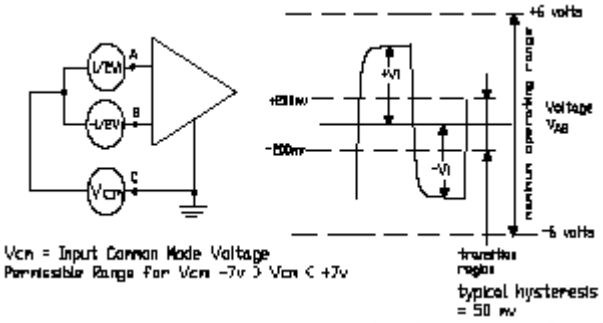
- System Bus
- Peripheral bus
- Inter chip communication

Transmission types







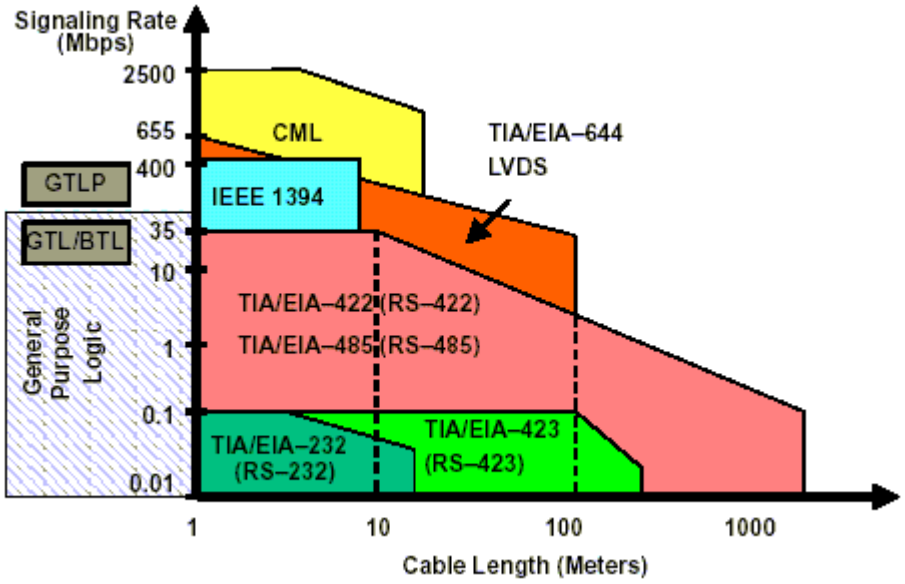


EIA/TIA standards

- System Bus
- Peripheral bus
- Inter chip communication

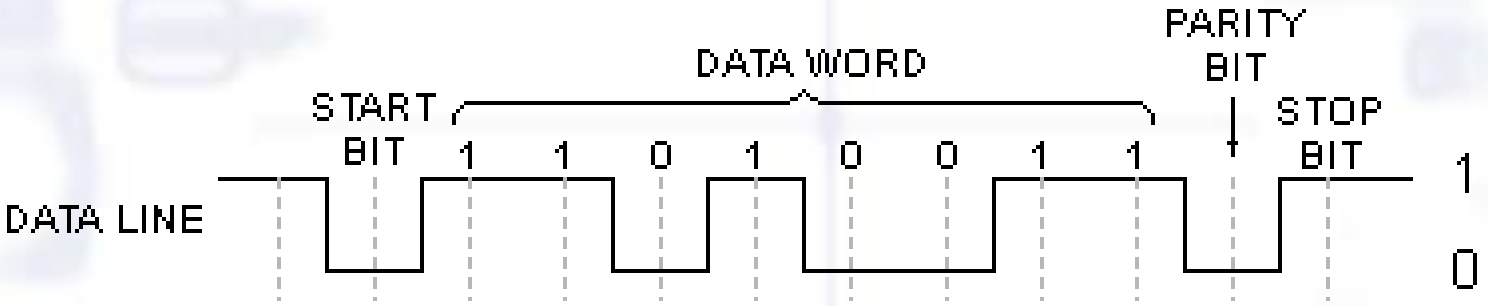
EIA = Electronic Industries Association

TIA = Telecommunications Industry Association



RS – 232 timing

- **Start bit** -1 bit
- **Data word** -7/8 bits
- **Parity** -1 bit
  - None / Even /Odd
- **Stop bit** -1/2 bits

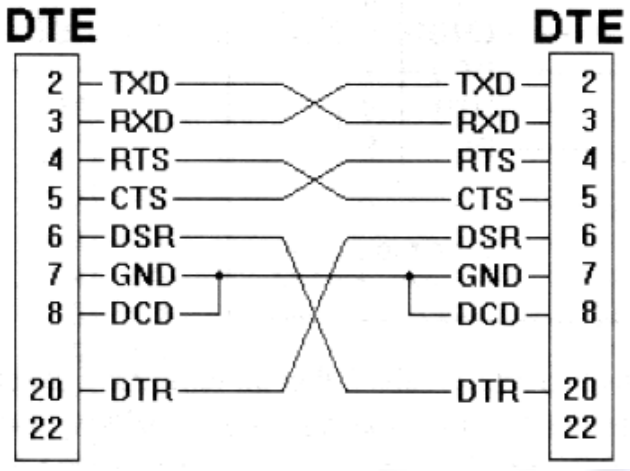


Handshaking

- System Bus
- Peripheral bus
  - RS232
- Inter chip communication

- 3 Wire (Transmit, Receive, Ground) is enough to transmit data
- What happens if the receiver is overloaded? → Data loss!

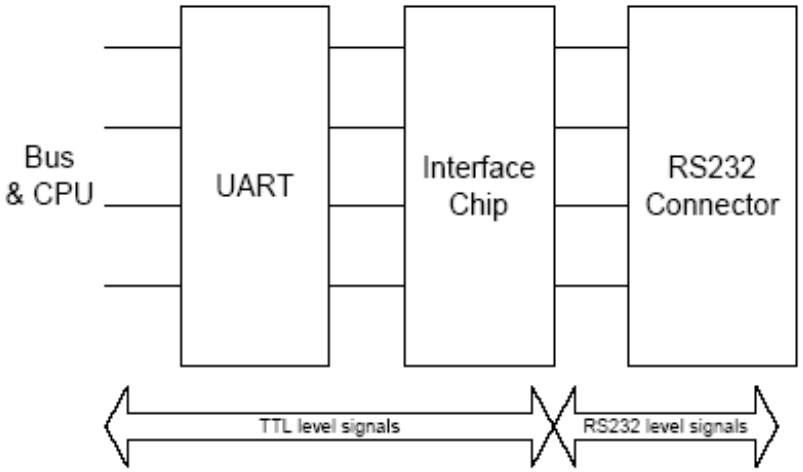
- **Software handshake:**
  - XON / XOFF to enable data transfer
  - Problem: 2 character is not available
    → Not good for binary transfer

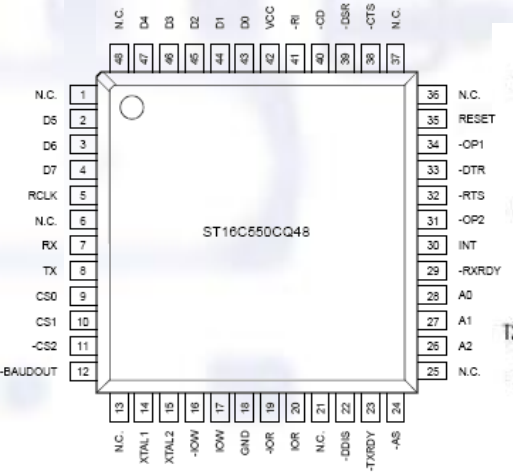


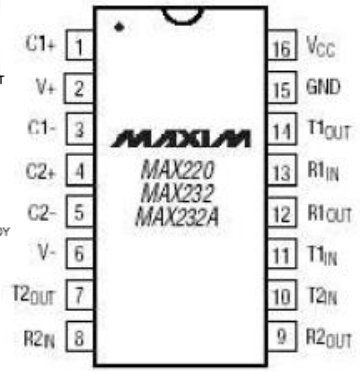
- **Hardware handshake:**
  - RTS/CTS for FIFO control
  - DTR/DSR for communication establishment

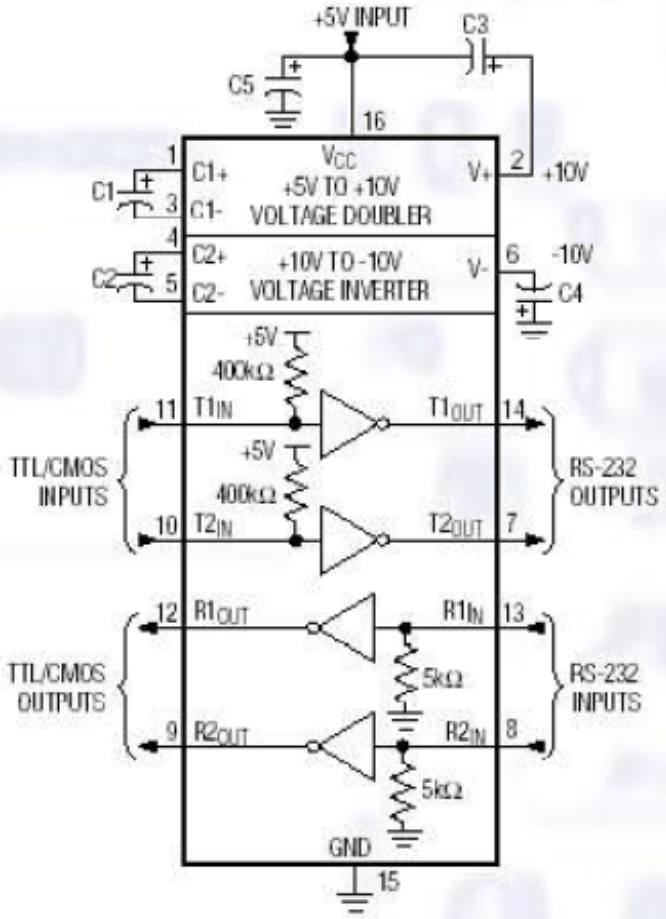
Typical system architecture

- System Bus
- Peripheral bus
  - RS232
- Inter chip communication





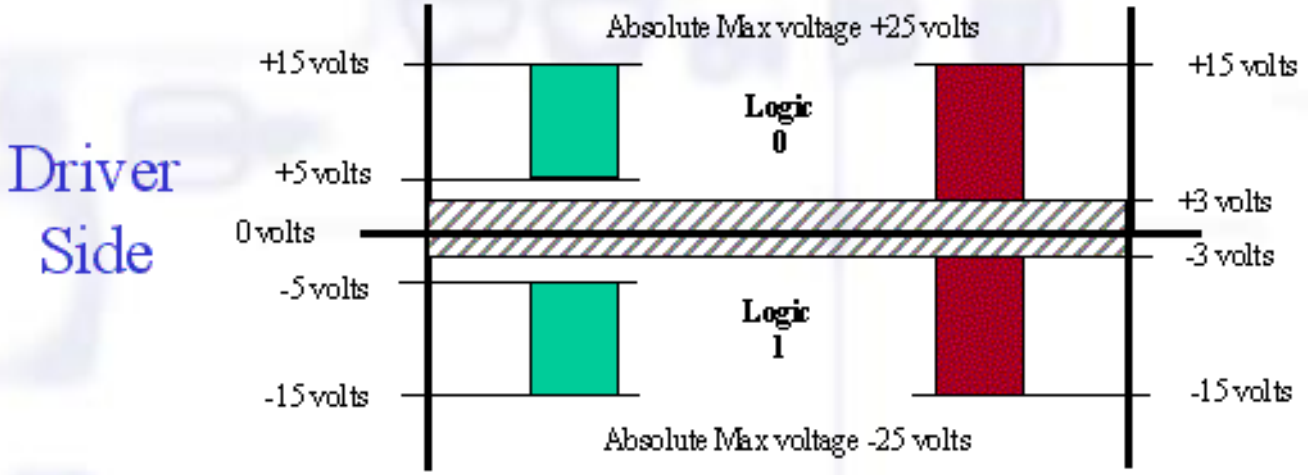




RS – 232 physical interface

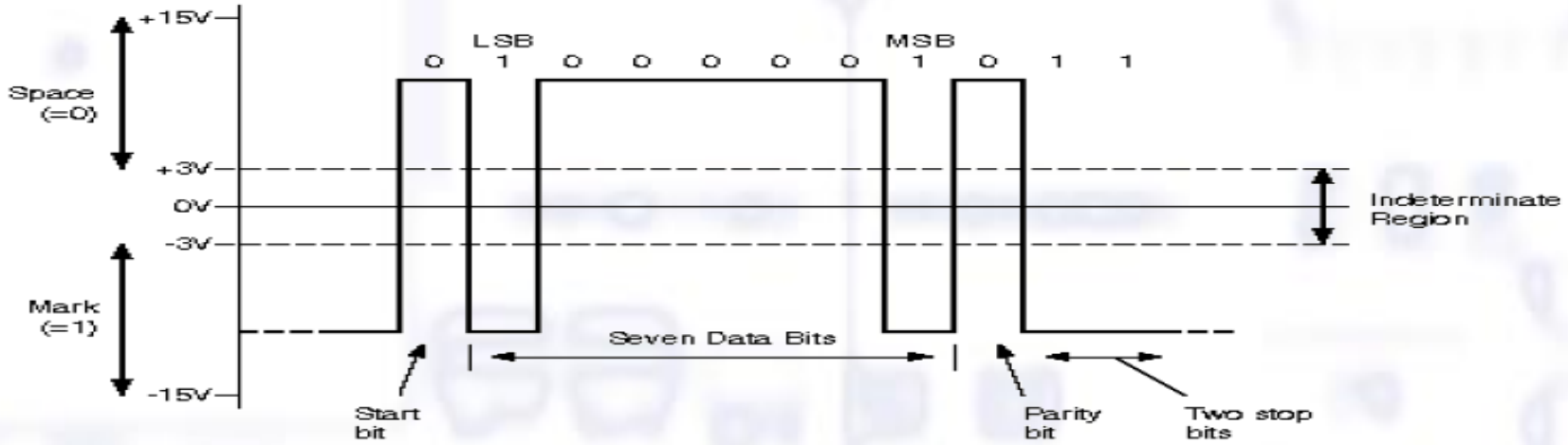
RS232 transmitter voltage values (V)

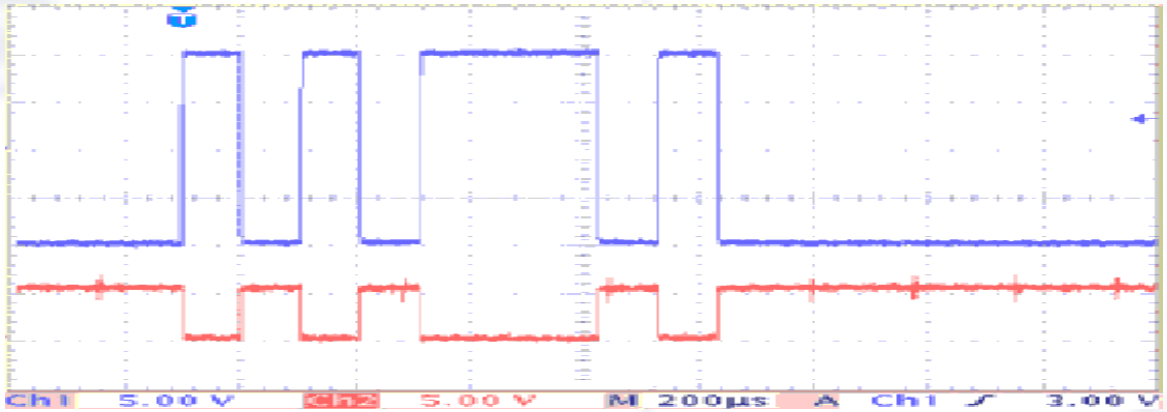
Space state (0)	+5 ... +15	+3 ... +25
Mark state (1)	-5 ... -15	-3 ... -25
Undefined	-	-3 ... +3



- System Bus
- Peripheral bus
  - RS232
- Inter chip communication

Voltage levels





Cable length

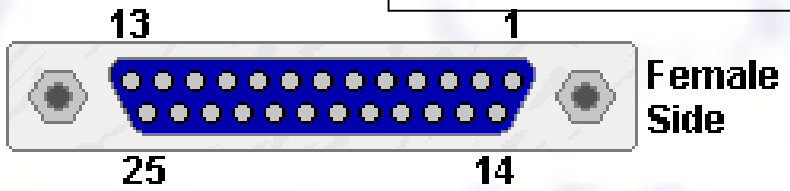
- Original specification : 20 kbit/s - 50 feet - 2500 pF
- Low capacitance cable:
  - Bigger distance
    - E.g.: CAT5 : 17 pF/feet → 147 feet
    - Half the speed → Ten times distance

RS232 cable length

Baud rate	Maximum cable length (ft)
19200	50
9600	500
4800	1000
2400	3000

- System Bus
- Peripheral bus
  - RS232
- Inter chip communication

RS - 232 pinout



9 pin #	25 pin#	Name	Full name	Dir	Meaning
3	2	TxD	Transmit Data	—»	Transmits bytes out of PC
2	3	RxD	Receive Data	«—	Receives bytes into PC
7	4	RTS	Request To Send	—»	RTS/CTS flow control
8	5	CTS	Clear To Send	«—	RTS/CTS flow control
6	6	DSR	Data Set Ready	«—	I'm ready to communicate
4	20	DTR	Data Terminal Ready	—»	I'm ready to communicate
1	8	DCD	Data Carrier Detect	«—	Modem connected to another
9	22	RI	Ring Indicator	«—	Telephone line ringing
5	7	SG	Signal Ground		

- System Bus
- Peripheral bus
- Inter chip communication

Field Buses

- System Bus
- Peripheral bus
  - Fieldbus
- Inter chip communication

Fieldbus

FIELD + BUS

Limited to a local area or “field”

Interconnection between multiple devices

- serial networking of components
- two-way, multi-drop communication links

- System Bus
- Peripheral bus
  - Fieldbus
- Inter chip communication

Why fieldbus ?

- A single bus cable replaces parallel cables
- The user is independent of any manufacturer-specific standards
- The system can be easily extended or modified
- Improved reliability and increased availability due to short signal paths
- Installation and startup time are reduced
- Self-diagnostics

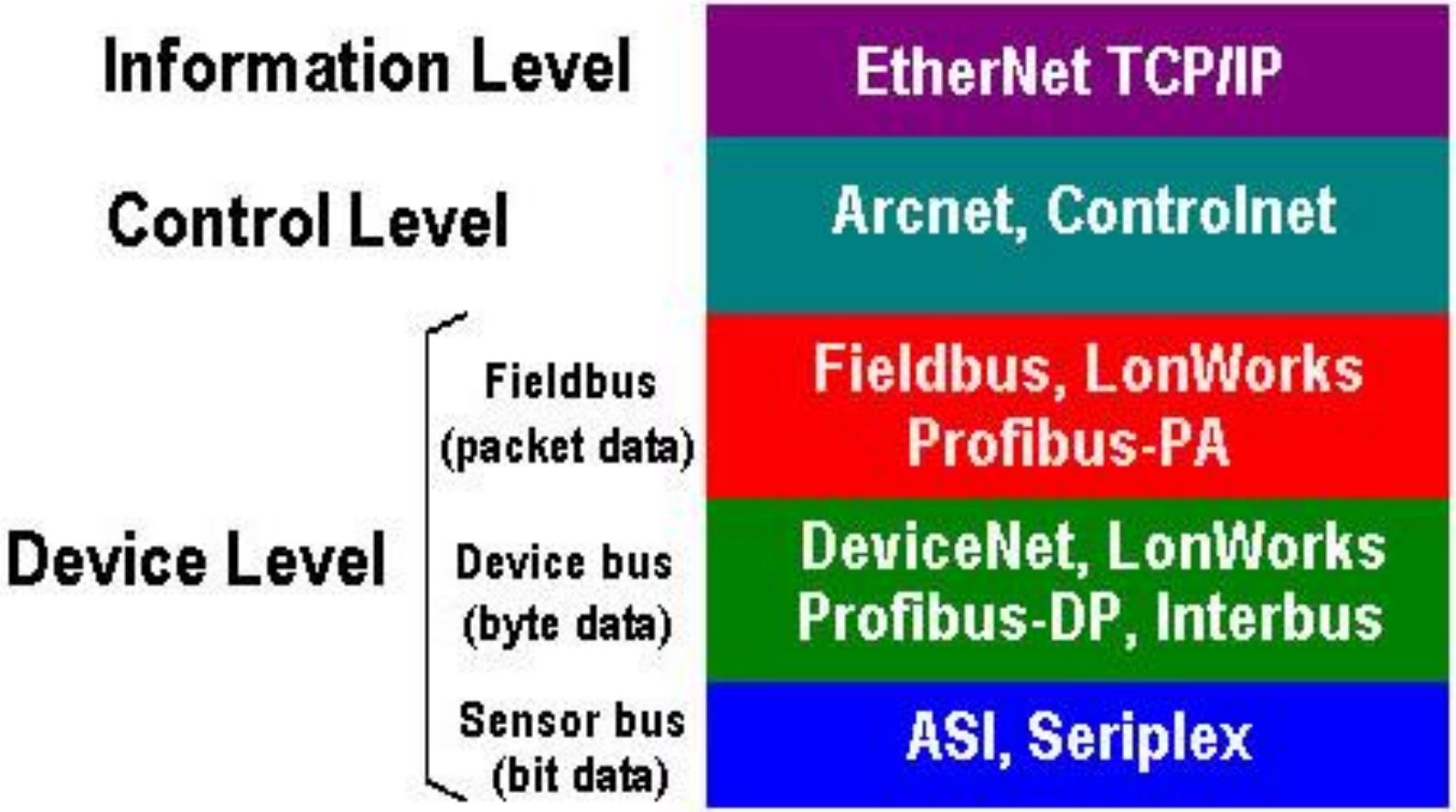
- System Bus
- Peripheral bus
  - Fieldbus
- Inter chip communication

Fieldbus Examples

- elevator control
- home automation
- medical instrumentation
- security systems
- slot machines
- traffic lights
- vehicle wiring systems
- vending machines
- automated supermarket pricing

- System Bus
- Peripheral bus
  - Fieldbus
- Inter chip communication

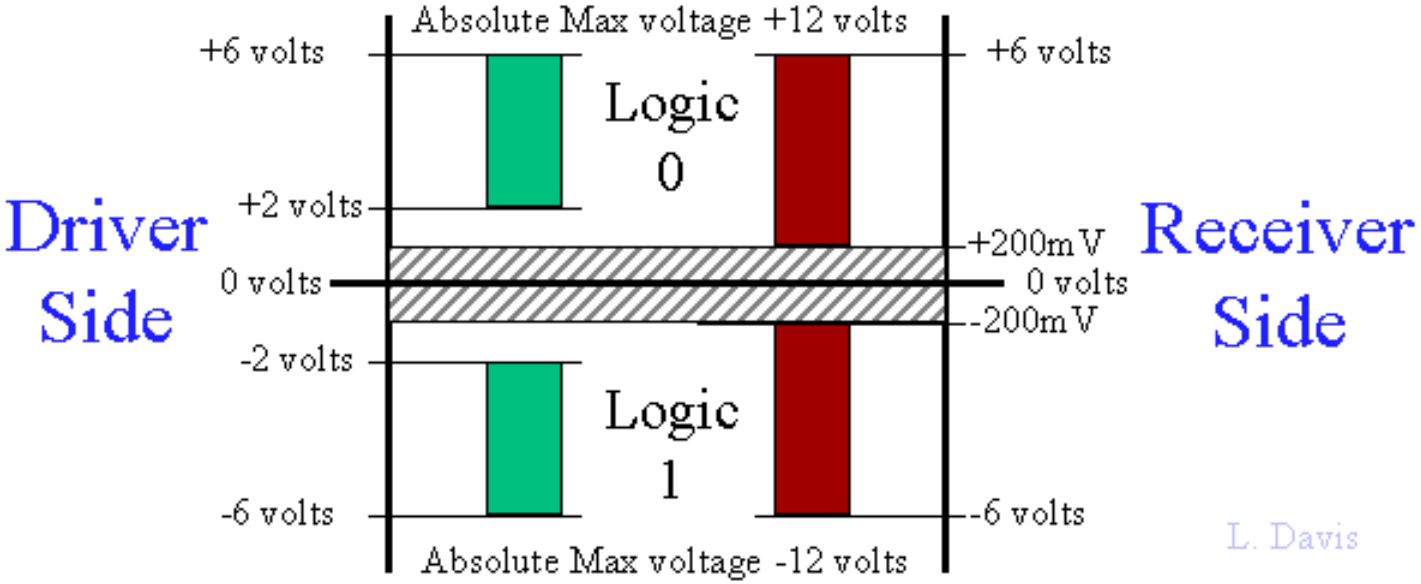
Fieldbus Types



Other „RS” standards

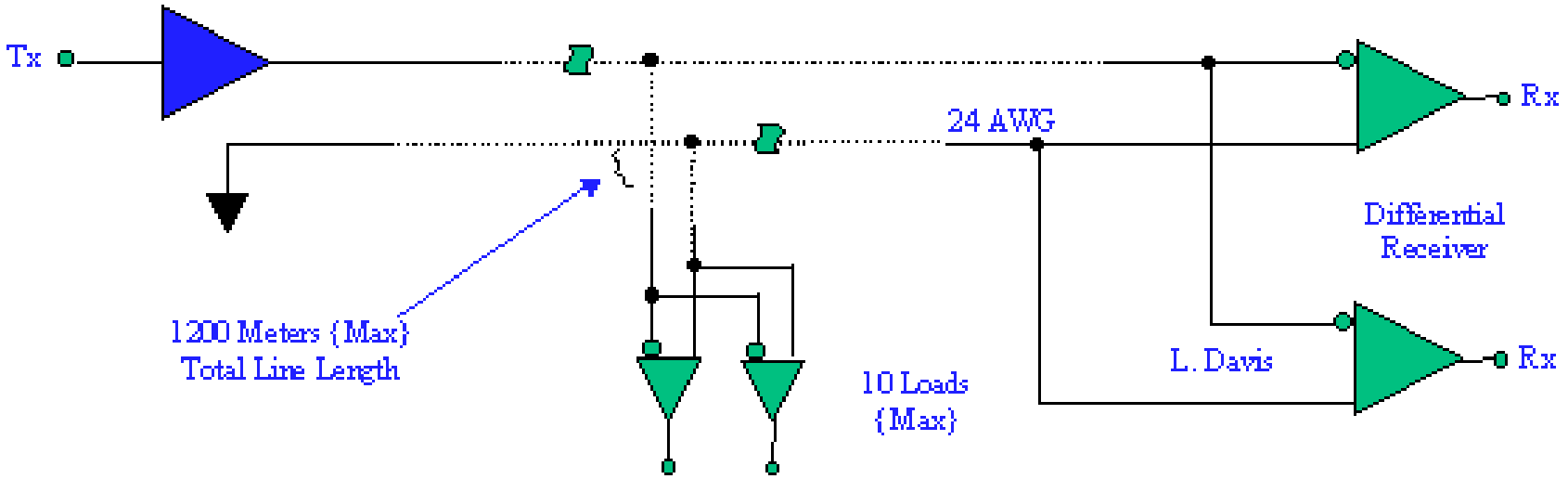
- System Bus
- Peripheral bus
- Inter chip communication

Differential voltage levels



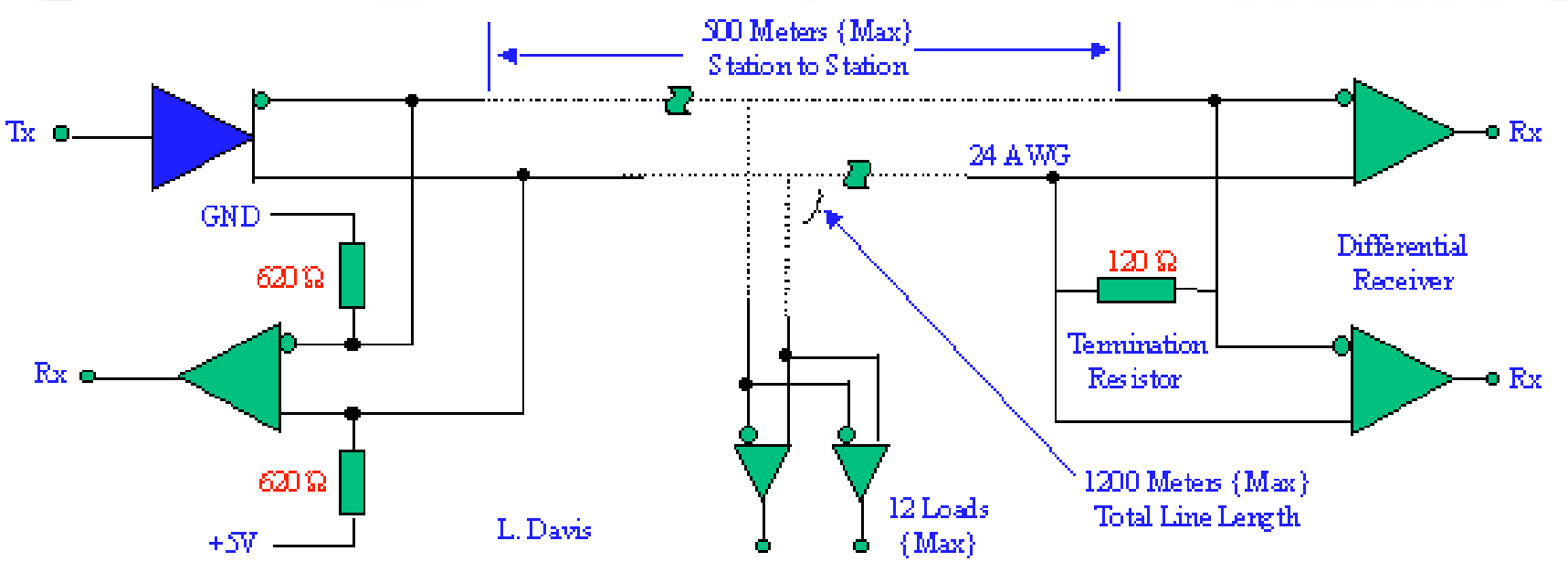
EIA/TIA-423

- System Bus
- Peripheral bus
- Inter chip communication



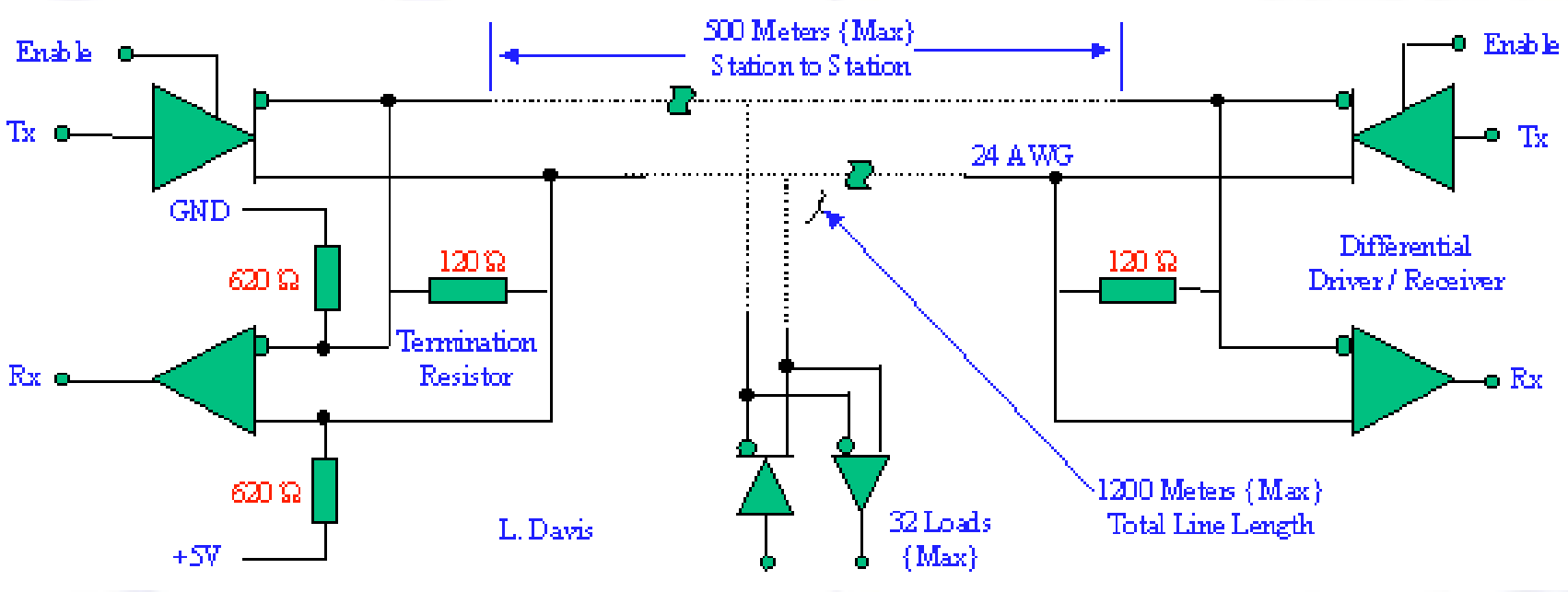
RS – 422

- System Bus
- Peripheral bus
- Inter chip communication



RS - 485

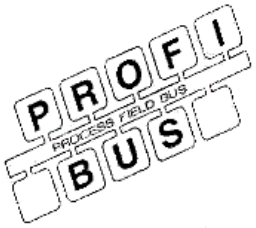
- System Bus
- Peripheral bus
- Inter chip communication



- System Bus
- Peripheral bus
- Inter chip communication

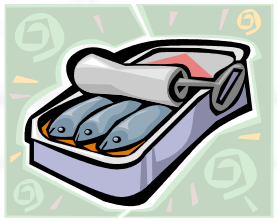
Higher protocol layers

- Profibus
- Interbus
- P-Net
- Bitbus
- Modbus
- LonWorks



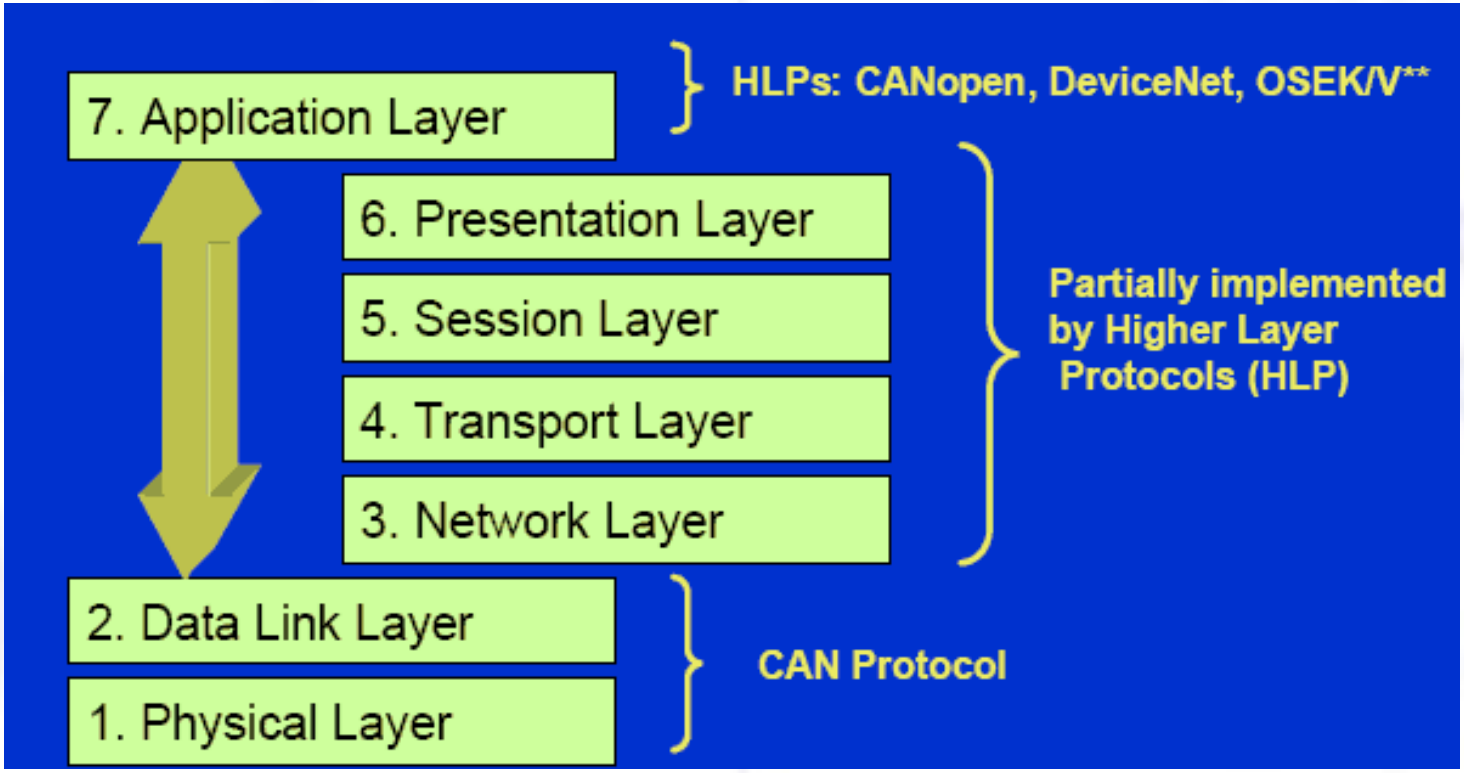
Handshaking

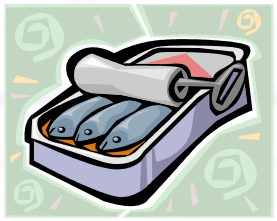
CAN bus



CAN Overview

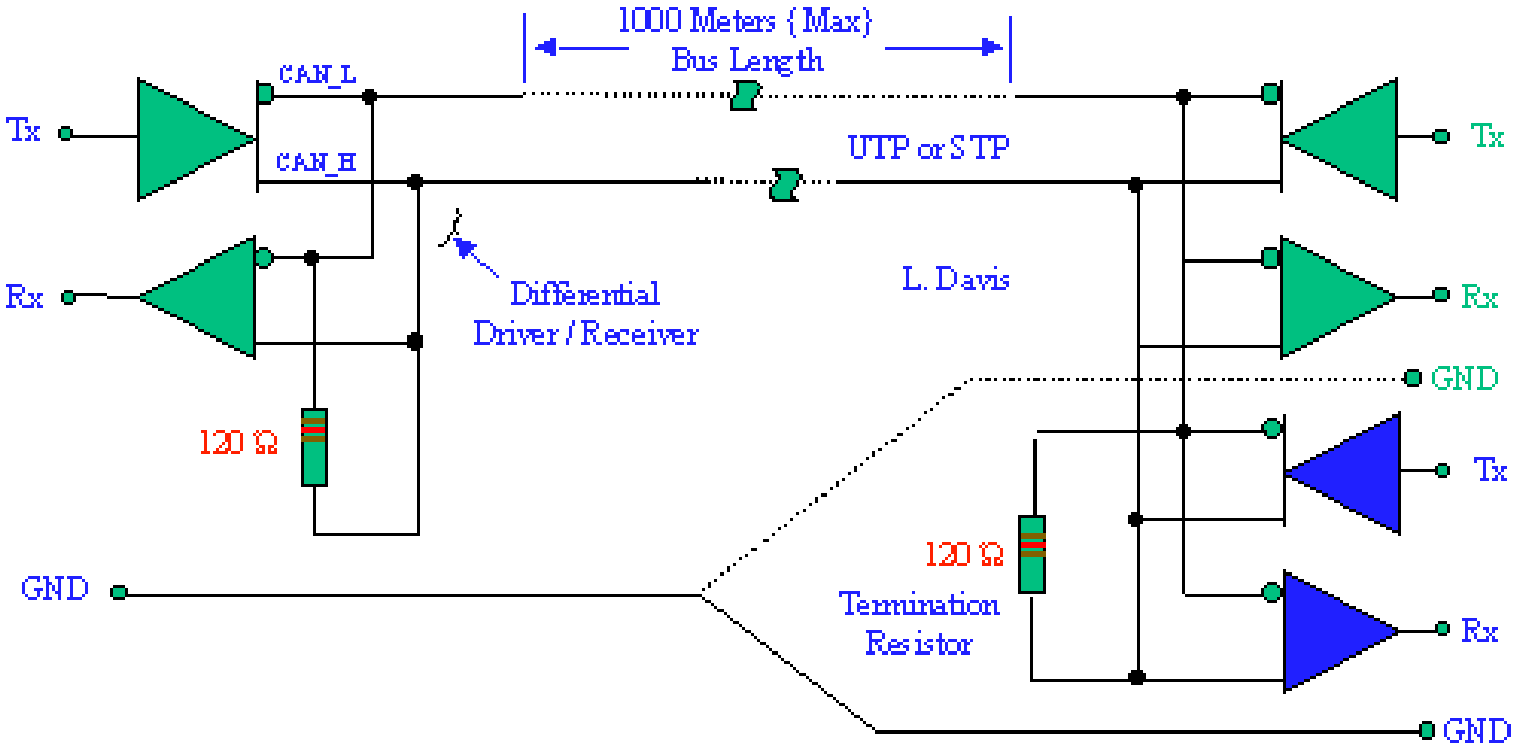
- System Bus
- Peripheral bus
  - CAN
- Inter chip communication

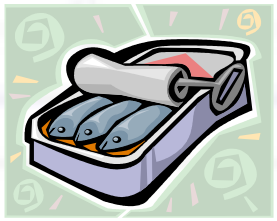




- System Bus
- Peripheral bus
  - CAN
- Inter chip communication

CAN Topology

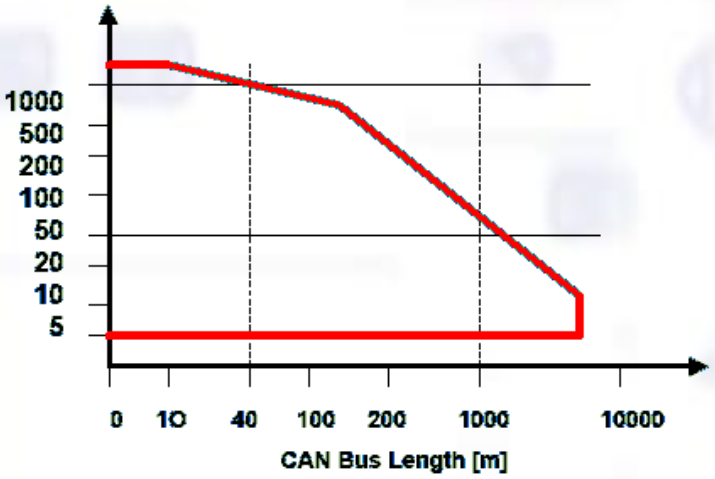




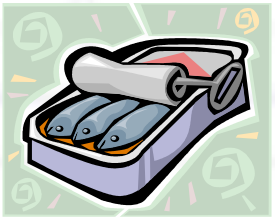
- System Bus
- Peripheral bus
  - CAN
- Inter chip communication

CAN Physical layer I.

Bus length (metres)	Maximum bit rate (bit/s)
40	1 Mbit/s
100	500 kbit/s
200	250 kbit/s
500	125 kbit/s
6 km	10 kbit/s



- Rs 232 -> 9.6 Kbit/s 200m



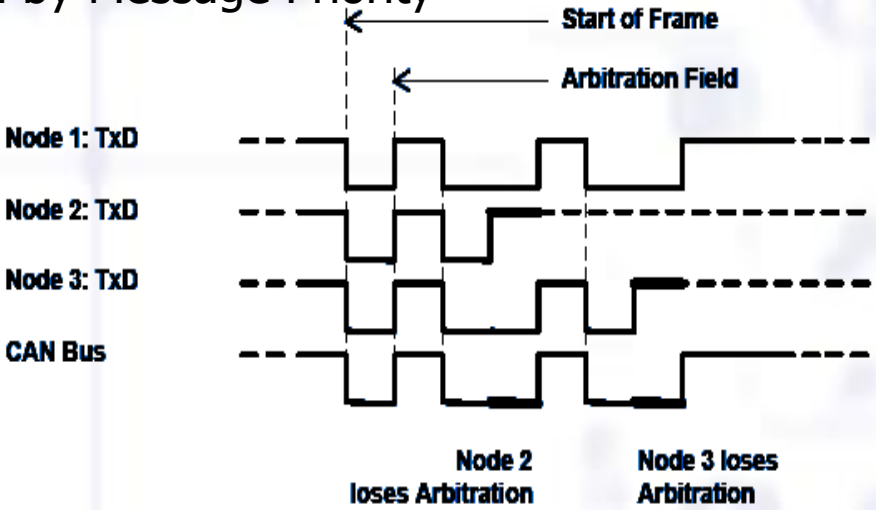
- System Bus
- Peripheral bus
  - CAN
- Inter chip communication

CAN Physical layer II.

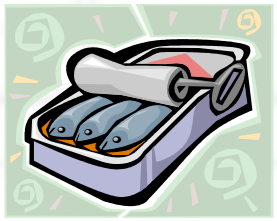
ISO 11898

Node A	Node B	Node C	BUS
D	D	D	D
D	D	R	D
D	R	D	D
D	R	R	D
R	D	D	D
R	D	R	D
R	R	D	D
R	R	R	R

- CSMA/CD
  - Carrier Sense Multiple Access/ Collision Detection
- AMP
  - Arbitration by Message Priority

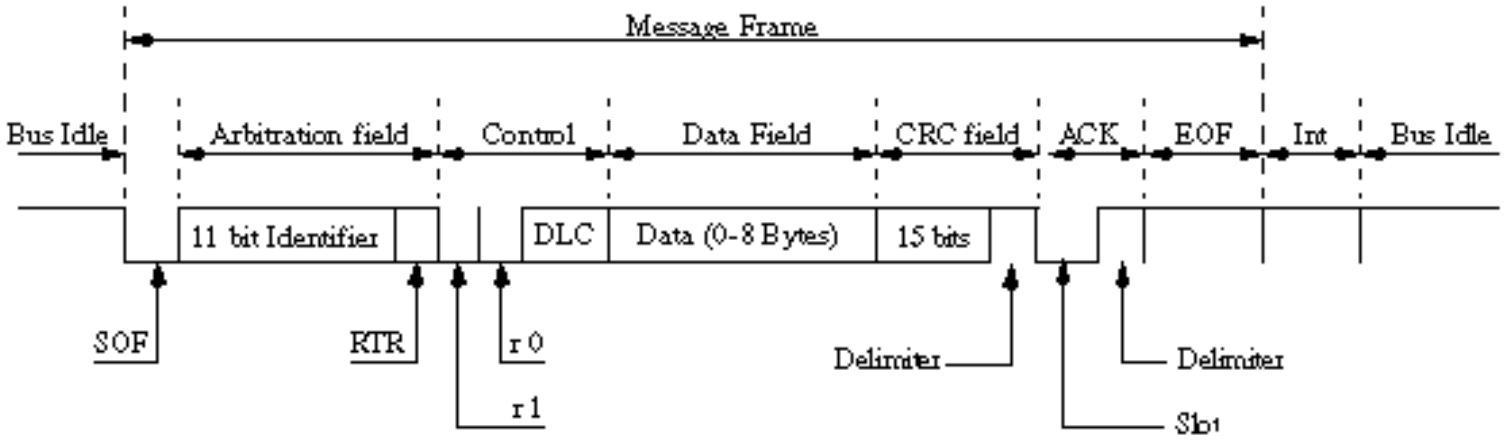


Signal	recessive state	dominant state
CAN-High	2.5	3.5
CAN-Low	2.5	1.5

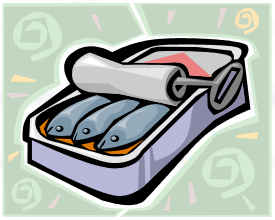


- System Bus
- Peripheral bus
  - CAN
- Inter chip communication

Datalink layer



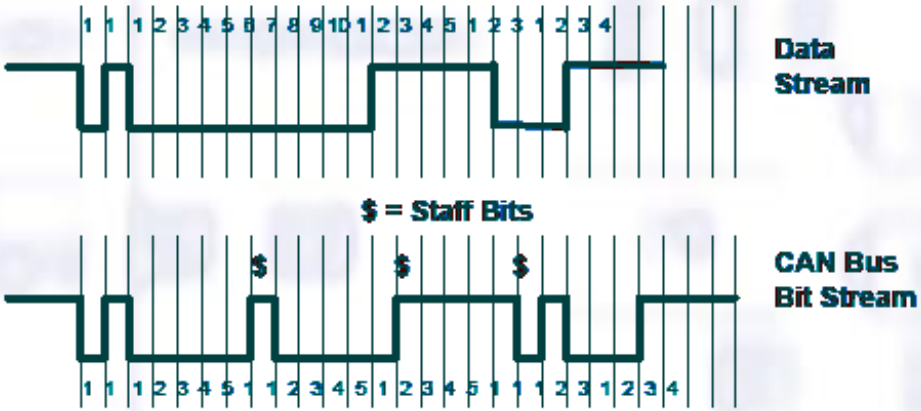
- The CAN protocol supports two Message Frame formats.
  - - Standard CAN (Version 2.0A)
  - - Extended CAN (Version 2.0B)

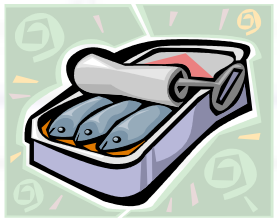


Error detection

- **Bit errors:**
  - Bit stuffing error
  - Bit error
- **Message errors:**
  - Checksum error
  - Frame error
  - Acknowledgement Error

Bit stuffing



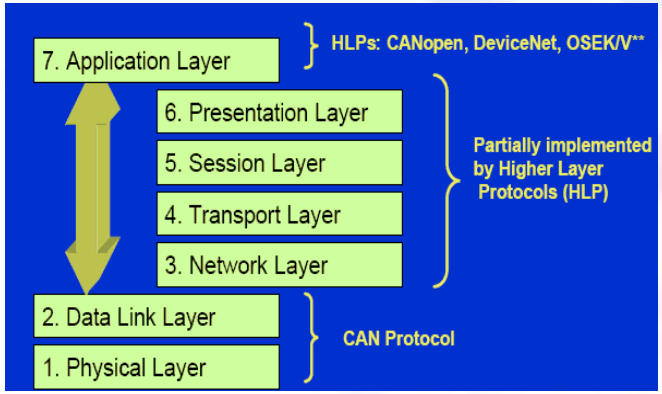


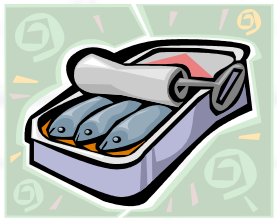
Higher protocol layers

- System Bus
- Peripheral bus
  - CAN
- Inter chip communication

- **The CAN protocol defines only the 'physical' and a low 'data link layer'!**

- **The HLP defines:**
  - Start-up behavior
  - Definition of message identifiers for the different nodes
  - Flow control
  - transportation of messages > 8bytes
  - Definition of contents of Data Frames
  - Status reporting in the system



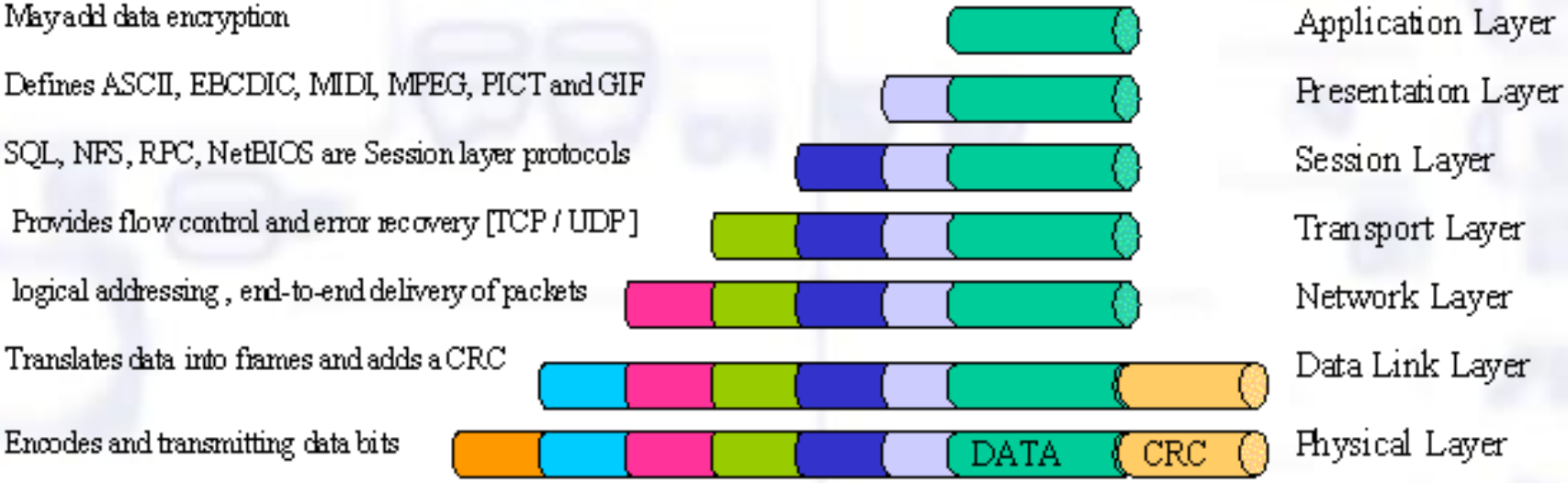


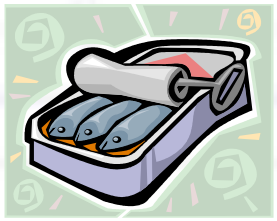
- System Bus
- Peripheral bus
  - CAN
- Inter chip communication

Higher protocol layers

- OSI /ISO layer stack

  International Standard Organization's Open System Interconnect





Higher protocol stacks

- **CANopen**
  - Auto configuration the network
  - Easy access to all device parameters
  - Device synchronization
  - Cyclic and event-driven data transfer
  - Synchronous reading or setting of inputs, outputs or parameters
- **DeviceNet**
  - Presented by the users group ODVA (Open DeviceNet Vendor Association)
  - Power and signal on the same network cable
  - Bus addressing by: Peer-to-Peer with multi-cast & Multi-Master & Master-Slave
  - Supports only standard CAN