PHILIPS

Serial Bus Overview

UART, CAN, USB, 1394, SPI and I^2C/SMBus

2003

Agenda

- Serial Bus Overview
  - UART Overview
  - CAN Overview
  - USB Overview
  - 1394 Overview
  - SPI Overview
  - I^2C/SMBus Overview
- Serial Bus Summary

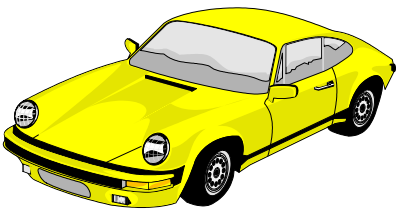
Serial Bus Overview

Communications

USB
UNIVERSAL SERIAL BUS

Consumer

Automotive



SERIAL BUSES

IEEE1394

UART

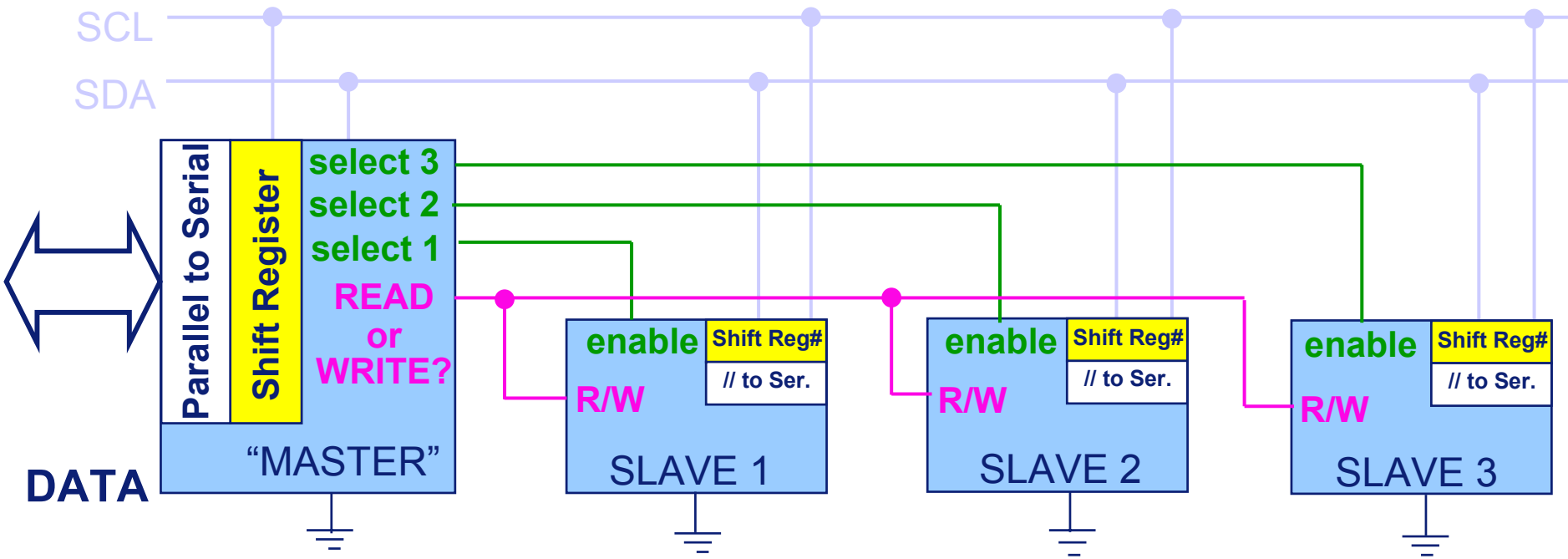
CAN
CONTROLLER AREA NETWORK

SPI

I²C BUS

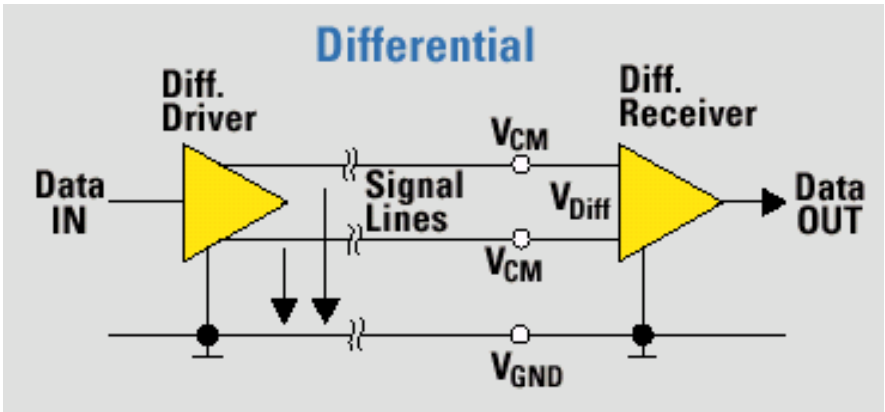
Industrial

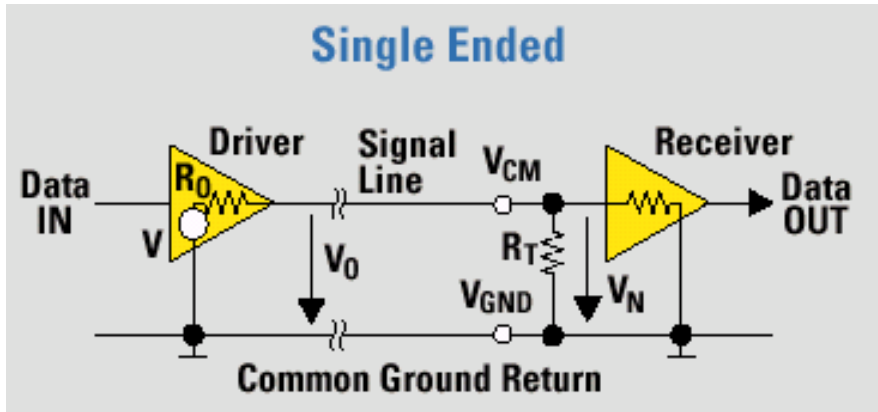
General concept for Serial Communications

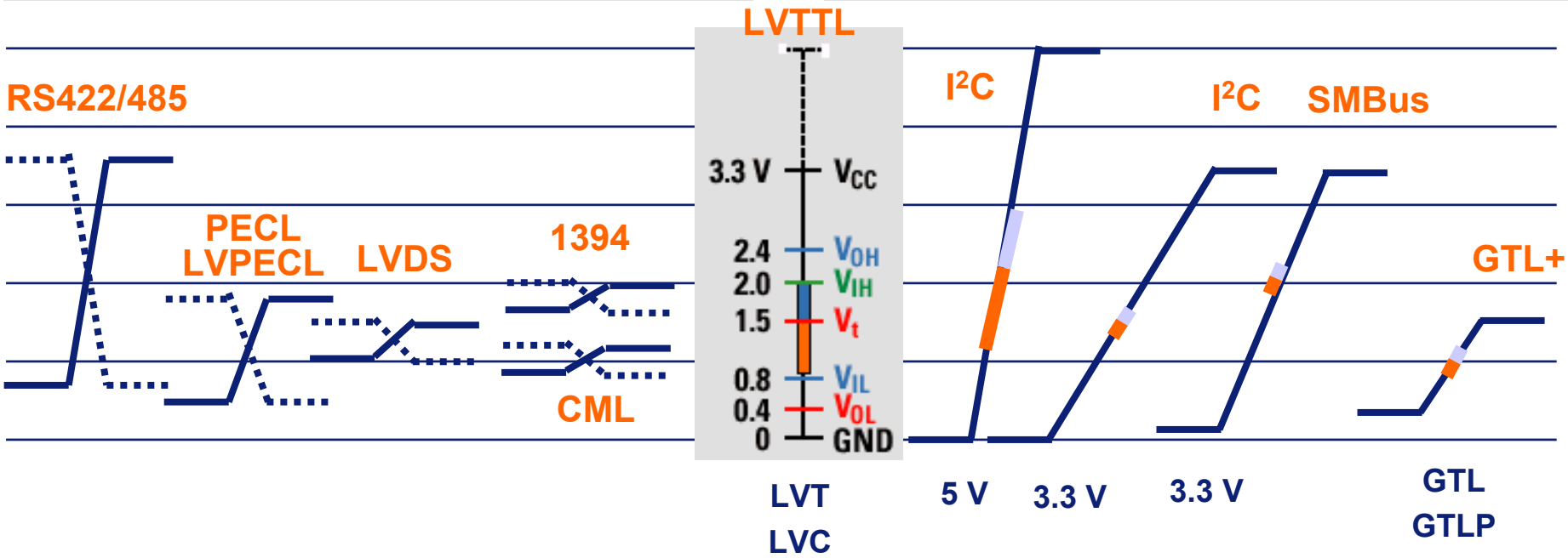


- A point to point communication does not require a Select control signal
- An asynchronous communication does not have a Clock signal
- Data, Select and R/W signals can share the same line, depending on the protocol
- Notice that Slave 1 cannot communicate with Slave 2 or 3 (except via the 'master')
  Only the 'master' can start communicating. Slaves can 'only speak when spoken to'

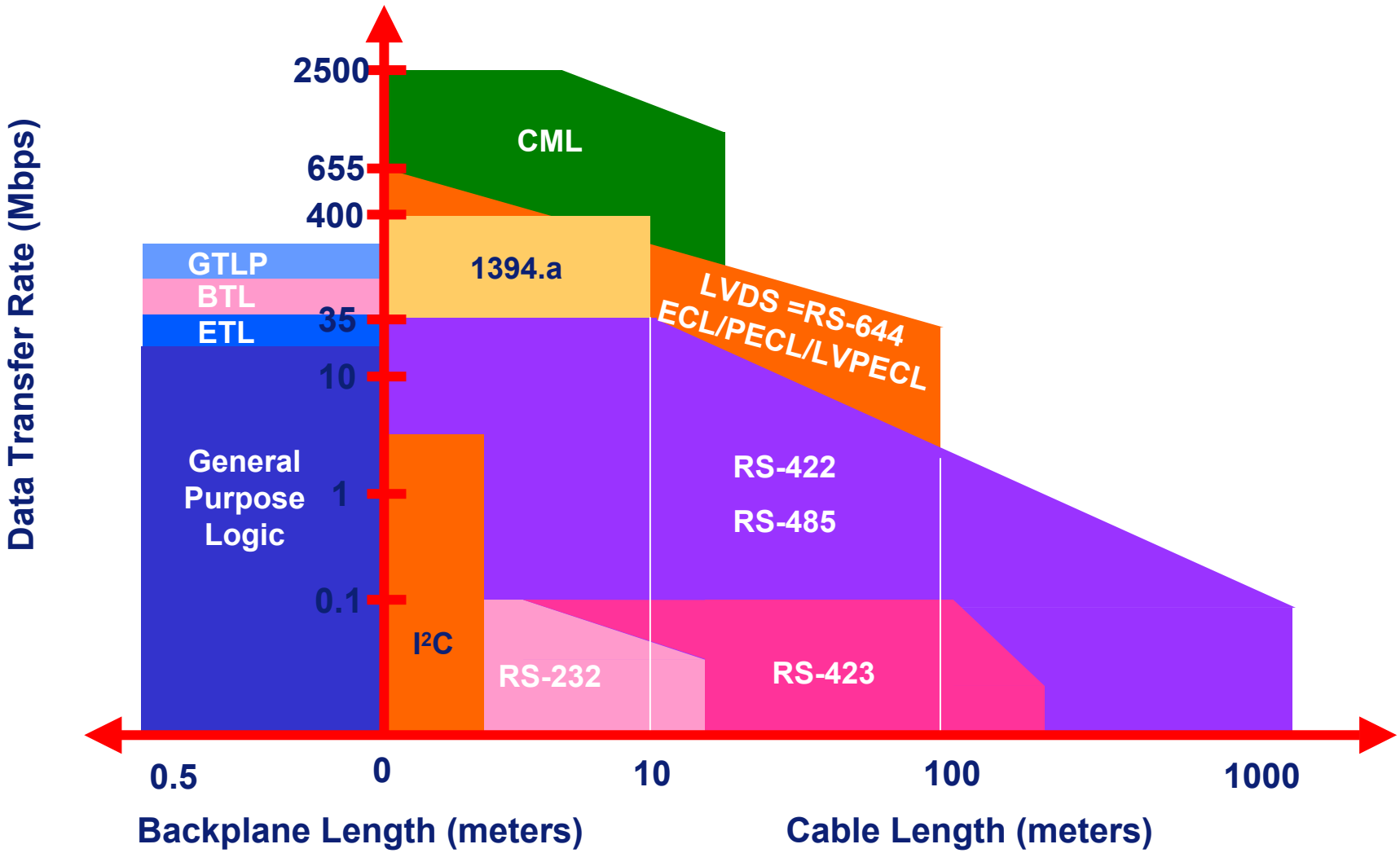
Typical Signaling Characteristics







Transmission Standards



Speed of various connectivity methods (bits/sec)

CAN (1 Wire)	33 kHz (typ)
I^2C ('Industrial', and SMBus)	100 kHz
SPI	110 kHz (original speed)
CAN (fault tolerant)	125 kHz
I^2C	400 kHz
CAN (high speed)	1 MHz
I^2C 'High Speed mode'	3.4 MHz
USB (2.0, Original Speed)	1.5 MHz or 12 MHz
SCSI (parallel bus)	40 MHz
Fast SCSI	8-80 MHz
Ultra SCSI-3	18-160 MHz
Firewire / IEEE1394	400 MHz
Hi-Speed USB (2.0, Hi-Speed)	480 MHz

Bus characteristics compared

Bus	Data rate (bits / sec)	Length (meters)	Length limiting factor	Nodes Typ.number	Node number limiting factor
I²C	**400k**	**2**	w iring capacitance	**20**	400pF max
I²C w ith buffer	**400k**	**100**	propagation delays	**any**	no limit
I²C high speed	**3.4M**	**0.5**	w iring capacitance	**5**	100pF max
CAN 1 w ire	**33k**	**100**	total capacitance	**32**	load resistance and transceiver current drive
CAN differential	**5k**	**10km**	propagation delays	**100**	
	125k	**500**			
	1M	**40**			
USB (2.0, Low speed)	**1.5M**	**25**	5 cables linking 6 nodes (5m cable node to node)	**127**	bus specs
USB (2.0, full -speed)	**12M**	**25**	5 cables linking 6 nodes (5m cable node to node)	**127**	bus and hub specs
Hi-Speed USB (2.0, Hi-Speed)	**480M**				
IEEE-1394	**100 to 400M+**	**72**	16 hops, 4.5M each	**63**	6-bit address

UART Overview

What is UART ?

- UART = Universal Asynchronous Receiver/Transmitter
- Asynchronous means that the data clock is not included in the data.
- “Start” and “Stop” bits indicates the data to be sent
- Old communication standard implemented in the 60’s.
- Simple, universal, well understood and well supported.
- Very slow speed communication standard

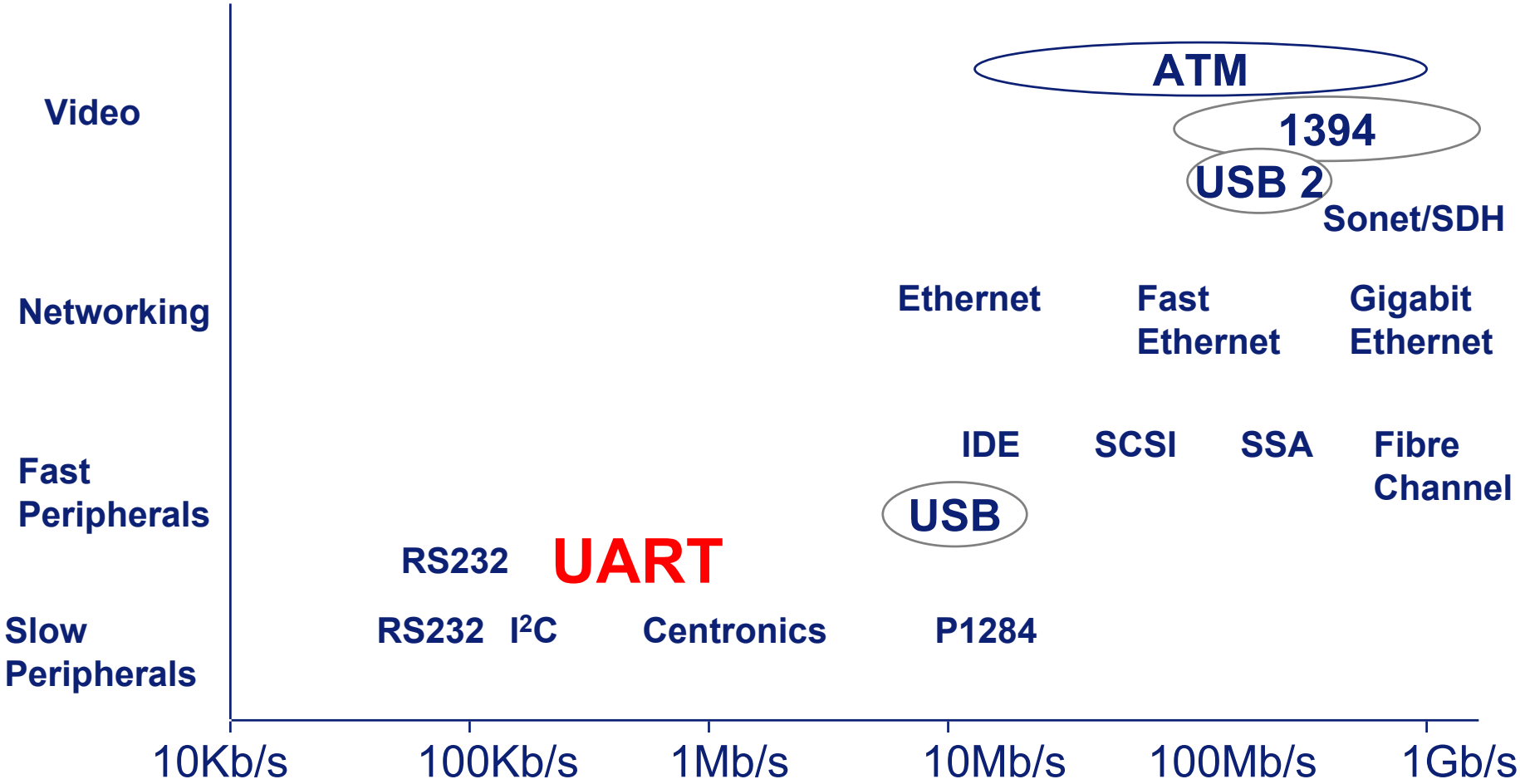
Where to find UART applications

- Everywhere were one needs a data connection between appliances UARTs could be a viable option
  - Any place where two or more systems (computers, alarm systems, engine control ...) must communicate.
  - Despite competing protocols (USB, CAN, Fieldbus, Ethernet ..) UARTs are often the products of choice, because of *simplicity* (2 wire),*legacy, product long life cycle and cost-effectiveness.*
  - Wherever micro-processors/controllers are used there is a good probability a UART channel exists.The more powerful the processor the higher probability that multiple datacom channels exist.
  - Even though the main processing system does not have a discrete UART on board the device the processing is communicating with may have one.

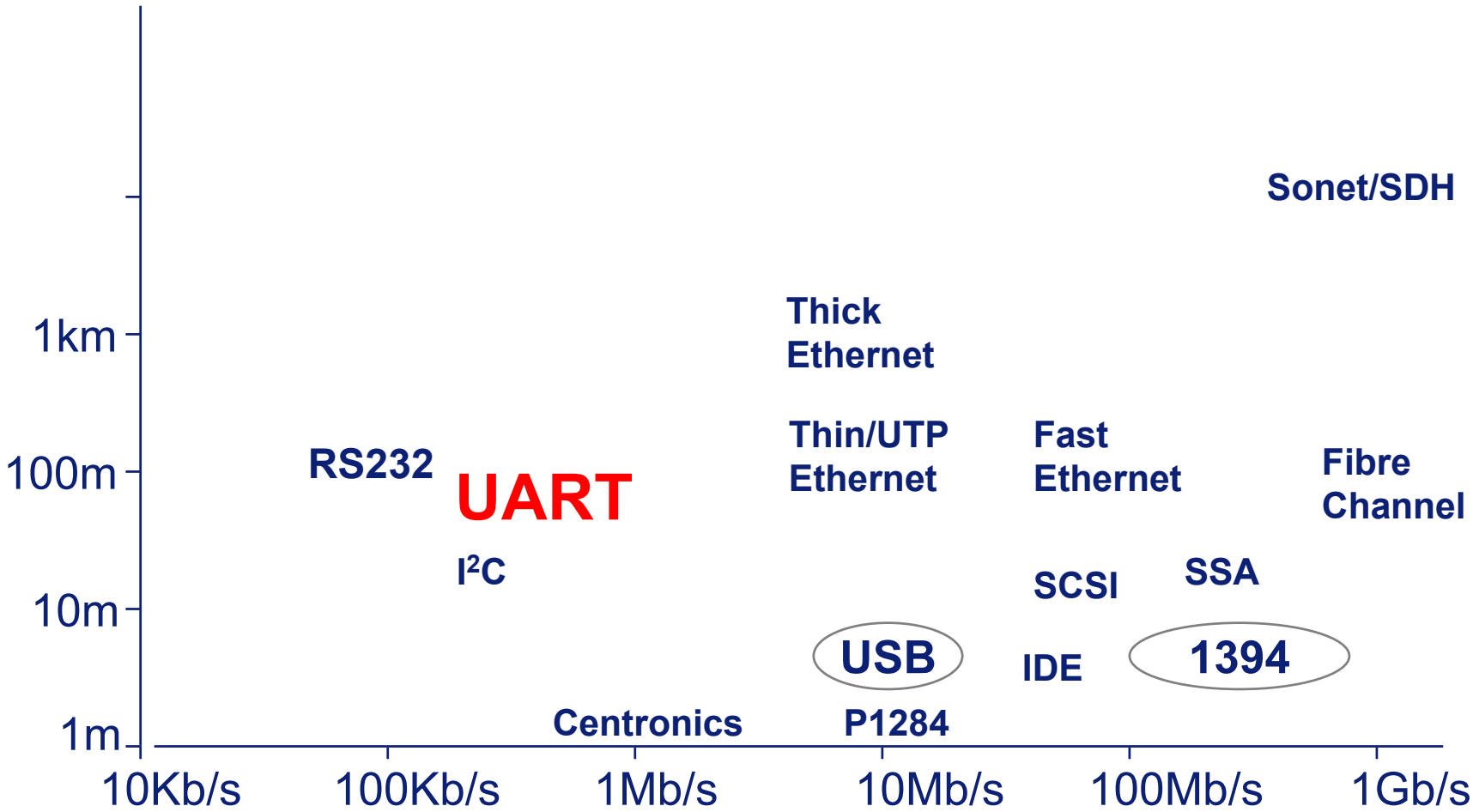
Where to find UART applications

- Integrated (micro)controllers still need external UARTs
  - Generic micro controllers provide one or two UARTs on-chip. Many systems require more serial channels and UARTs may have to be added
  - Datacom controllers such as the MC6836x controllers provide several serial interfaces but not enough for functions such as concentrators. Many datacom controllers must add external UARTs and USARTs

Applications and speed



Speed versus distance



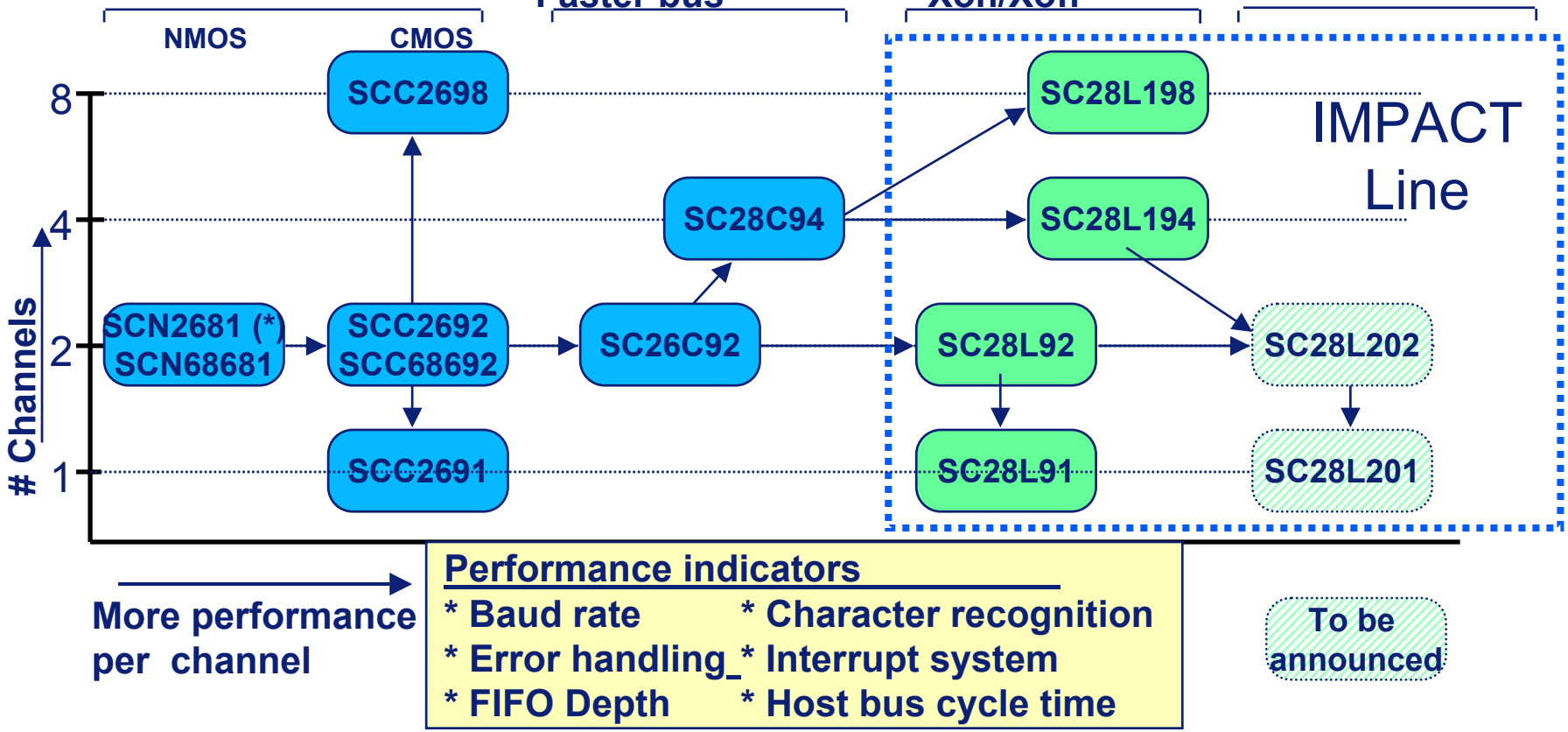
Industrial UART Overview

First Generation
- Basic Feature Set
- 3 deep Rcv FIFO's

Second Generation
- 9 deep FIFO's
- Higher baudrates
- Faster bus

Third Generation
- 16 deep FIFO's
- 3V- 5V Operation
- Xon/Xoff

Next Generation
- 256 deep FIFO's
- 3V- 5V Operation



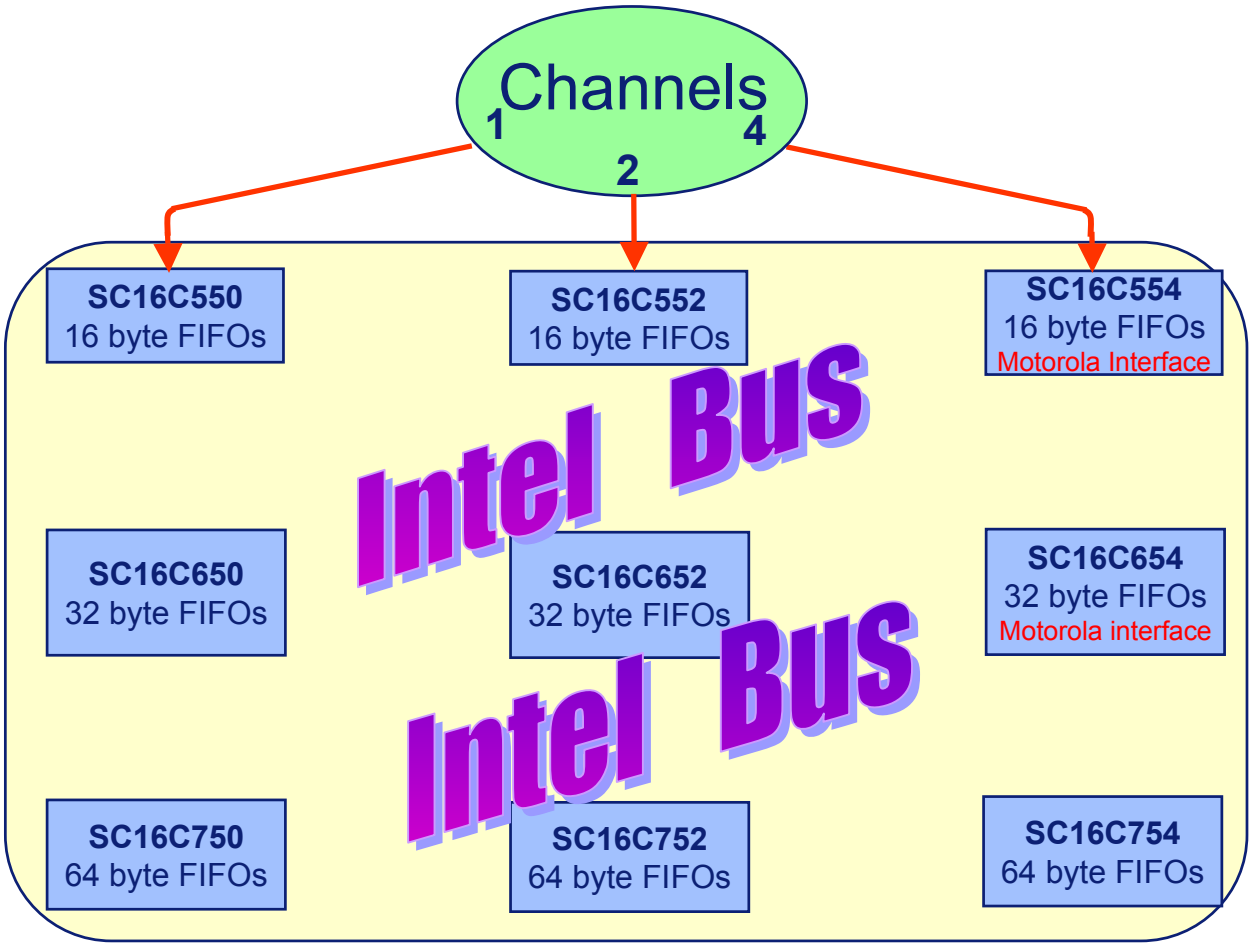
Note : (*) SCNxxx not recommended for new designs

Features & Benefits Industrial UARTs

Feature	Benefit
• Broad Line of single to octal UARTs	• Variety of choices for different applications
• Power down mode	• Ideal for low power consumption
• Extensive interrupt support	• Reduced SW overhead
• HW RS485/Multi-drop support	• Reduce CPU overhead
• Automatic Out band flow control	• Avoids loss of data
• All RX/TX are independent with respect to speed and clock frequency	• RX/TX can operate at different baud rates
• Flexible and programmable I/O structure	• Allows usage of I/O pins for general purposes
• **Impact Line Series**: – All of the above – 3.3V and 5V operations – Motorola and Intel interface – Industry Temp (-40 +85 ºC)	• Single part can be used for multiple operating environment. Significantly lower cost of ownership

Philips SC16Cxxx UARTs

All: CMOS, 2.5 V, 3.3 V to 5 V, Very Fast bus cycle times, Baud rates up to 5 Mbps, Industrial Temp; Pin Compatible to current devices in the market



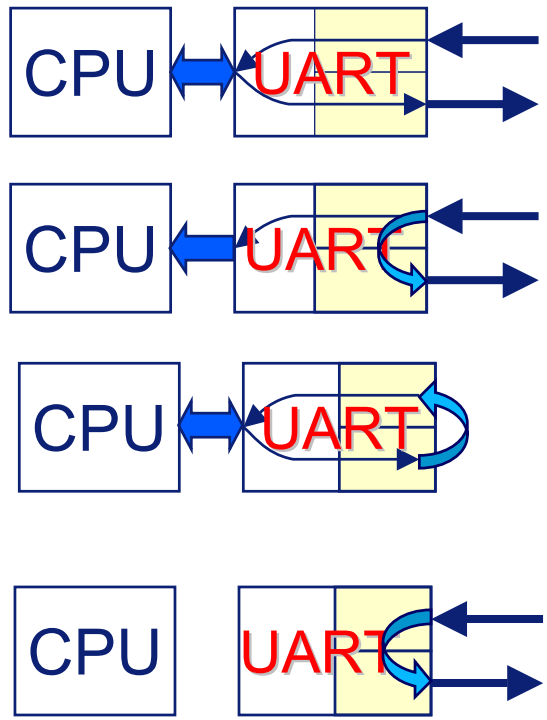
Feature and Benefit of Philips SC16Cxxx UARTs

Feature	Benefit
• Broad Line of single to quad UARTs • All devices operate at 2.5 V, 3.3 V and 5 V and specified at Industrial Temp (-40 to 85 ºC).	• One stop shopping • Single part can be used for multiple operating environment. Significant lower cost of ownership compared to competition
• Fastest bus cycle times (> 25%) and baud rates up to 5 Mb/s	• Compatible with high speed processors
• Power down mode • IRDA features • SW readily available • Auto. SW and HW flow control • DMA mode + wide variety of FIFO depth • Pin to pin compatible to existing 16C devices	• Ideal for battery operated systems • Allows wireless short range applications • Shortens design cycle • Reduce CPU overhead and loss of data • Reduces CPU overhead • Alternative source to other manufacturers

Meaning & Benefits of Philips UART Features

Programmable channel mode

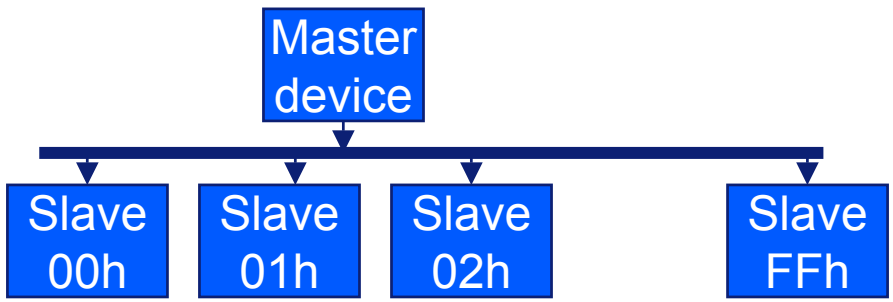
- Four Programmable channel modes
  - Normal, Full duplex
  - Automatic echo
    - Received characters are retransmitted
  - Local loop back (diagnostics and tests)
    - Transmitter output is internally connected to receiver
  - Remote loop back (diagnostics and tests)
    - Received characters are retransmitted (as automatic echo)
    - Received characters are NOT send to local CPU

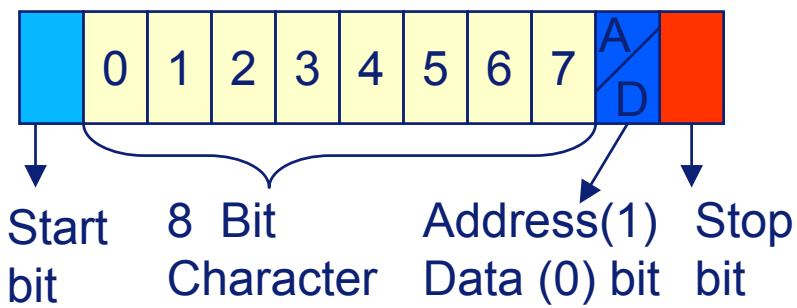


Meaning & Benefits of Philips UART Features

Multi-drop mode

- Multi-drop mode (also called “wake up” and 9 bits mode)
  - Parity bit is used as Address or Data indicator
  - In multi-drop mode slave only accept data after a valid address is received
  - Wake-up if an address enters the receiver
  - In sleep all characters with the A/D bit set to 0 are discarded
  - UARTs can be programmed to disable transmitter after end of last character

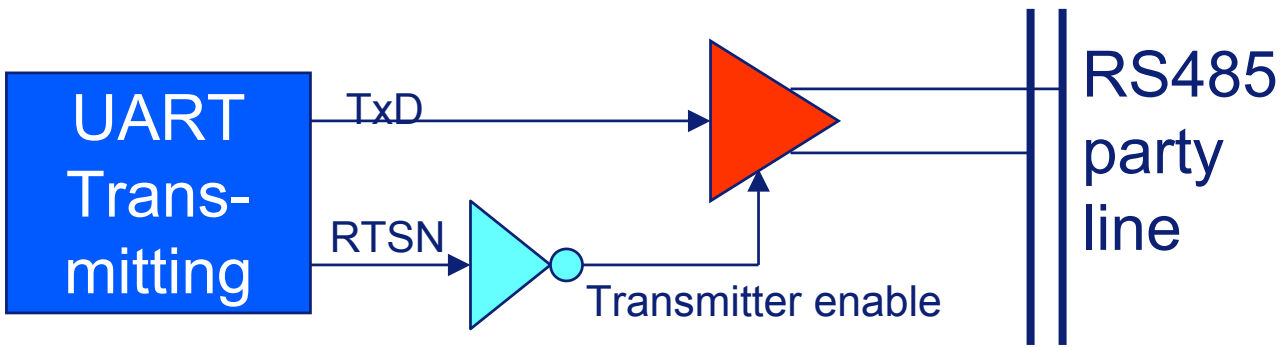




Meaning and Benefits of Philips UART Features

RS485 support

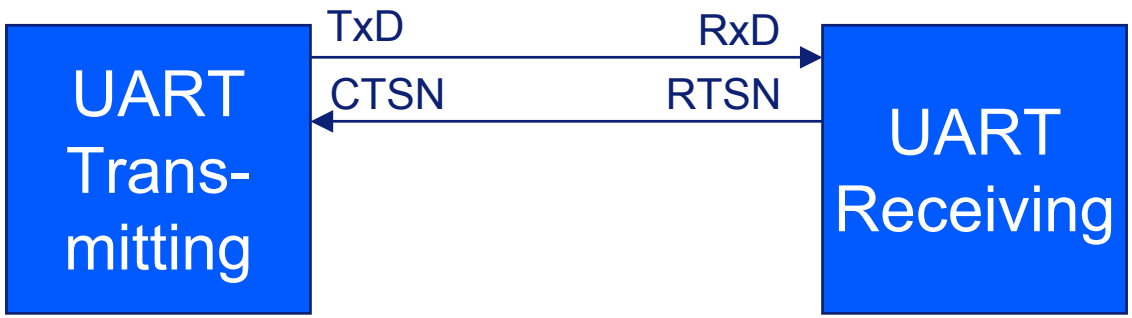
- RS485 requires that the line driver (transmitter) is disabled as soon as possible after the message has been send
- One of the outputs (RTSN) can be programmed to this automatically one bit time after the last bit of the message has been send
- Advantage: reduction of software overhead
  - Note: in this mode RTSN has a different meaning than when used for hardware handshake



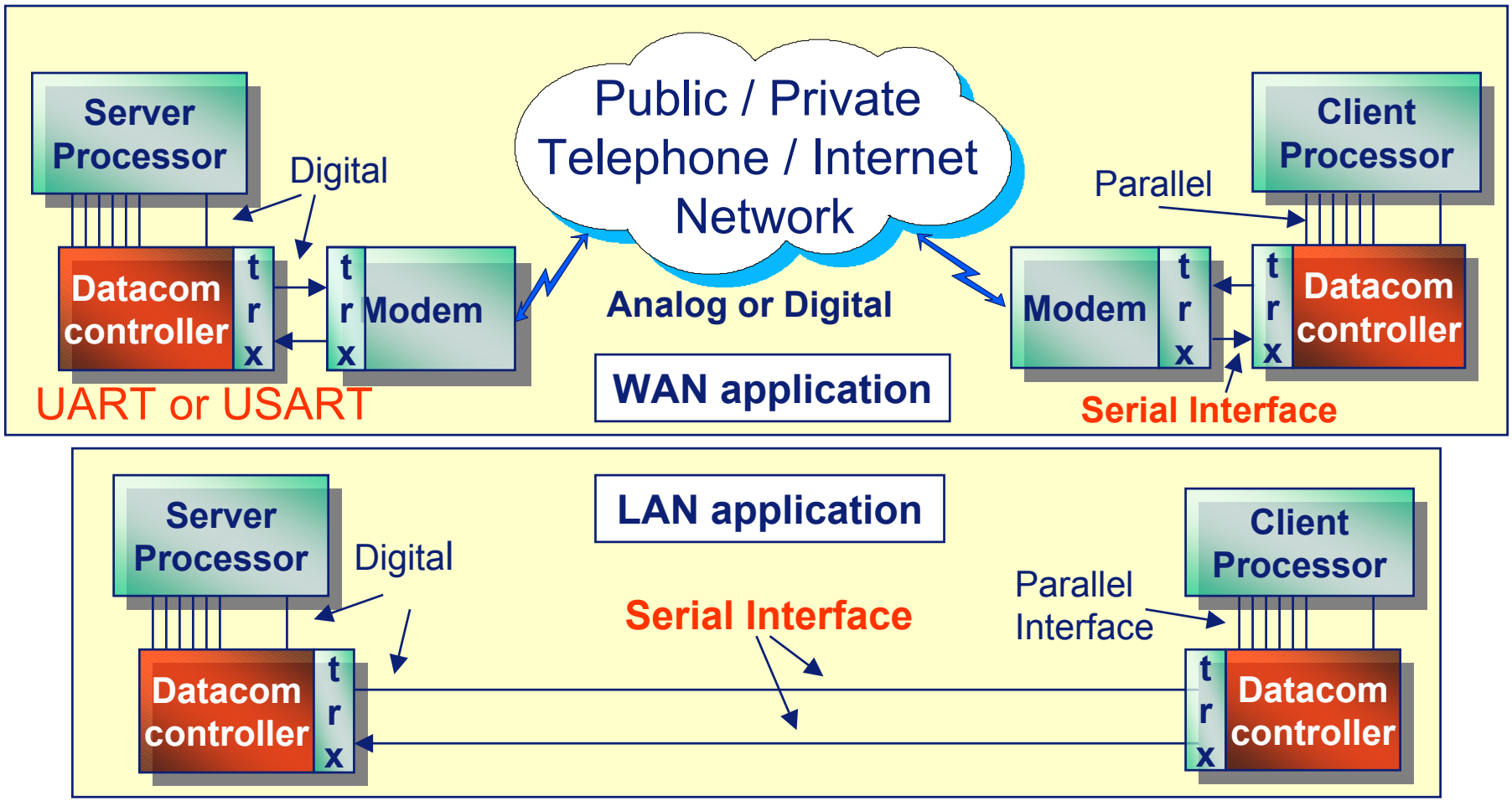
Meaning and Benefits of Philips UART Features

Hardware Handshaking

- Hardware handshaking is used to control the data stream between UARTS by means of Modem control signals
  - Receiver asserts RTSN if its FIFO is full
  - Receiver negates RTSN if a position in FIFO is available
  - Transmitter stops transmission if CTSN is asserted
- Automatic hardware handshaking prevents FIFO overflow without CPU intervention

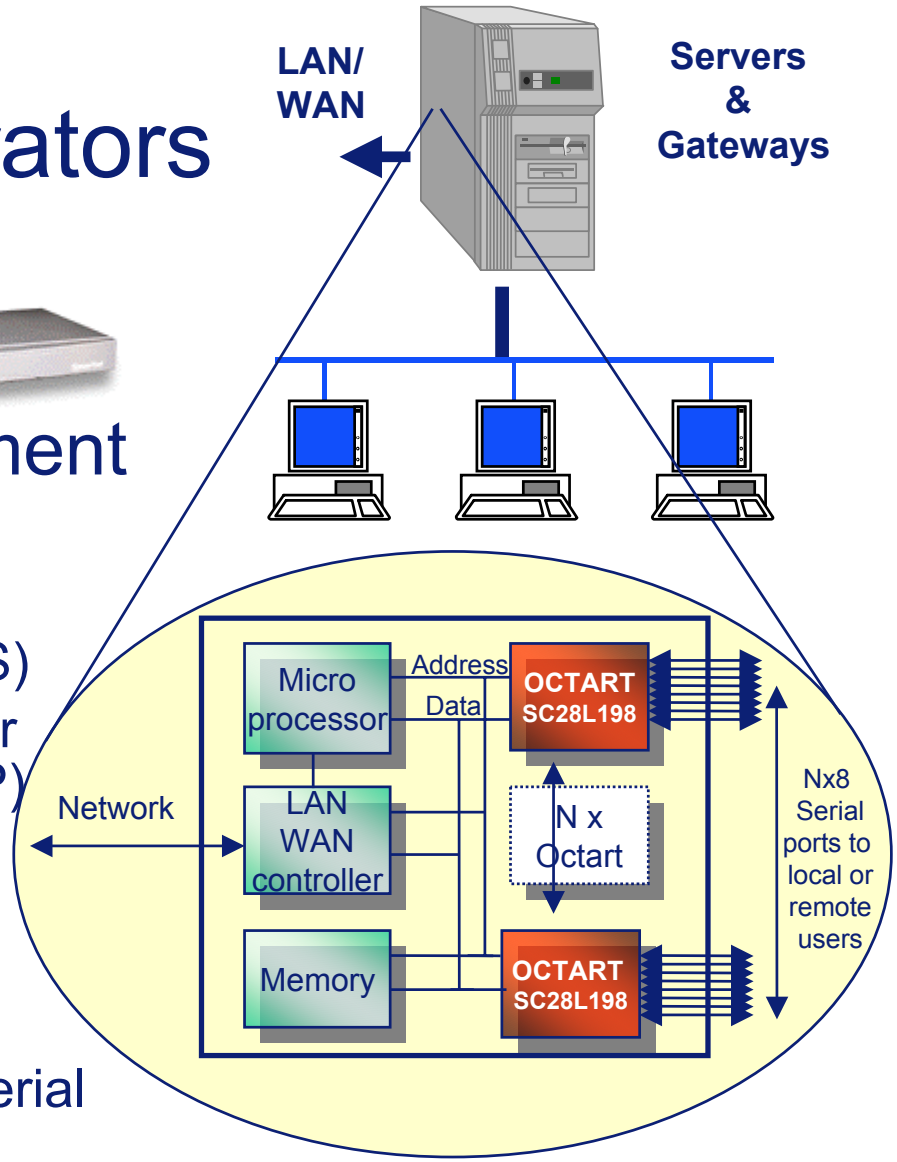


Generic Datacom applications



Servers & Concentrators

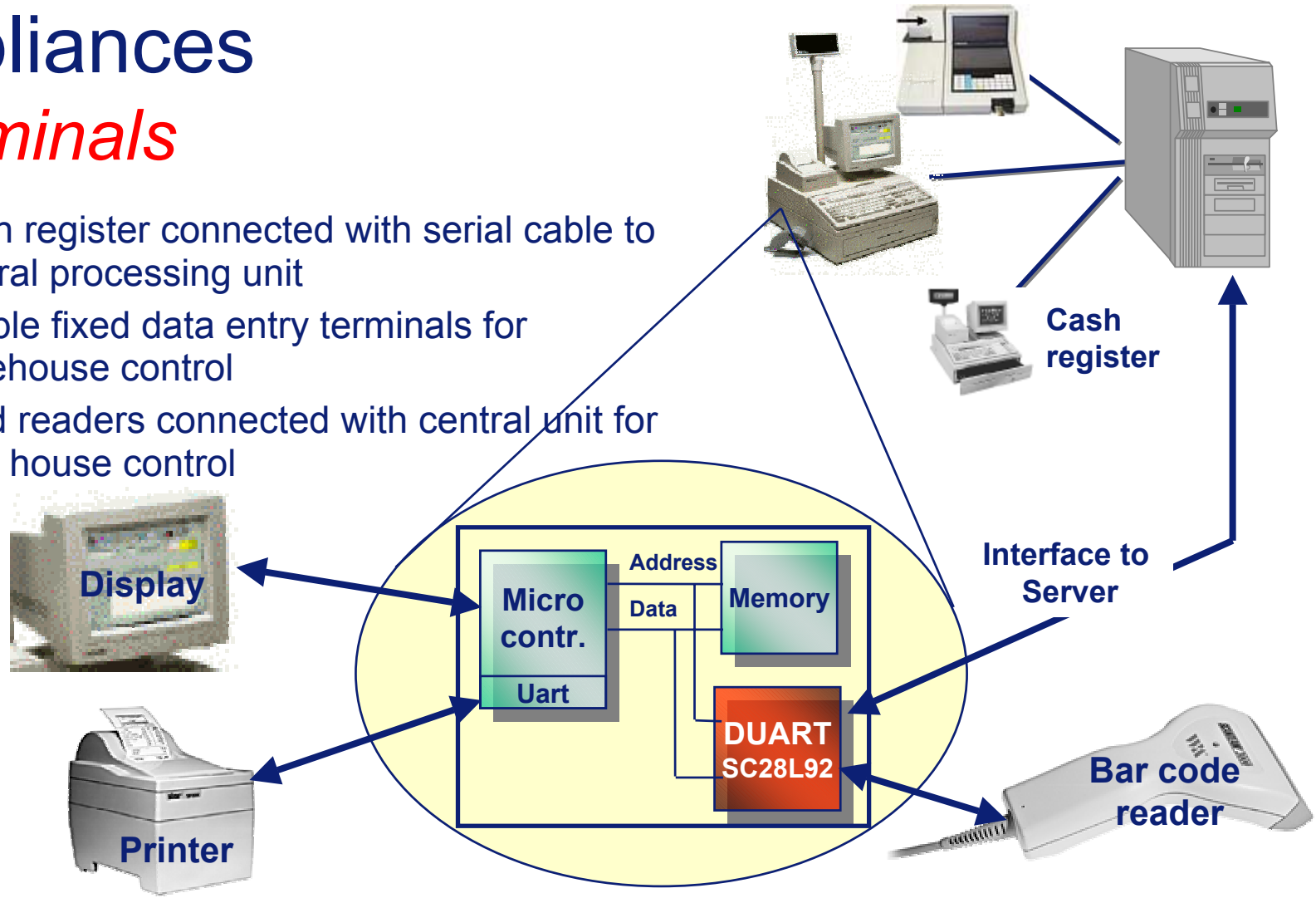
- Internet Access Equipment
  - Routers
  - High-end modems
  - Remote access service (RAS)
  - Modem access equipment for Internet service provider (ISP)
- Industrial
  - Warehouse control
  - Point of Sales terminals
  - Equipment control through serial connection
  - Remote measurement



Appliances

Terminals

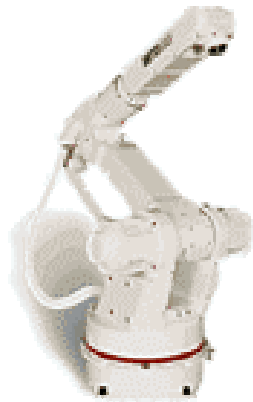
- Cash register connected with serial cable to central processing unit
- Simple fixed data entry terminals for warehouse control
- Card readers connected with central unit for park house control



More Application: Entertainment, Home, Security, Robotics & More…...

- Entertainment
  - Midi interface on musical devices and Stage equipment
  - Data exchange and control of gambling equipment
  - Toys (Lego®, etc.)

- Home, Security
  - Remote control of audio equipment from central unit
  - Light and heating control devices in homes or offices
  - Security sensors and surveillance devices
- Robotics
  - Industrial control of CNC equipment
  - Remote sensor equipment
  - Motor control

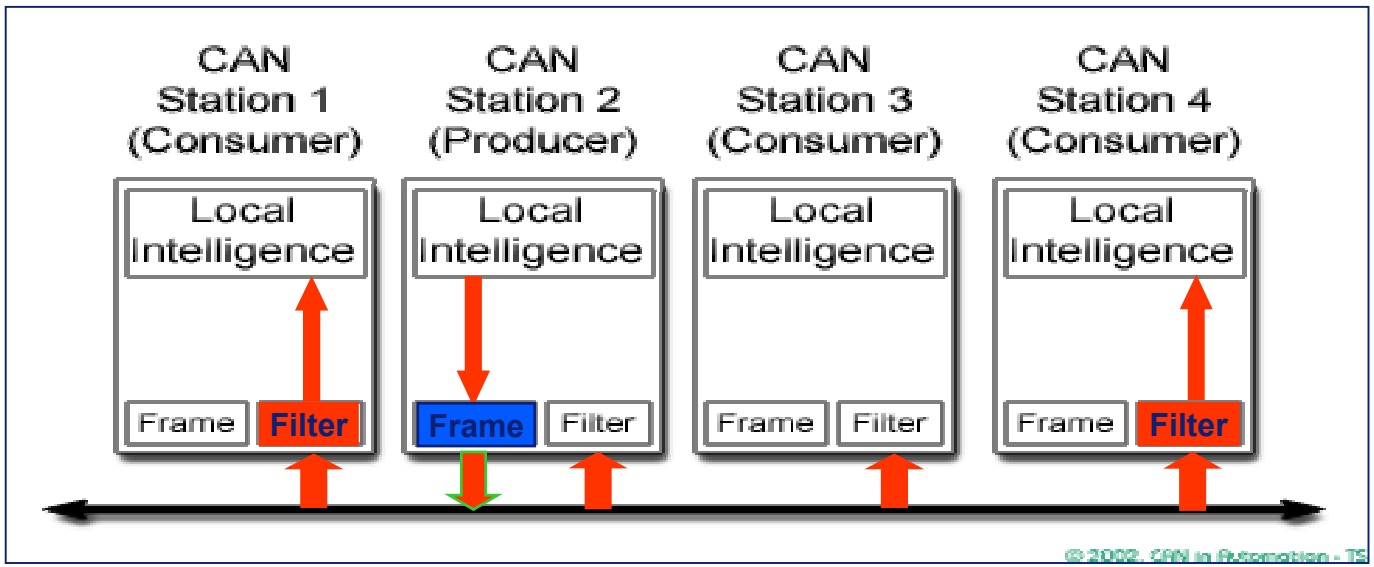


- Automotive, Cellular, Medical and Much More!

CAN Overview

What is CAN ? (Controller Area Network)

- Relatively complex coding of the messages
- Relatively accurate and (usually) fixed timing
- All modules participate in every communication
- Proposed by Bosch with automotive applications in mind (and promoted by CIA - of Germany - for industrial applications)



CAN protocol

Start Of Frame

Identifier

Remote Transmission Request

Identifier Extension

Data Length Code

Data

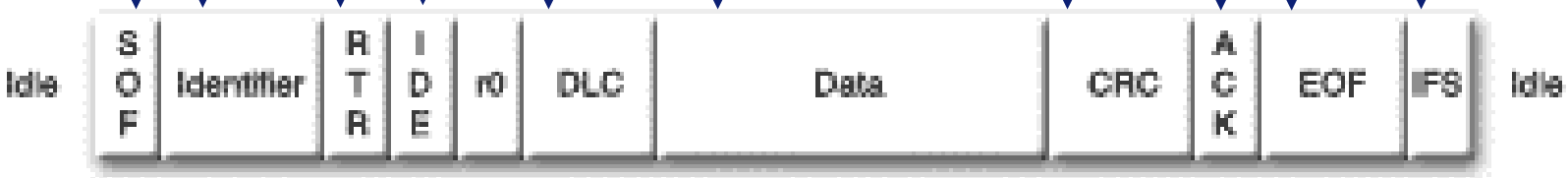
Cyclic Redundancy Check

Acknowledge

End Of Frame

Intermission Frame

Space



CAN Bus Advantages

- Accepted standard for Automotive and industrial applications
- Freedom to select suitable hardware
  - differential or one wire bus
  - very ‘long haul’ possible for industrial applications
- Secure communications
  - 15 bit CRC in messages (Cyclic Redundancy Check)
  - Reporting/logging of errors
  - Differential bus can be ‘fault tolerant’ e.g. to loss of one wire
  - Faulty devices can disconnect themselves
- High degree of EMC immunity (when using Si-On-Insulator technology)
  - Transients, short circuits to supply/ground, ‘Load dump’

How are connected devices recognised

- All devices receive, decode, and acknowledge all bus messages.
  - Each bus controller is programmed to ‘filter’ the data to decide whether a message is intended for its use.
- Any number of devices can act upon the same message.
  - And any / all devices will report any errors in received data
- All devices use fixed and accurate bus timing.
  - Needs a ceramic resonator or possibly a crystal
  - So primarily not intended to be a ‘plug & play’ bus

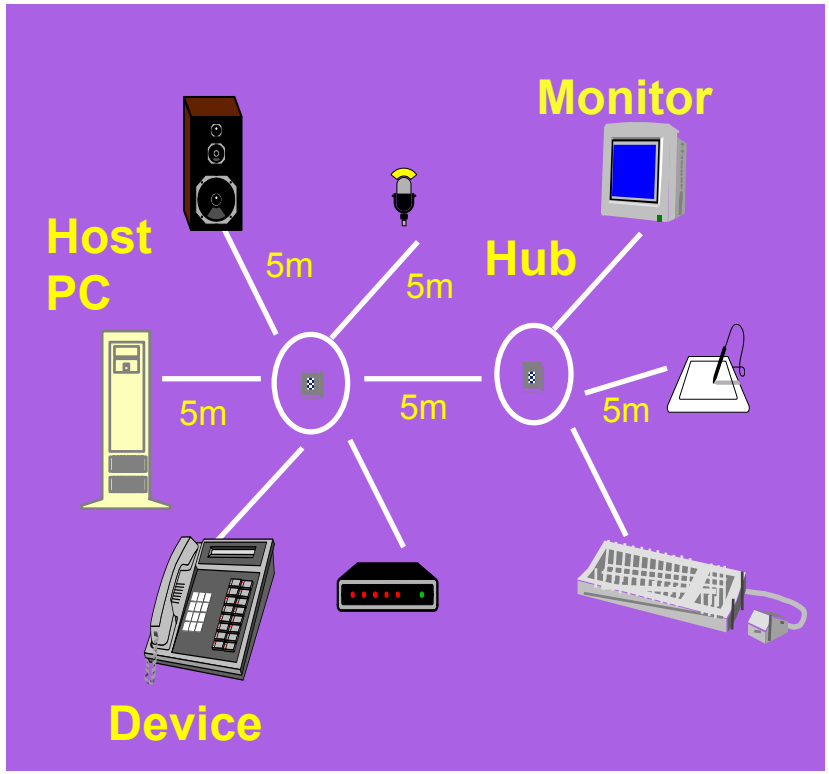
USB Overview

What is USB ? (Universal Serial Bu

- Originally a standard for connecting PCs to peripherals
  - Philips one of 7 core companies that defined it
  - Intended to replace the large number of legacy ports in the PC
  - Single master (=Host) system with up to 127 peripherals
  - Is simple plug and play; no need to open the PC
  - Standardized plugs, ports, cables
  - Has over 99% penetration on all new PCs
    - Serial ports NOT on 2003 PC's
  - New On-The-Go Supplement eliminates PC!
    - Allows dynamic exchanging of Host/Slave roles
    - PC is no longer the only system Host. Can be a camera, cell phone, printer, PDA, etc.

USB Topology (original concept, USB1.1, USB2.0)

- **Host**
  - One PC host per system
  - Provides power to peripherals
- **Hub**
  - Provides ports for connecting more peripheral devices.
  - Provides power, terminations
  - External supply or Bus Powered
- **Device, Interfaces and Endpoints**
  - Device is a collection of data interface(s)
  - Interface is a collection of endpoints (data channels)
  - Endpoint associated with FIFO(s) - for data I/O interfacing



USB A Connector

USB B Connector

Advantage of USB

- Convenient link between PC and wide range of peripherals
  - printer, scanner, digital cameras, speakers, monitor, keyboard, mouse, etc.
- Push for USB to become THE standard on PCs
  - standard for IBM compatible, (recently) Apple, now on > 99% of PCs
- Interfaces (bridges) to other comms channels exist
  - USB to serial port (serial port vanishing from laptops)
  - USB to IrDA or to Ethernet
  - USB to Bluetooth
- Extreme volumes force down IC and hardware prices

Versions of USB specification

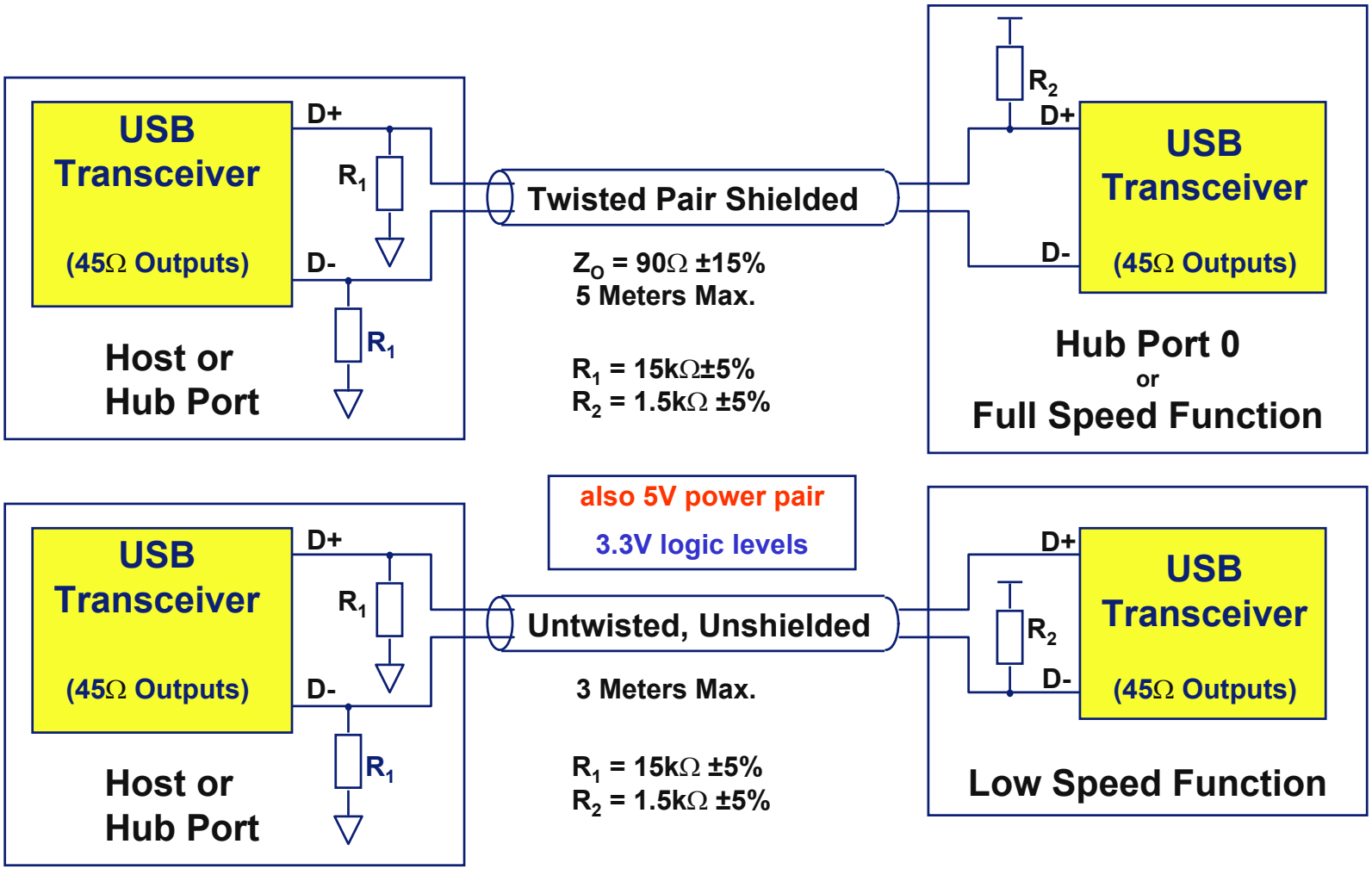
- **USB 1.1**
  - Established, large PC peripheral markets
  - Well controlled hardware, special 4-pin plugs/sockets
  - 12 MBits/sec (normal) or 1.5 Mbits/sec (low speed) data rate
- **USB 2.0**
  - Supercedes 1.1 specification
    - Backwards compatible
    - Encompasses LS, FS, HS
  - Introduces smaller “mini-B” connector
  - Challenging IEEE1394/Firewire for video possibilities
  - 480 MHz clock for Hi-Speed means it’s real “UHF” transmission
  - Hi-Speed option needs more complex chip hardware and software
  - Hi-Speed component prices x3 approx. compared to full speed
- **USB “OTG” (On The Go) Supplement**
  - **Supplement to the 2.0 specification**
  - Allows peripherals to “talk” to each other
  - Dynamic host/peripheral switch
  - Introduces new “mini-AB” receptacle and “mini-A” plug
  - Lower power requirements

How are the connected devices recognised ?

- USB is essentially a ‘one point to one point’ physical connection
  - The information flow is multiplexed and ‘repeated’ by ‘hubs’
  - ‘T’ connectors and extension cables are not permitted
- Balanced (differential) bus is unbalanced by resistors in a device to signal a connection has been made, and set Full/Low speed
  - Resistors from bus wires to supply or ground are detected
- Host (= Master = PC usually) asks device to identify itself
- Device sends specific identification
  - Host / PC needs to have correct software to communicate
  - Hi-Speed devices send extra signals during the reset period requesting upgrade to high speed mode

USB hardware

Connections and Terminations



1394 Overview

What is IEEE1394 ?

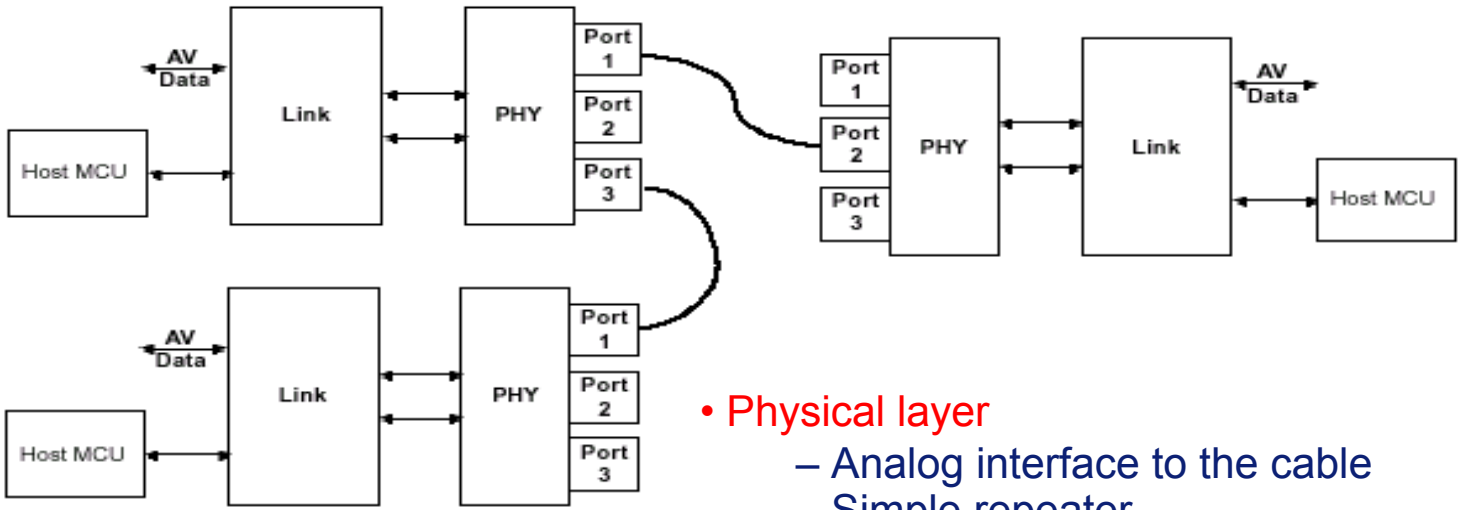
- A bus standard devised to handle the high data throughput requirements of MPEG-2 and DVD
  - Satisfies video transmission requirements by offering constant transfer rates with guaranteed bandwidth
  - Data rates 100, 200, 400 Mbits/sec and looking to 3.2 Gb/s
  - Isochronous data transfers
    - One to One, One to Many
    - Bandwith Management
  - Asynchronous data transfers
    - Error checking and retransmission
  - Managed bus architecture
    - Bus Manager
    - Isochronous Resource Manager

IEEE1394 Features

- Peer to Peer Network Architecture
  - Point to Point signaling environment
  - Automatically re-configures itself as each device is added
  - True plug & play
  - Hot-plugging of devices allowed
  - Up to 63 devices, 4.5M cable ‘hops’, with max. 16 hops
  - Bus wiring uses two shielded twisted pairs - data and strobePowering pair may be included with signal pairs
  - Specified connectors (same each end)
- Content Protection
  - Digital Transmission Content Protection (DTCP / 5C) Standard

1394 Topology

Multiple Nodes interconnected with a multiple twisted-pair cable



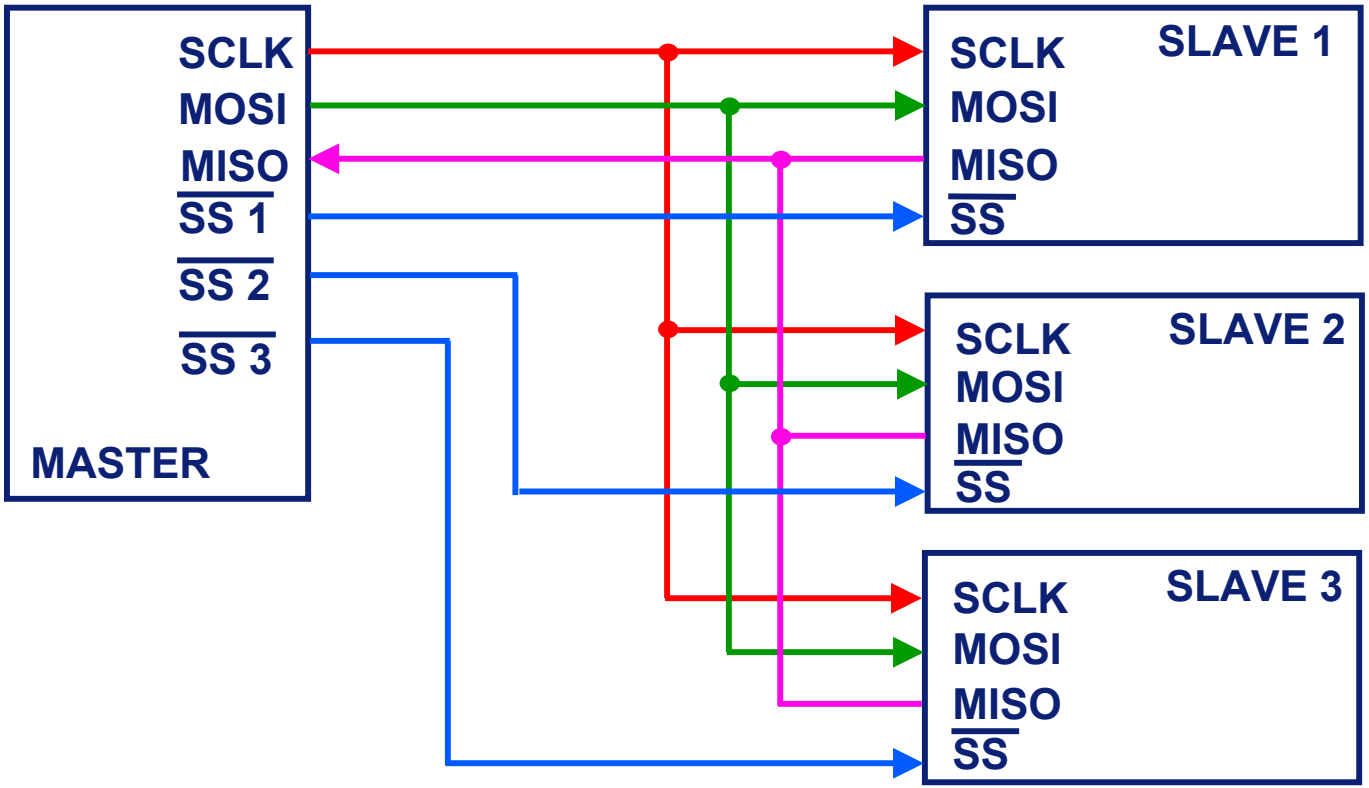
- Physical layer
  - Analog interface to the cable
  - Simple repeater
  - Performs bus arbitration
- Link layer
  - Assembles and dis-assembles bus packets
  - Handles response and acknowledgment functions
- Host controller
  - Implements higher levels of the protocol

SPI Overview

What is SPI?

- Serial Peripheral Interface (SPI) is a 4-wire full-duplex synchronous serial data link:
  - SCLK: Serial Clock
  - MOSI: Master Out Slave In - Data from Master to Slave
  - MISO: Master In Slave Out - Data from Slave to Master
  - SS: Slave Select
- Originally developed by Motorola
- Used for connecting peripherals to each other and to microprocessors
- Shift register that serially transmits data to other SPI devices
- Actually a “3 + n” wire interface with n = number of devices
- Only one master active at a time
- Various Speed transfers (function of the system clock)

SPI - How are the connected devices recognized?



- Simple transfer scheme, 8 or 16 bits
- Allows many devices to use SPI through the addition of a shift register
- Full duplex communications
- Number of wires proportional to the number of devices in the bus

I²C Overview

What is I²C ? (Inter-IC)

A set of specifications to build a universal bus guaranteeing compatibility of parts (ICs) from different manufacturers:

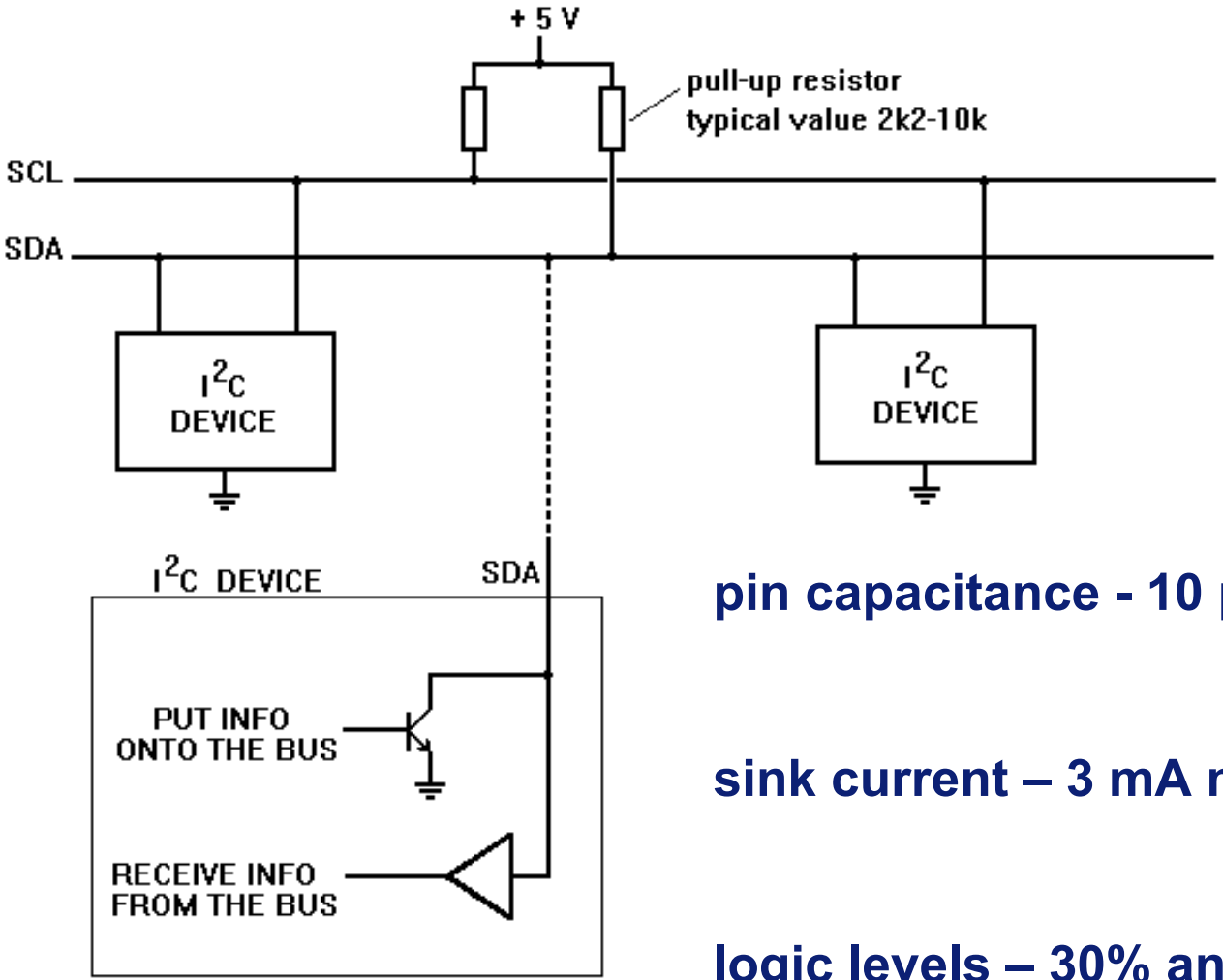
- Simple hardware standards
- Software protocol standard - not complex

No specific wiring or connectors - most often it's just PCB tracks

- Buffers like P82B96 adapt it to long cables also (30 m, 300 m, etc)

Has become a recognised standard throughout our industry and is used now by ALL major IC manufacturers

I²C 'hardware'



pin capacitance - 10 pF maximum

sink current – 3 mA minium

logic levels – 30% and 70% of V_{DD}

I²C Bus - Software

Simple procedures that allow communication to start, to achieve data transfer, and to stop

- Described in the Philips protocol (rules)
- Message serial data format is very simple
- Often generated by simple software in general purpose micro
- Dedicated peripheral devices contain a complete interface

Each IC on the bus is identified by its own address code

The IC that initiates communication provides the clock signal (SCL)

- There is a maximum clock frequency but NO MINIMUM SPEED

I²C Bus - Information format

- The ‘bits’ are the data bits sent on the data wire (SDA)
- They are requested by, or supplied after, a falling SCL edge
- They are ‘clocked in’ by the rising edges on the clock wire (SCL)

Start bit	Slave Address bits	R/W bit	Ack bit	8 Data bits	Ack bit	8 Data bits	Ack bit	Stop bit

How are the connected devices recognised?

- Master device ‘polls’ -- checks for specific (Slave) parts with specific identification or “addresses” that the designer has included in the system
  - Allows for easy auto-configuration of product ‘optional extras’
- Devices with Master capability can identify themselves to other specific Master devices and advise their own specific address and functionality
  - Allows designers to build ‘plug and play’ systems
  - Bus speed can be different for each device, only a max. limit
- Only two devices exchange data during one ‘conversation’

Why choose I²C Bus ?

Simpler PCB layout - less tracks, less IC pins needed

Easy to test - just make a connection to the bus

Easy to offer different product features/options

Fast enough for all 'human interfaces'

-- simple displays, switches, keyboard

Reasonably high security of data

-- suits control or alarms systems

-- byte acknowledge, software improvement by requesting feedback

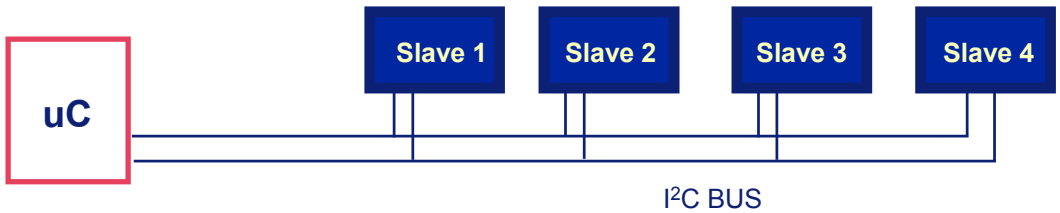
What doesn't it do?

VERY fast transfer of large amounts of data

-- e.g. computer LANs, video data streams

….. But a 3.4 Mbits/sec protocol version does exist!

What do I need to drive the I²C bus?



There are 4 basic ways to drive the I²C bus:

1) With a Microcontroller with on-chip I²C Interface
These come in two flavors:
Bit oriented - CPU is interrupted after every bit transmission
Byte oriented - CPU can be interrupted after every byte transmission

2) With ANY microcontroller: 'Bit Banging'
The I²C protocol can be emulated bit by bit via any bi-directional open drain port

3) Using another manufacturer's serial interface (software modified).
Example: using Motorola's SPI

4) With a microcontroller in conjunction with our PCF8584 or PCA9564 parallel to I²C bus interface IC

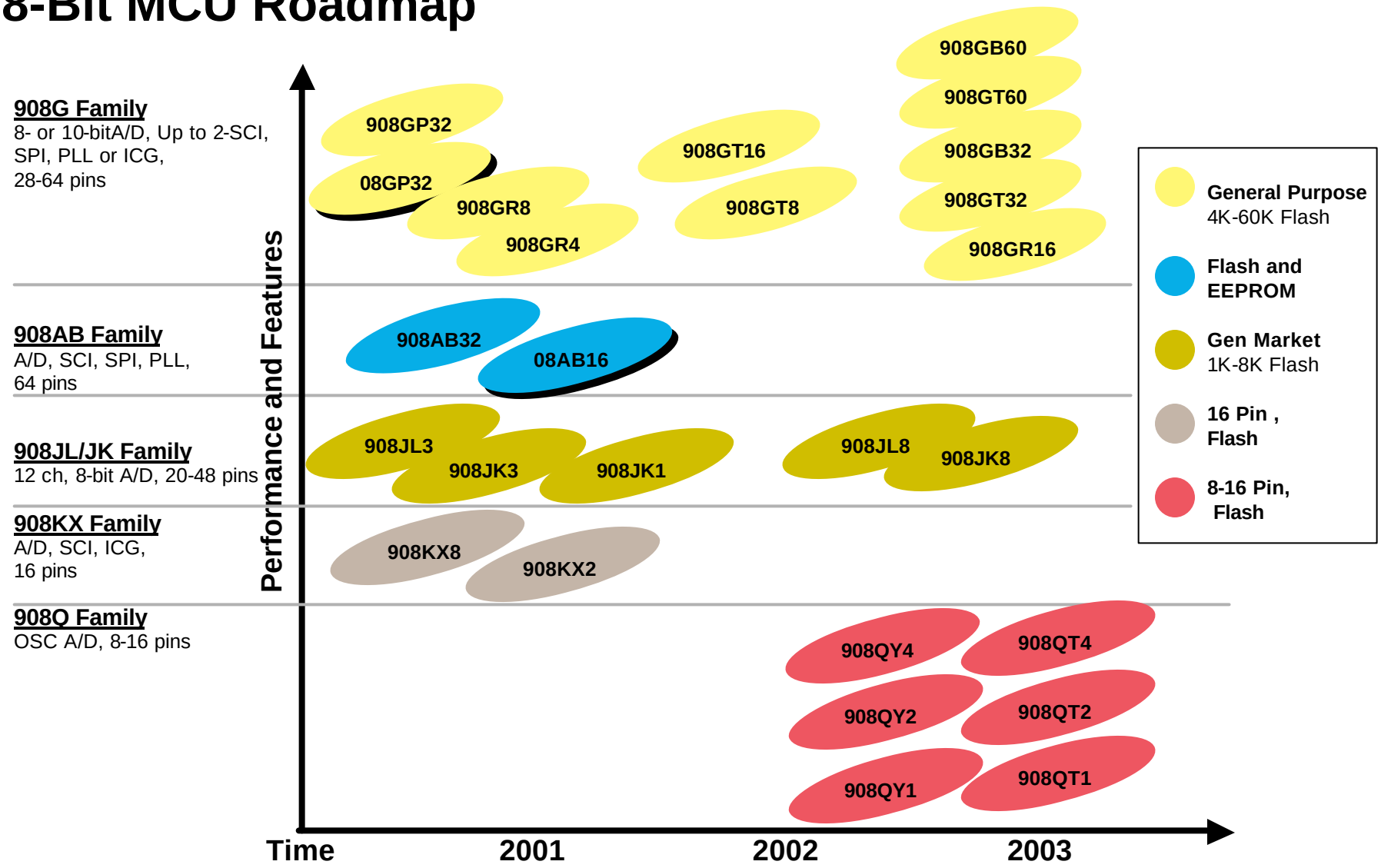
Serial Bus Summary

Pros and Cons of the different buses

UART	CAN	USB	SPI	I²C
• Well Known • Cost effective • Simple	• Secure • Fast	• Fast • Plug&Play HW • Simple • Low cost	• Fast • Universally accepted • Low cost • Large Portfolio	• Simple • Well known • Universally accepted • Plug&Play • Large portfolio • Cost effective
• Limited functionality • Point to Point	• Complex • Automotive oriented • Limited portfolio • Expensive firmware	• Powerful master required • No Plug&Play SW - Specific drivers required	• No Plug&Play HW • No “fixed” standard	• Limited speed


PHILIPS

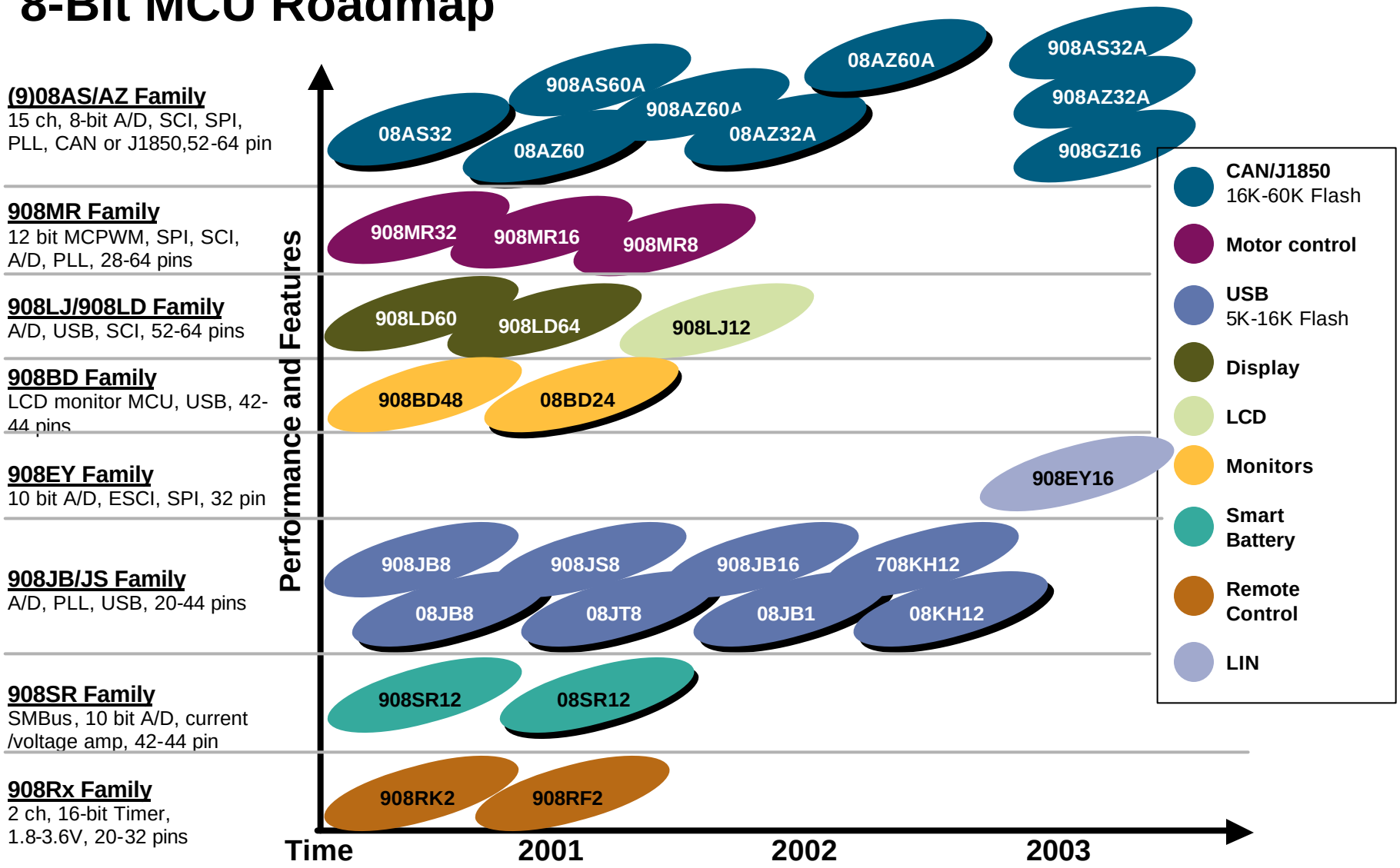
8-Bit MCU Roadmap



MOTOROLA intelligence everywhere™

digital dna™

8-Bit MCU Roadmap



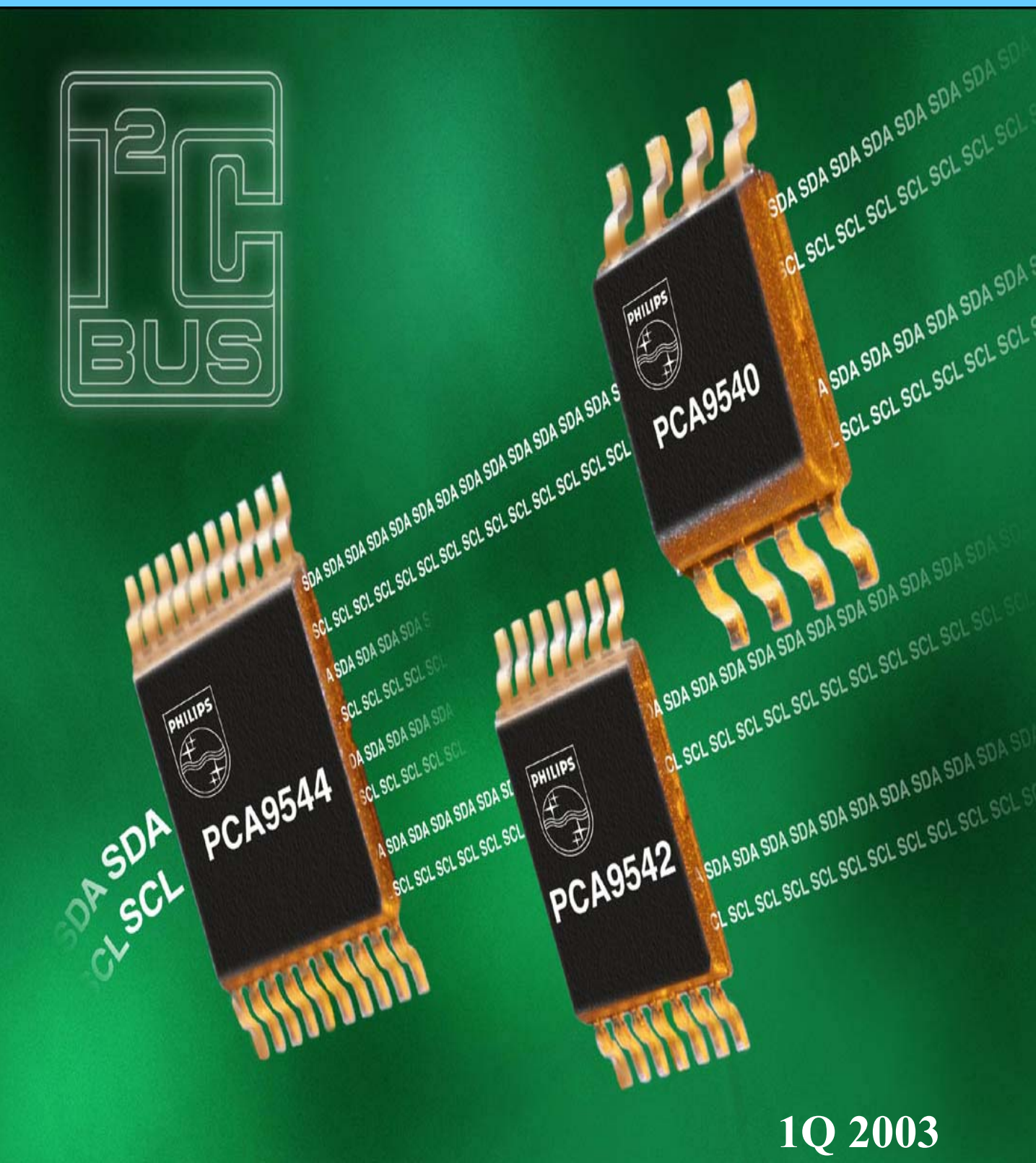
MOTOROLA intelligence everywhere™

digital dna™

Safe Harbor Announcement

Except for historical information, all of the expectations and assumptions, contained in the foregoing are forward-looking statements involving risk and uncertainties. Important factors that could cause actual results to differ materially from such forward-looking statements, include, but are not limited to, the competitive environment for our products, changes of rates of all related services, and legislation that may affect the industry. For additional information regarding these and other risks associated with Company's business refer to the Company's reports with the SEC.

Philips Semiconductors
I²C Handbook



1Q 2003

Philips I²C Products

I²C Functions:

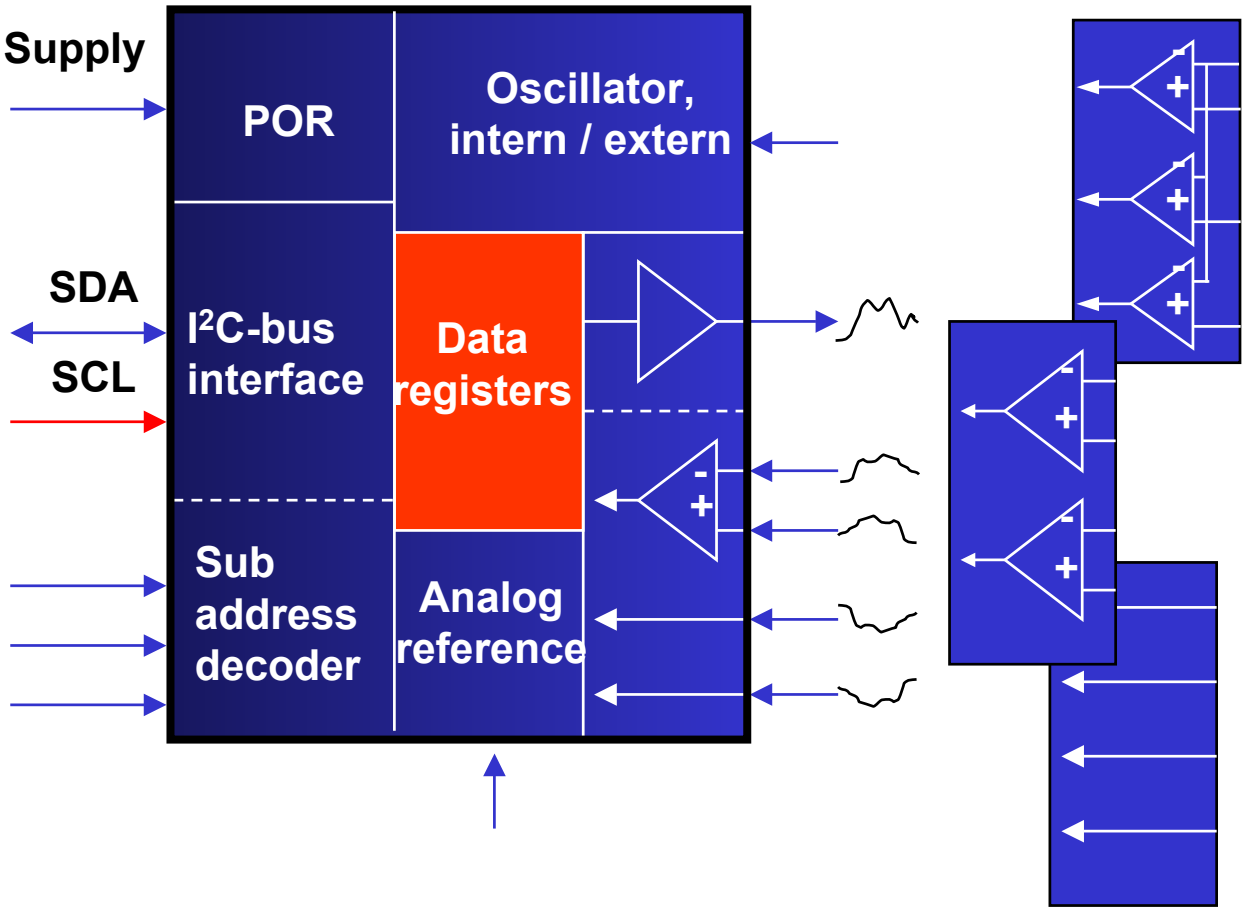
- A/D and D/A Conversion
- Bus Master – 8-bit parallel to I²C protocol
- Bus repeater, hubs and expansion
- Jumper and DIP switch replacement
- Power, dim and blink LEDs
- General Purpose I/O expansion
- Bus isolation and extension
- Bus multiplexing
- Non-volatile storage -Serial EEPROMs
- System monitoring (temperature/voltage)
- Voltage Level Translations

Family Features:

- 100 and 400 kHz I²C operation
- 2.3 to 5.5 V operating supply voltage
- -40 to +85 ºC operating temperature
- I²C and SMBus compatibility
- Hardware address pins (A_O, A_1, A_2)
- SO, SSOP, TSSOP and HVQFN packages
- High volume process manufacturing

www.philipslogic.com/i2c

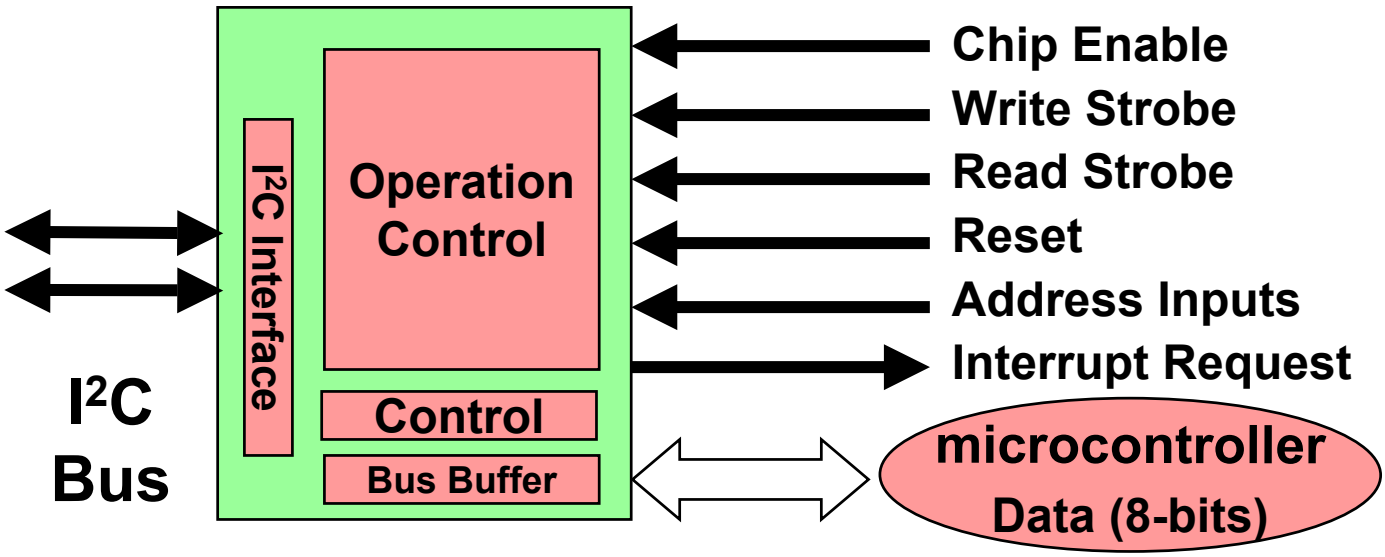
I²C Analog to Digital Converter



-Converts signals from digital to analog and analog to digital
-4 channel Analog to Digital converter
-1 channel Digital to Analog converter
-Internal oscillator
-Power On Reset (POR)

- **PCF8591, 100 kHz 4 channel 8 bit ADC**

Parallel Bus to I²C Bus Controller



–Serves as an interface between most standard 8-bit parallel-bus microcontrollers/microprocessors and the serial I²C bus.

–Allows the parallel bus system to communicate bi-directionally with the I²C bus.

–Provides both master and slave functions.

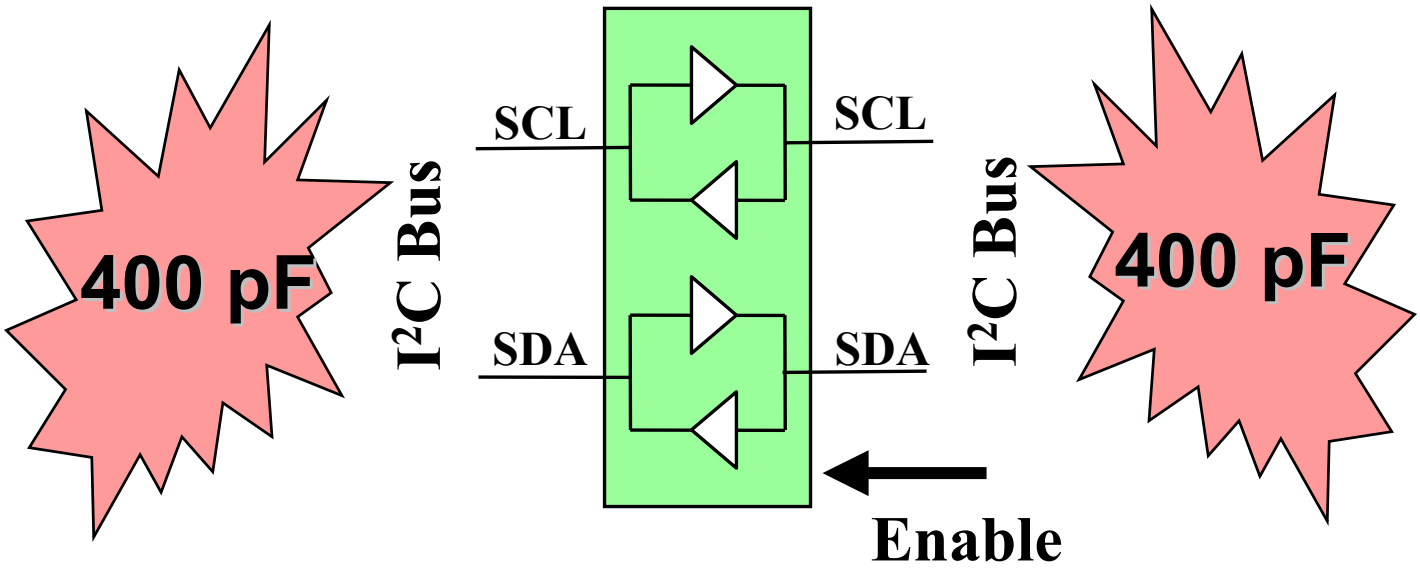
–Communication with the I²C bus is carried out on a byte-wise basis using interrupt or polled handshake.

–Controls all the I²C bus specific sequences, protocol, arbitration and timing.

–Internal oscillator

- **PCA9564, 400 kHz Parallel Bus to I²C Bus Controller**
- **PCF8584, 100 kHz Parallel Bus to I²C Bus Controller**

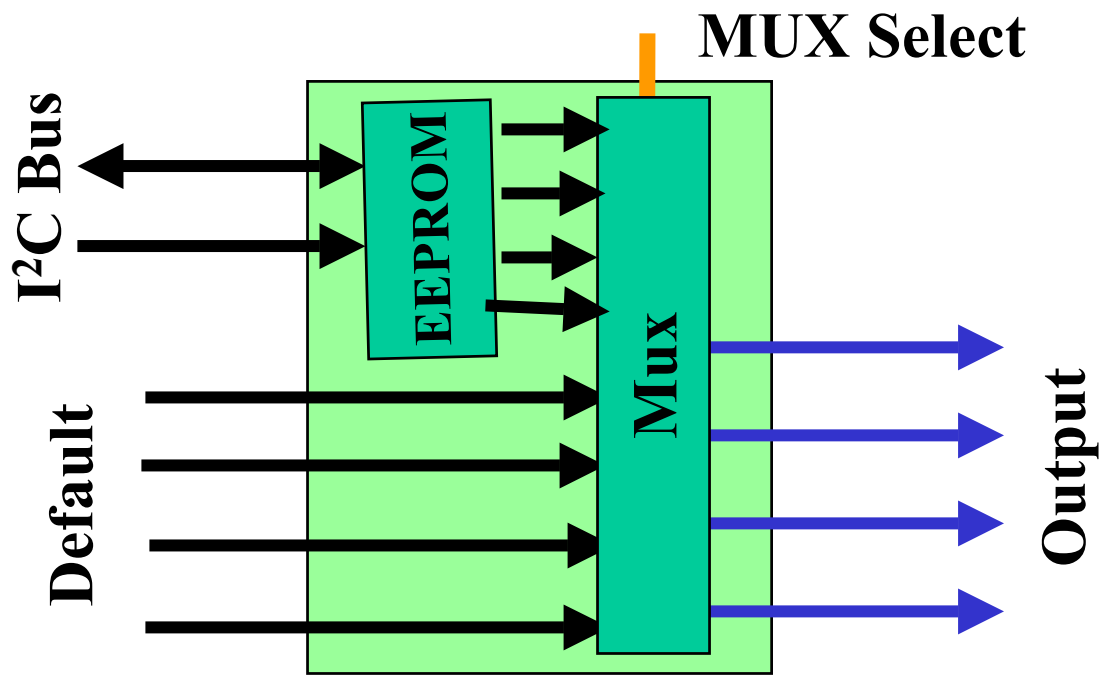
I²C Bus Repeater, Hub and Extenders



-Bi-directional I²C drivers, enable designers to isolate the I²C bus capacitance, accommodating more I²C devices or a longer bus length.
-400 pF load allowed on each segment.
-Different voltages (3.3 V or 5.0 V) allowed on each segment.
-Devices transparent to bus arbitration and contention protocols in a multi-master environment.
-P82B96 allows twisted pair, differential transmission or Opto-Electrical isolation of the I²C Bus

- **PCA9515, I²C Bus Repeater**
- **PCA9516, 5-Channel I²C Hub**
- **PCA9518, Expandable 5-Channel I²C Hub**
- **P82B96, Dual Bi-Directional Bus Buffer**
- **P82B715, I²C Bus Extender**

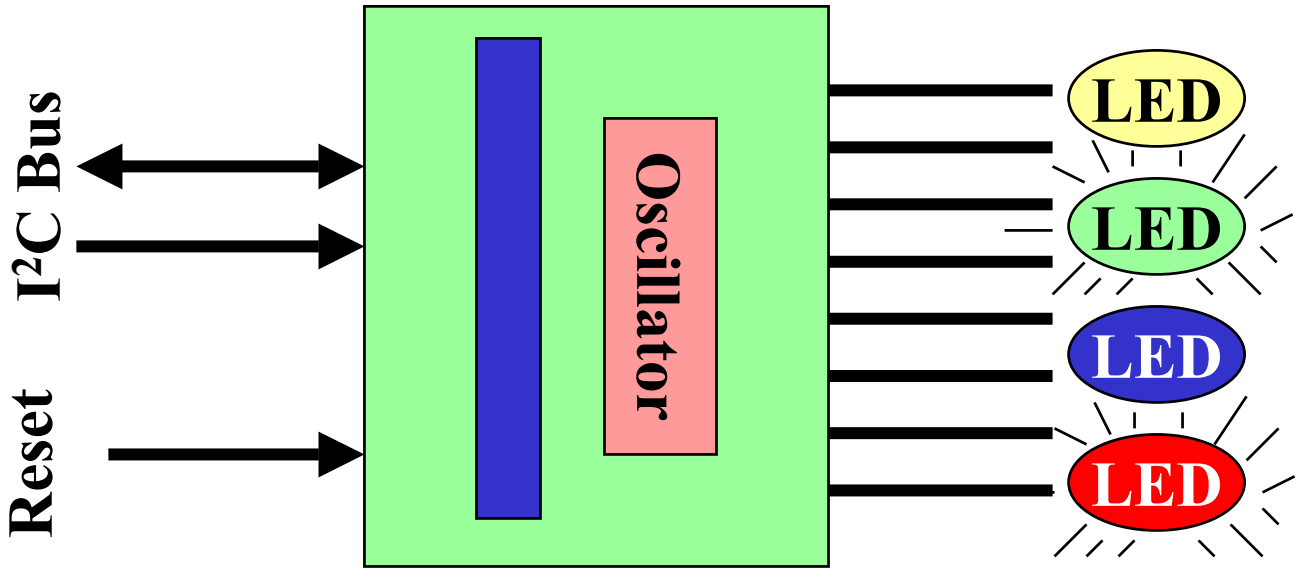
I²C EEPROM based DIP Switches



-Replacement for jumpers or dip switches.
-I²C controlled integrated EEPROM and Multiplexer eliminates need to open equipment to modify jumpers/dip switches by hand.
-Multiplex between the default values or the setting stored in the onboard I²C EEPROM register and programmed from the I²C bus.
-The non-volatile I²C EEPROM register values stay resident even when the device is powered down

- **PCA8550, 4-bit Multiplexed/1-bit Latched 5-bit EEPROM**
- **PCA9559, 5-bit Multiplexed/1-bit Latched 6-bit EEPROM**
- **PCA9560, Dual 5-bit Multiplexed/1-bit Latched EEPROM**
- **PCA9561, Quad 6-bit Multiplexed EEPROM**

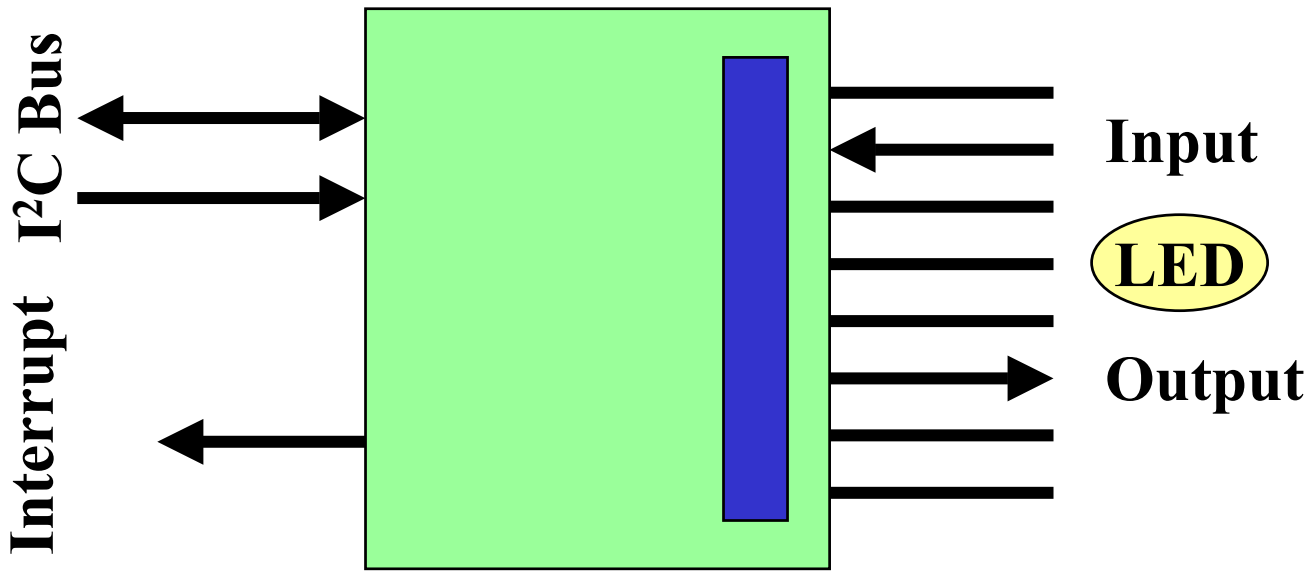
I²C LED Dimmers/Blinkers



-Used for general-purpose LED dimming and blinking.
-Internal oscillator, no external components required.
-Two user definable blink rates and duty cycles.
-Single transmission to turn LEDs on, off, or blink at one of the two programmable blink rates.
-I²C/SMBus not tied up by sending repeated transmissions to blink LEDs.
-Hardware Reset pin.
-High current open drain outputs.
-PCA953x Dimmers vary between 160 Hz and 1.6 seconds
-PCA955x Blinkers vary between 40 Hz and 6.4 seconds

- **PCA9530/50, 2-Bit I²C/SMBus LED Controller**
- **PCA9531/51, 8-bit I²C/SMBus LED Controller**
- **PCA9532/52, 16-bit I²C/SMBus LED Controller**
- **PCA9533/53, 4-bit I²C/SMBus LED Controller**

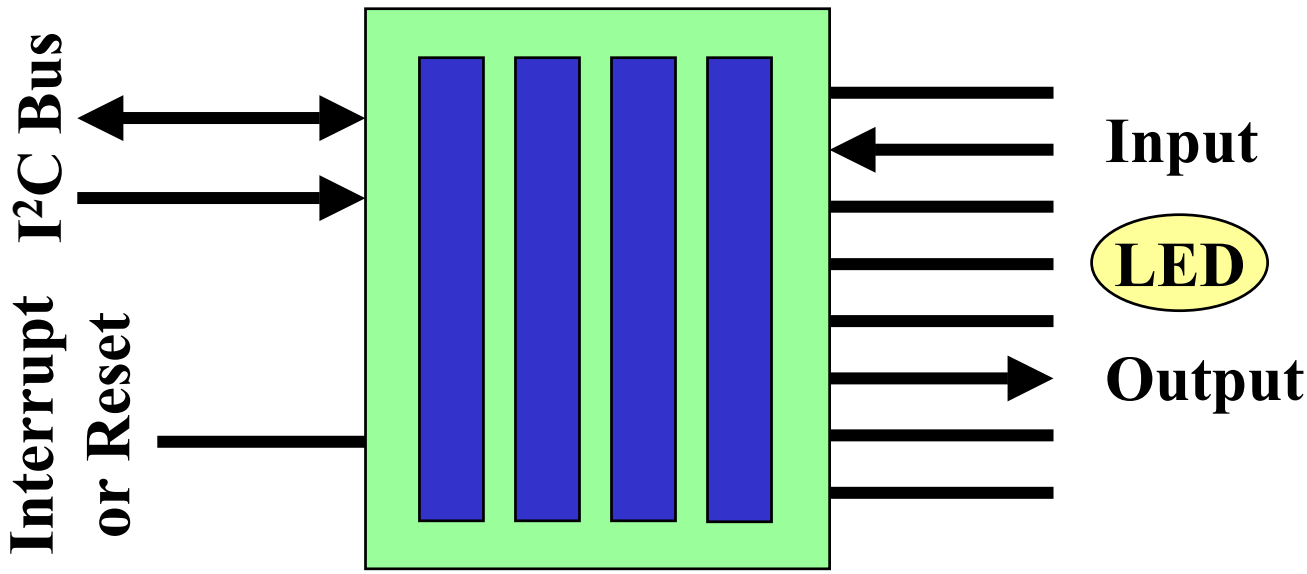
I²C General Purpose Quasi I/O



-Used for general-purpose input and output (GPIO) extension from the I²C/SMBus.
-Connect parallel I/O to the serial I²C/SMBus and provide expansion capability for the I/O controller Hub (South Bridge).
-One registers for Quasi Output (upper transistor on for one/half clock cycle):
    -Input or Output (value)
-Devices may have Interrupt output pin.

- **PCA9500, 8-bit GPIO w/ 2kbit EEPROM**
- **PCA9501, 8-bit GPIO w/2kbit EEPROM & Interrupt**
- **PCA9558, 8-bit GPIO w/5-bit DIP Switch & 2kbit EEPROM**
- **PCF8574/74A, 8-bit I²C I/O Port with Interrupt**
- **PCF8575/75C, 16-bit I²C I/O Port with Interrupt**

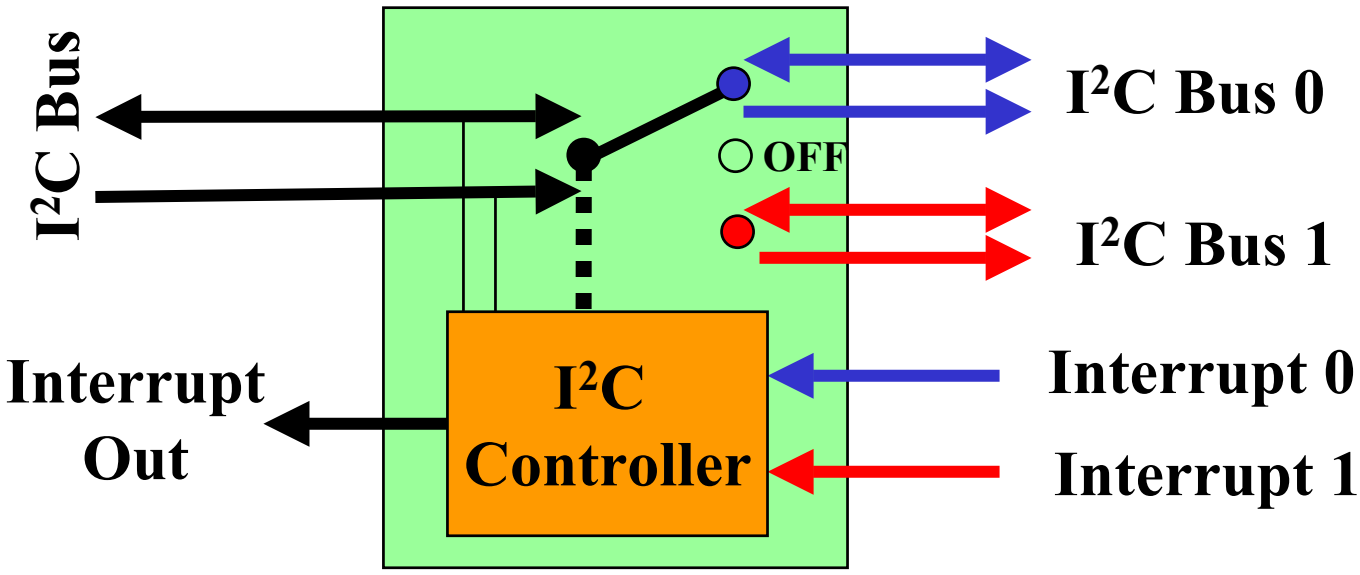
I²C General Purpose True I/O



-Used for general-purpose input and output (GPIO) extension from the I²C/SMBus.
-Connect parallel I/O to the serial I²C/SMBus and provide expansion capability for the I/O controller Hub (South Bridge).
-Four sets of registers for true outputs (push-pull outputs):
    -Configuration (Input or Output)
    -Input (Read I/O value)
    -Output (Force output value)
    -Polarity (active high or low)
-Devices have either Interrupt or Hardware Reset pin.

- **PCA9534/54/54A, 8-bit I²C and SMBus I/O Port with Int**
- **PCA9535/55, 16-bit I²C and SMBus I/O Port with Interrupt**
- **PCA9556, Octal SMBus and I²C Registered Interface**
- **PCA9557, Octal SMBus and I²C Registered Interface**

I²C Multiplexers



-The SCL/SDA upstream channel fans out to multiple SCx/SDx channels that are selected by the programmable control register.

-The Multiplexers can select none or only one SCx/SDx channels at a time.

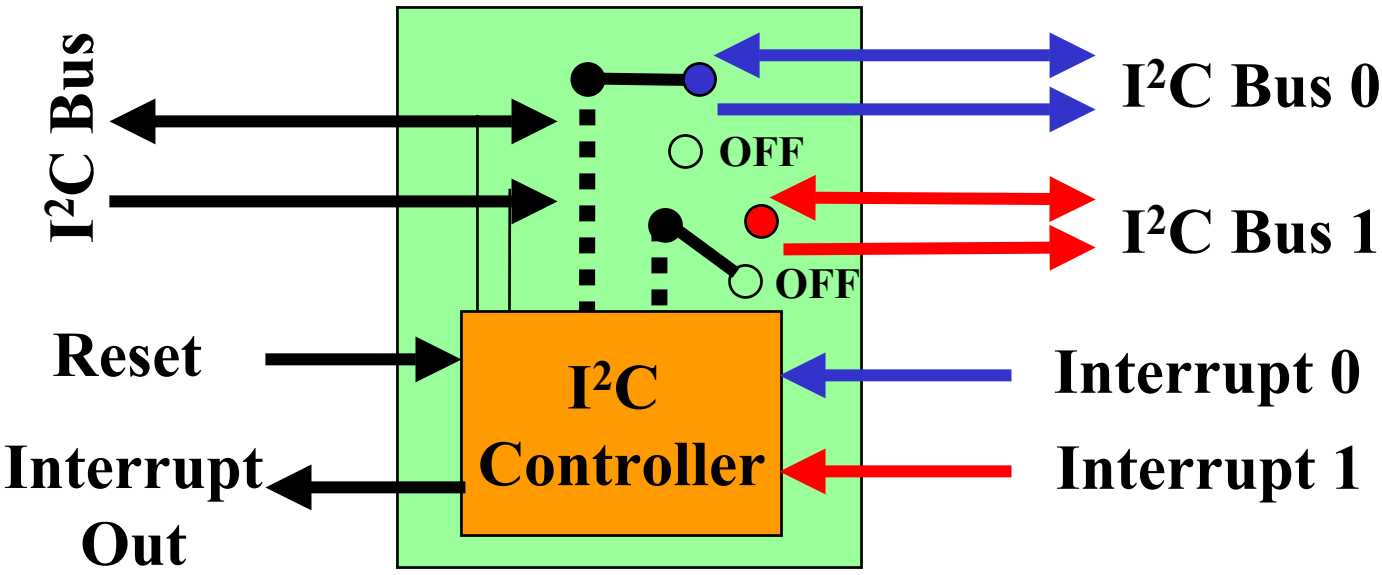
-Applications include:

- -Address conflict resolution (e.g., SPD EEPROMs on DIMMs).
- -I²C sub-branch isolation
- -I²C bus level shifting (e.g., individual SCx/SDx channels at 5.0 V, 3.3 V, 2.5 V or 1.8 V).

-Interrupt logic provides flag to master for system monitoring.

- **PCA9540, 1 to 2-Channel Multiplexer**
- **PCA9541, 2 to 1 Master Selector with Int Logic & Reset**
- **PCA9542, 1 to 2-Channel Multiplexer with Interrupt Logic**
- **PCA9544, 1 to 4-Channel Multiplexer with Interrupt Logic**

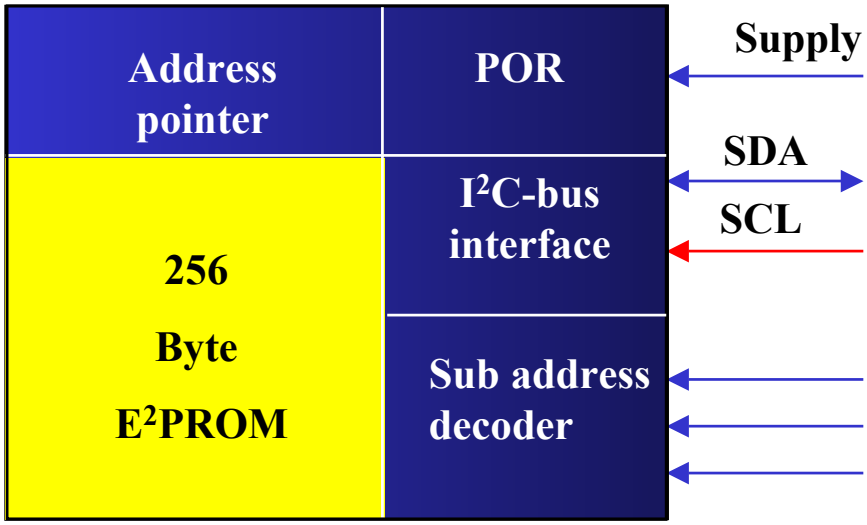
I²C Switches



-The SCL/SDA upstream channel fans out to multiple SCx/SDx channels that are selected by the programmable control register.
-The Switches can select individual SCx/SDx channels one at a time, all at once or in any combination through I²C commands.
-Applications include:
    -Address conflict resolution (e.g., SPD EEPROMs on DIMMs).
    -I²C sub-branch isolation
    -I²C bus level shifting (e.g., individual SCx/SDx channels at 5.0 V, 3.3 V, 2.5 V or 1.8 V).
-Hardware Reset provides a means of resetting the bus should it hang up, without rebooting the entire system.

- **PCA9543, 1 to 2-Channel Switch with Int Logic and Reset**
- **PCA9545, 1 to 4-Channel Switch with Int Logic and Reset**
- **PCA9546, 1 to 4-Channel Switch with Reset**
- **PCA9548, 1 to 8-Channel Switch with Reset**

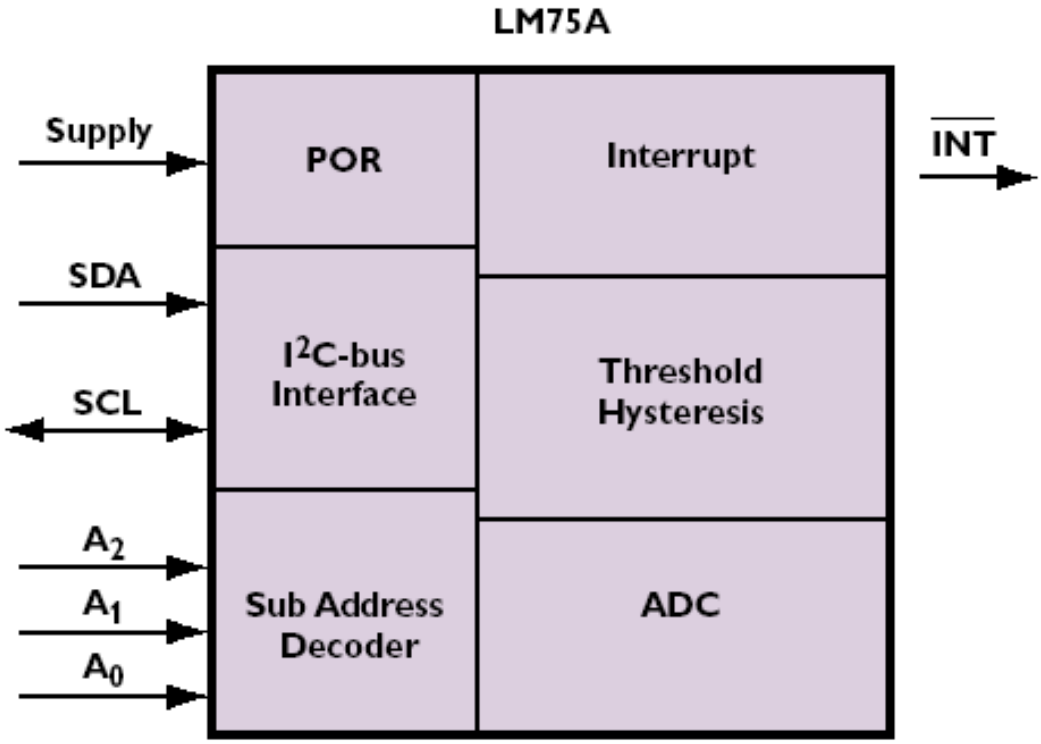
I²C Serial EEPROM



-Wide voltage range minimizes the number of EEPROMs that need to be in inventory
-I²C bus is used to read and write information to and from the memory
-Wide voltage range of 2.5 to 5.5V
-1,000,000 read and write cycles
-10 year data retention

- **PCF8581(C), 128 x 8-bit CMOS EEPROM**
- **PCF8582C-2, 256 x 8-bit CMOS EEPROM**
- **PCF8594C-2, 512 x 8-bit CMOS EEPROM**
- **PCF8598C-2, 1024 x 8-bit CMOS EEPROM**
- **PCF85102C-2, 256 x 8-bit CMOS EEPROM**
- **PCF85103C-2, 256 x 8-bit CMOS EEPROM**
- **PCF85116-3, 2048 x 8-bit CMOS EEPROM**

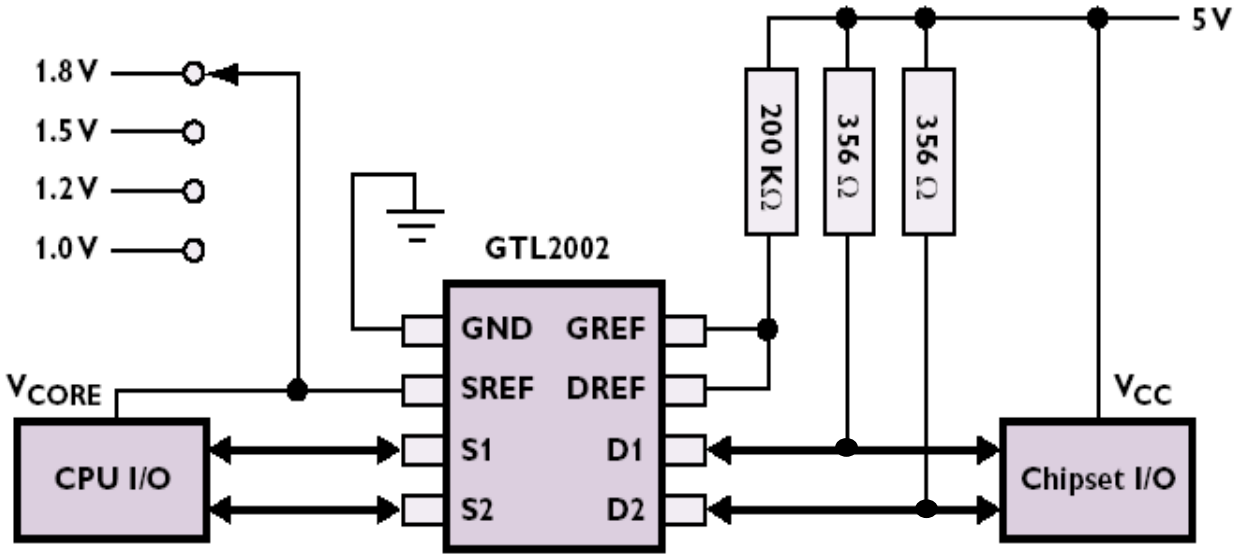
I²C Temperature and Voltage Sensors



-Hardware monitors use the I²C bus to report temperature and/or voltage.
-Some of the temperature monitors include analog input or external sense pins that allow external transistors/diodes to be located in external components (e.g., processors) that sense the temperature much more accurately than if the sensor was mounted externally on the package.

- **NE1617A, I²C Temperature Monitor**
- **NE1618, High Accuracy I²C Temperature Monitor**
- **NE1619, I²C Temperature and Voltage Monitor**
- **LM75A, Digital Temp Sensor and Thermal Watchdog™**

I²C Voltage Level Translator



–Voltage translation between any voltage from 1.0 V to 5.0 V

–Open drain on both sides with no drive

–Bi-directional with no direction pin

–Reference voltage clamps the input voltage with low propagation delay

–Used for bi-directional translation of I²C buses at 3.3V and/or 5 V to the processor I²C port at 1.2 V or 1.5 V or any voltage in-between

–BiCMOS process provides excellent ESD performance

- **GTL2000, 20-bit Bi-directional Lev el Translator**
- **GTL2002, 20-bit Bi-directional Lev el Translator**
- **GTL2010, 20-bit Bi-directional Lev el Translator**