


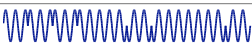


Communication methods

- Communication methods
 - Media and signaling conventions used to transmit data between digital devices
 - Different physical layers methods including:
 - wires, radio frequency (RF), optical (IR, fiber)
 - Different encoding schemes including:
 - amplitude, frequency, and pulse-width modulation

| Modulation Technique | Waveform |
|----------------------------------|--|
| No encoding (Baseband) |  |
| On-Off Keying (OOK) |  |
| Frequency Shift Keying (FSK) |  |
| Binary Phase Shift Keying (BPSK) |  |

Communication methods

- Dimensions to consider
 - bandwidth – number of wires – serial/parallel
 - speed – bits/bytes/words per second
 - timing methodology – synchronous or asynchronous
 - number of destinations/sources
 - arbitration scheme – daisy-chain, centralized, distributed
 - protocols – provide some guarantees as to correct communication

Bandwidth

- **Serial**
 - Single wire or channel to transmit information one bit at a time
 - Requires synchronization between sender and receiver
 - Sometimes includes extra wires for clock and/or handshaking
 - Good for inexpensive connections (e.g., terminals)
 - Good for long-distance connections (e.g., LANs)
 - Examples: RS-232, Ethernet, I2C, IrDA, USB, Firewire, Bluetooth
- **Parallel**
 - Multiple wires to transmit information one byte or word at a time
 - Good for high-bandwidth requirements (CPU to disk)
 - More expensive wiring/connectors/current requirements
 - Examples: SCSI-2, PCI bus (PC), PCMCIA (Compact Flash)
- **Issues**
 - Encoding, data transfer rates, cost of connectors and wires, modularity, error detection and/or correction

Speed

- **Serial**
 - low-speed, cheap connections
 - RS-232 1K–20K bits/sec, copper wire
 - medium-speed efficient connections
 - I2C 10K–400K bits/sec, board traces
 - IrDA 9.6K–4M bits/sec, line-of-sight, 0.5–6.0m
 - high-speed, expensive connections
 - USB 1.5M bytes/sec, USB2 60M bytes/sec, USB3 625M bytes/s
 - Ethernet 1.5M–1G bits/sec, twisted-pair or co-axial
 - Firewire 12.5–50M bytes/sec
- **Parallel**
 - low-speed, not too wide
 - SCSI-2 10M bytes/sec, 8 bits wide
 - PCI bus, 250M bytes/sec, 32 bits wide
 - PCMCIA (CF+), 9–10M bytes/sec, 16 bits wide
 - high-speed, very wide – memory systems in large multi-processors
 - 200M–2G bytes/sec, 128–256 bits wide

Speed

- Issues
 - length of the wires (attenuation, noise, capacitance)
 - connectors (conductors and/or transducers)
 - environment (RF/IR interference, noise)
 - current switching (spikes on supply voltages)
 - number and types of wires (cost of connectors, cross-talk)
 - flow-control (if communicating device can't keep up)

Timing methodology

- Asynchronous
 - Fewer wires (no clock)
 - no skew concerns
 - synchronization overhead
 - appropriate for loosely-coupled systems (CPU and peripherals)
 - common in serial schemes
- Synchronous
 - clock wires and skew concerns
 - no synchronization overhead
 - can be high-speed if delays are small and can be controlled
 - appropriate for tightly-coupled systems (CPU and memory/disk)
 - common in parallel schemes

Timing methodology

- **Issues**
 - clock period and wire delay
 - synchronization and skew
 - encoding of timing and data information
 - handshaking
 - flow-control
 - power consumption

Number of devices communicating

- **Single source – single destination**
 - point-to-point
 - cheap connections, no tri-stating necessary
- **Single source – multiple destination**
 - fanout limitations
 - addressing scheme to direct data to one destination
- **Multiple source – multiple destination**
 - arbitration between senders
 - tri-stating capability is necessary
 - collision detection
 - addressing scheme
 - priority scheme
 - fairness considerations

Arbitration schemes

- Daisy-chain or token passing
 - devices either act or pass to next
 - fixed priority order
 - as many wires as devices
 - fairness issues
- Centralized
 - request to central arbiter
 - central arbiter implements priority scheme
 - wires from/to each device can be costly
 - can be dynamically changing priority/fairness
- Distributed
 - no central arbiter
 - common set of wires (or ether) observed by all devices
 - fixed priority/fairness scheme

Serial case studies

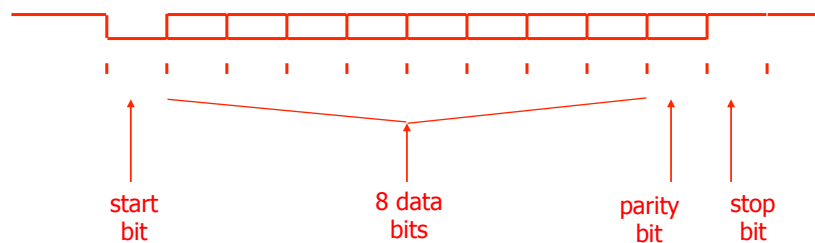
- RS-232 (IEEE standard)
 - serial protocol for point-to-point, low-cost, low-speed applications for PCs
- SPI (Motorola)
 - 10Mbits/sec, commonly used for microcontroller to peripheral connections
- I2C (Philips) TWI (Atmel)
 - up to 400Kbits/sec, serial bus for connecting multiple components
- USB (Intel – followed by USB-2, and now USB-3)
 - 12-480Mbits/sec, isochronous transfer, desktop devices
- Bluetooth (Ericsson – cable replacement)
 - 700Kbits/sec, multiple portable devices, special support for audio
- Ethernet (popularized by Xerox)
 - most popular local area network protocol with distributed arbitration
- IrDA (Infrared Data Association)
 - up to 115kbps wireless serial (Fast IrDA up to 4Mbps)
- IEEE1394 (Apple's Firewire)
 - 12.5-50Mbytes/sec, consumer electronics (video cameras, TVs, audio, etc.)

RS-232

standard serial line

RS-232 (standard serial line)

- Point-to-point, full-duplex
- Synchronous or asynchronous
- Flow control
- Variable baud (bit) rates
- Cheap connections (low-quality and few wires)
- Variations: parity bit; 1, 1.5, or 2 stop bits



RS-232 wires

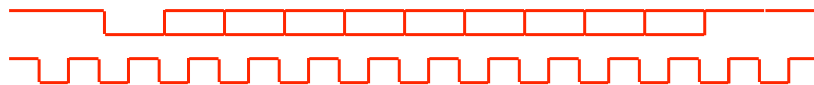
- TxD – transmit data
 - TxC – transmit clock
 - RTS – request to send
 - CTS – clear to send

 - RxD – receive data
 - RxC – receive clock
 - DSR – data set ready
 - DTR – data terminal ready

 - Ground
- all wires active low
- "0" = -12v, "1" = 12v
- special driver chips that generate $\pm 12v$ from 5v

Transfer modes

- Synchronous
 - clock signal wire is used by both receiver and sender to sample data
- Asynchronous
 - no clock signal in common
 - data must be oversampled (16x is typical) to find bit boundaries
- Flow control
 - handshaking signals to control rate of transfer



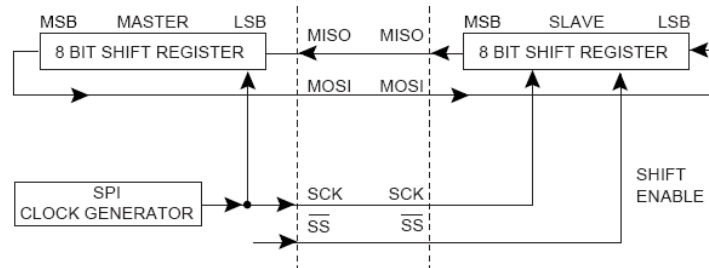
Serial Peripheral Interface (SPI)

Serial Peripheral Interface

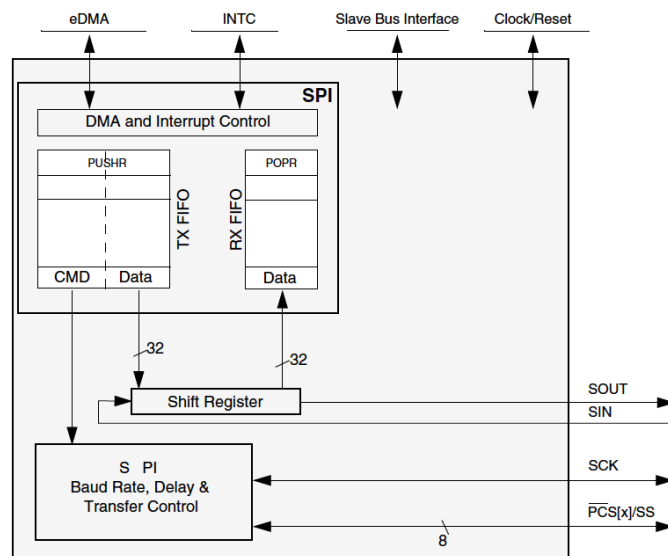
- Common serial interface on many microcontrollers
- Simple 8-bit exchange between two devices
 - Master initiates transfer and generates clock signal
 - Slave device selected by master
- One-byte at a time transfer
 - Data protocols are defined by application
 - Must be in agreement across devices

SPI Block Diagram

- 8-bits transferred in each direction every time
- Master generates clock
- Shift enable used to select one of many slaves



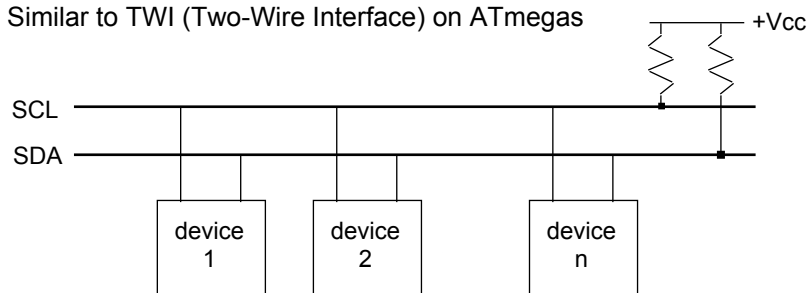
Freescale SPI module



Inter-Integrated Circuit Bus (I2C)

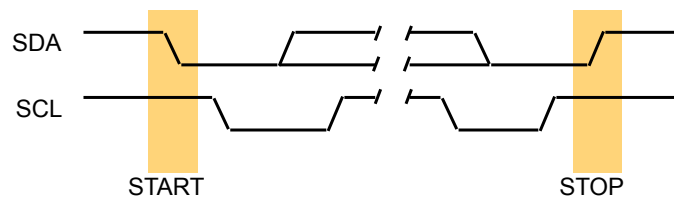
Inter-Integrated Circuit Bus (I2C)

- Modular connections on a printed circuit board
- Multi-point connections (needs addressing)
- Synchronous transfer (but adapts to slowest device)
- Similar to Controller Area Network (CAN) protocol used in automotive applications
- Similar to TWI (Two-Wire Interface) on ATmegs



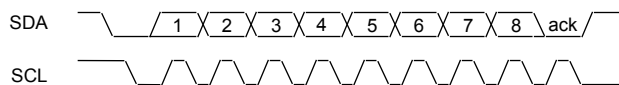
Serial data format

- SDA going low while SCL high signals start of data
- SDA going high while SCL high signals end of data
- SDA can change when SCL low
- SCL high (after start and before end) signals that a data bit can be read



Byte transfer

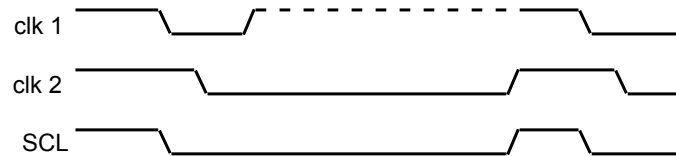
- Byte followed by a 1 bit acknowledge from receiver
- Open-collector wires
 - sender allows SDA to rise
 - receiver pulls low to acknowledge after 8 bits



- Multi-byte transfers
 - first byte contains address of receiver
 - all devices check address to determine if following data is for them
 - second byte usually contains address of sender

Clock synchronization

- Synchronous data transfer with variable speed devices
 - go as fast as the slowest device involved in transfer
- Each device looks at the SCL line as an input as well as driving it
 - if clock stays low even when being driven high then another device needs more time, so wait for it to finish before continuing
 - rising clock edges are synchronized



Arbitration

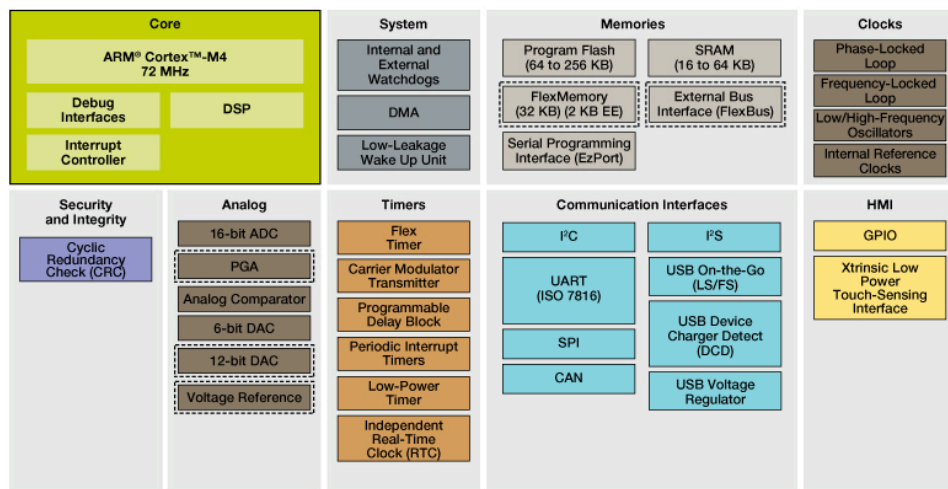
- Devices can start transmitting at any time
 - wait until lines are both high for some minimum time
 - multiple devices may start together - clocks will be synchronized
- All senders will think they are sending data
 - possibly slowed down by receiver (or another sender)
 - each sender keeps watching SDA - if ever different (driving high, but its really low) then there is another driver
 - sender that detects difference gets off the bus and aborts message
- Device priority given to devices with early 0s in their address
 - 00...111 has higher priority than 01...111

Inter-Integrated Circuit Bus (I2C)

- Supports data transfers from 0 to 400KHz
- Philips (and others) provide many devices
 - microcontrollers with built-in interface
 - A/D and D/A converters
 - parallel I/O ports
 - memory modules
 - LCD drivers
 - real-time clock/calendars
 - DTMF decoders
 - frequency synthesizers
 - video/audio processors

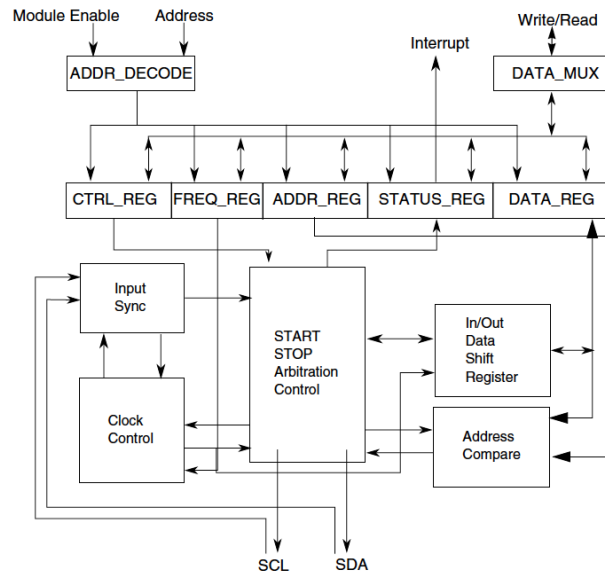
Freescal Block Diagram

Kinetis K20 Mid-Performance USB MCUs



Optional

Freescale I2C module



USB

Universal Serial Bus

- Connecting peripherals to PCs
 - Ease-of-use
 - Low-cost
 - Up to 127 devices (optionally powered through bus)
 - Transfer rates up to 480 Mb/s (USB 2.0)
 - Variable speeds and packet sizes
 - Full support for real-time data for voice, audio, and video
 - Protocol flexibility for mixed-mode isochronous data transfers and asynchronous messaging
 - PC manages bus and allocates slots (host controller)
 - Can have multiple host controllers on one PC
 - Support more devices than 127

USB versions

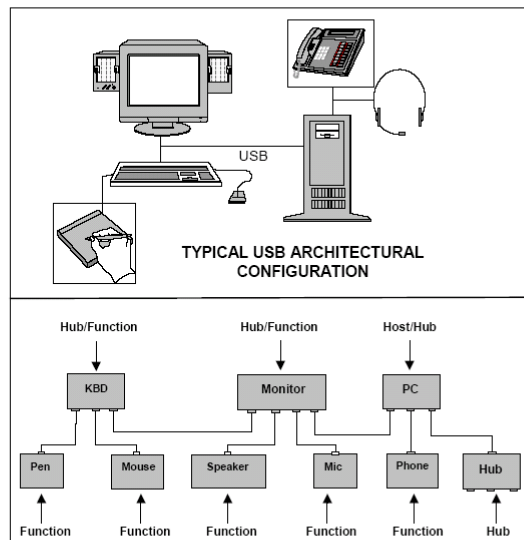
| Feature | USB 2.0 | USB 3.0 |
|-----------------------------|---------------------------------|---|
| Transmit/Receive | Half-Duplex | Duplex |
| Data Rate | 1.5Mbs/12Mbs/480Mbs | USB 2.0 Data Rates + 5Gbs |
| Useful Data Rate | 480Mbs (Including NRZ overhead) | 4Gbs * 2 (Excluding 8b10b overhead) |
| Burst | No | 16 bursts |
| Stream | No | Each Bulk endpoint can have 64K Stream (more pipes) |
| Host – Device Communication | Broadcast | Unicast (Addressed) |
| Device Data Throttle | ACK, NYET/PING | ACK, NRDY/ERDY |
| Device Initiated Packets | No | ERDY, LMPs and Device notifications |
| Power States | Low Power Mode (LPM), Suspend | U1, U2, and U3 Power States |
| SOF/ITP | Fixed at 125µs | ITP-variable |
| Packet Size (bytes) | 64 Control, 512 Bulk, 1024 ISOC | 512 Control, 1024 Bulk & ISOC |
| Error Rate | 10** ⁻⁹ | 10** ⁻¹² |

USB Peripherals

| PERFORMANCE | APPLICATIONS | ATTRIBUTES |
|--|--|--|
| LOW-SPEED <ul style="list-style-type: none"> Interactive Devices 10 – 100 kb/s | Keyboard, Mouse Stylus Game Peripherals Virtual Reality Peripherals | Lowest Cost Ease-of-Use Dynamic Attach-Detach Multiple Peripherals |
| FULL-SPEED <ul style="list-style-type: none"> Phone, Audio, Compressed Video 500 kb/s – 10 Mb/s | POTS Broadband Audio Microphone | Lower Cost Ease-of-Use Dynamic Attach-Detach Multiple Peripherals Guaranteed Bandwidth Guaranteed Latency |
| HIGH-SPEED <ul style="list-style-type: none"> Video, Storage 25 – 400 Mb/s | Video Storage Imaging Broadband | Low Cost Ease-of-Use Dynamic Attach-Detach Multiple Peripherals Guaranteed Bandwidth Guaranteed Latency High Bandwidth |

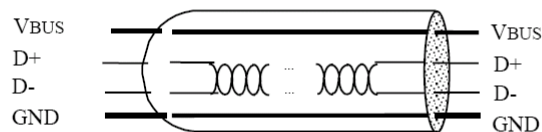
USB

- Tree of devices – one root controller



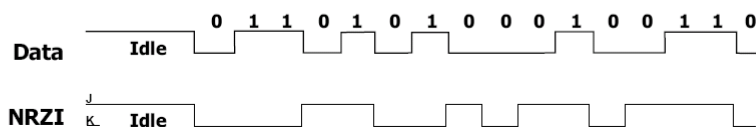
USB Data Transfer

- Data transfer speeds
 - Low is <0.8v, high is >2.0v differential
 - 480Mb/sec, 12Mb/sec, 1.5Mb/sec
 - Data is NRZI encoded (data and clock on one wire)
 - SYNC at beginning of every packet

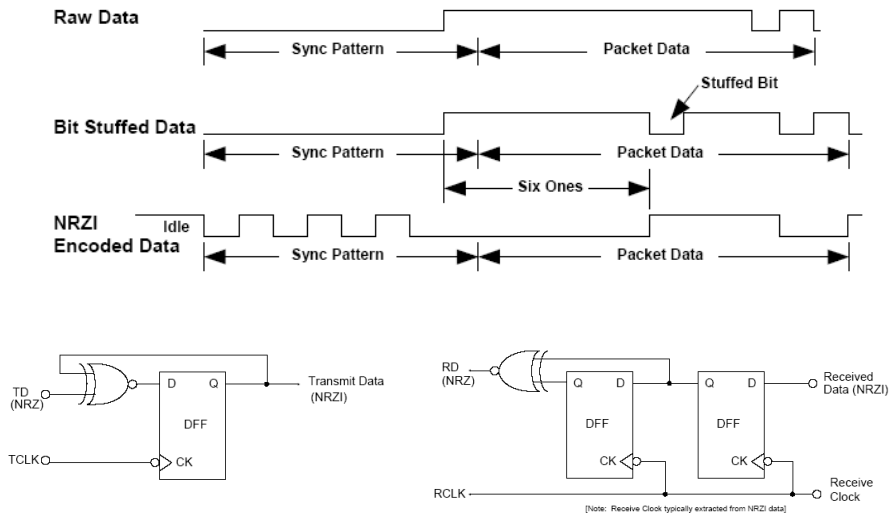


NRZI Encoding

- NRZI – Non-return to zero inverted
 - Toggles a signal to transmit a “0” and leaves the signal unchanged for a “1”
 - Also called transition encoding
 - Long string of 0s generates a regular waveform with a frequency half the bit rate
 - Long string of 1s generates a flat waveform – bit stuff a 0 every 6 consecutive 1s to guarantee activity on waveform



NRZI Encoding (cont' d)



USB Data Transfer Types

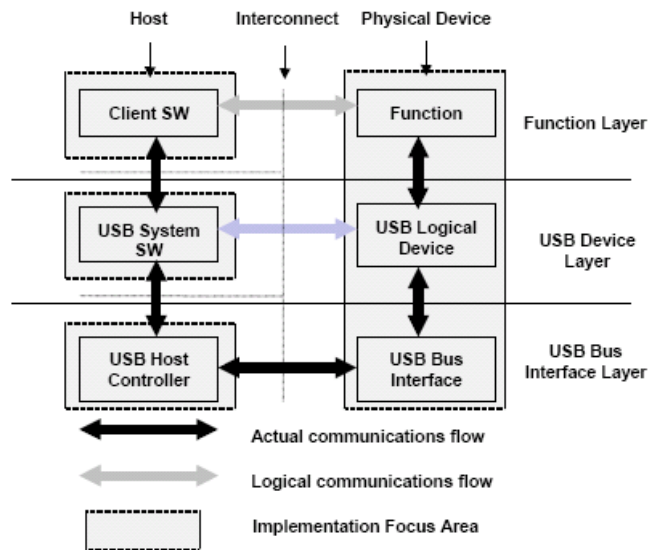
- Control Transfers:
 - Used to configure a device at attach time and can be used for other device-specific purposes, including control of other pipes on the device.
- Bulk Data Transfers:
 - Generated or consumed in relatively large and bursty quantities and have wide dynamic latitude in transmission constraints.
- Interrupt Data Transfers:
 - Used for timely but reliable delivery of data, for example, characters or coordinates with human-perceptible echo or feedback response characteristics.
- Isochronous Data Transfers:
 - Occupy a prenegotiated amount of USB bandwidth with a prenegotiated delivery latency. (Also called streaming real time transfers)

USB Packet Format

- Sync + PID + data + CRC
- Basic data packet
 - Sync: 8 bits (00000001)
 - PID: 8 bits (packet id – type)
 - Data: 8-8192 bits (1K bytes)
 - CRC: 16 bits (cyclic redundancy check sum)
- Other data packets vary in size
 - May be as short as only 8 bits of PID

USB Protocol Stack

- FTDI USB chip implements right side
- Communicates to physical device through SPI



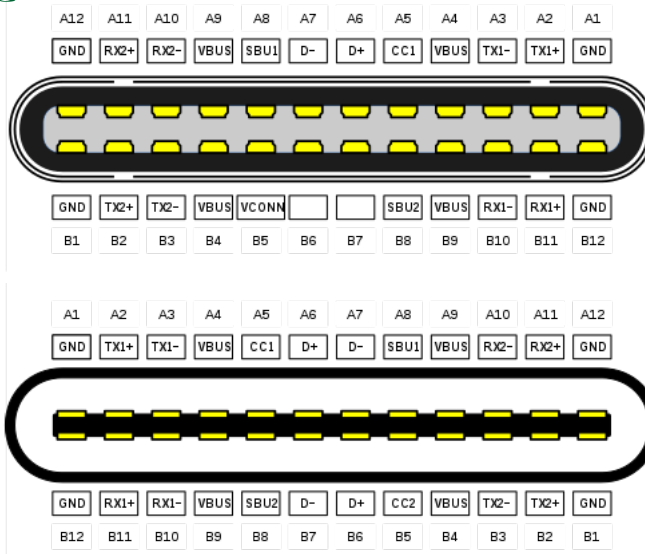
USB-C

- technically known as **USB Type-C**
- a 24-pin **USB** connector system, which is distinguished by its rotationally-symmetrical connector.
- The USB Type-C Specification 1.0 was published by the **USB Implementers Forum** (USB-IF) and was finalized in August 2014. It was developed at roughly the same time as the **USB 3.1** specification.
- A device that implements USB-C does not necessarily implement **USB 3.1**, **USB Power Delivery**, or Alternate Mode.

USB-C



USB-C



Full-featured USB 3.1 and 2.0 Type-C cable wiring

| Plug 1, USB Type-C | | USB Type-C cable | | | | | | Plug 2, USB Type-C | |
|--------------------|-------------------|-----------------------|-------|------------------------|---|--------------------|------------------|--------------------|--|
| Pin | Name | Wire colour | No | Name | Description | 2.0 ^[a] | Pin | Name | |
| Shell | Shield | Braid | Braid | Shield | Cable external braid | ✓ | Shell | Shield | |
| A1, B12, B1, A12 | GND | Tin-plated | 16 | GND_PWRr1 | Ground for power return | ✓ | A1, B12, B1, A12 | GND | |
| | | | 1 | GND_PWRr2 | | x | | | |
| A4, B9, B4, A9 | V _{BUS} | Red | 2 | PWR_V _{BUS} 1 | V _{BUS} power | ✓ | A4, B9, B4, A9 | V _{BUS} | |
| | | | 17 | PWR_V _{BUS} 2 | | x | | | |
| B5 | V _{CONN} | Yellow | 18 | PWR_V _{CONN} | V _{CONN} power, for active cables ^[b] | ✓ | B5 | V _{CONN} | |
| A5 | CC | Blue | 3 | CC | Configuration channel | ✓ | A5 | CC | |
| A6 | Dp1 | White | 4 | UTP_Dp ^[c] | Unshielded twisted pair, positive | ✓ | A6 | Dp1 | |
| A7 | Dn1 | Green | 5 | UTP_Dn ^[c] | Unshielded twisted pair, negative | ✓ | A7 | Dn1 | |
| A8 | SBU1 | Red | 14 | SBU_A | Sideband use A | x | B8 | SBU2 | |
| B8 | SBU2 | Black | 15 | SBU_B | Sideband use B | x | A8 | SBU1 | |
| A2 | SSTXp1 | Yellow ^[d] | 6 | SDPp1 | Shielded differential pair #1, positive | x | B11 | SSRXp1 | |
| A3 | SSTXn1 | Brown ^[d] | 7 | SDPn1 | Shielded differential pair #1, negative | x | B10 | SSRXn1 | |
| B11 | SSRXp1 | Green ^[d] | 8 | SDPp2 | Shielded differential pair #2, positive | x | A2 | SSTXp1 | |
| B10 | SSRXn1 | Orange ^[d] | 9 | SDPn2 | Shielded differential pair #2, negative | x | A3 | SSTXn1 | |
| B2 | SSTXp2 | White ^[d] | 10 | SDPp3 | Shielded differential pair #3, positive | x | A11 | SSRXp2 | |
| B3 | SSTXn2 | Black ^[d] | 11 | SDPn3 | Shielded differential pair #3, negative | x | A10 | SSRXn2 | |
| A11 | SSRXp2 | Red ^[d] | 12 | SDPp4 | Shielded differential pair #4, positive | x | B2 | SSTXp2 | |
| A10 | SSRXn2 | Blue ^[d] | 13 | SDPn4 | Shielded differential pair #4, negative | x | B3 | SSTXn2 | |

a. ^ USB 2.0 Type-C cables do not include wires for SuperSpeed or sideband use.
b. ^ V_{CONN} must not traverse end-to-end through the cable. Some isolation method must be used.
c. ^ ^ b There is only a single differential pair for non-SuperSpeed data in the cable, which is connected to A6 and A7. Contacts B6 and B7 should not be present in the plug.
d. ^ a b c d e f g h Wire colours for differential pairs are not mandated.

USB-C

Type-C plug and receptacle pinouts

| Pin | Name | Description | Pin | Name | Description |
|-----|------------------|--|-----|------------------|---|
| A1 | GND | Ground return | B12 | GND | Ground return |
| A2 | SSTXp1 | SuperSpeed differential pair #1, TX, positive | B11 | SSRXp1 | SuperSpeed differential pair #2, RX, positive |
| A3 | SSTXn1 | SuperSpeed differential pair #1, TX, negative | B10 | SSRXn1 | SuperSpeed differential pair #2, RX, negative |
| A4 | V _{BUS} | Bus power | B9 | V _{BUS} | Bus power |
| A5 | CC1 | Configuration channel | B8 | SBU2 | Sideband use (SBU) |
| A6 | Dp1 | Non-SuperSpeed differential pair, position 1, positive | B7 | Dn2 | Non-SuperSpeed differential pair, position 2, negative ^[a] |
| A7 | Dn1 | Non-SuperSpeed differential pair, position 1, negative | B6 | Dp2 | Non-SuperSpeed differential pair, position 2, positive ^[a] |
| A8 | SBU1 | Sideband use (SBU) | B5 | CC2 | Configuration channel |
| A9 | V _{BUS} | Bus power | B4 | V _{BUS} | Bus power |
| A10 | SSRXn2 | SuperSpeed differential pair #4, RX, negative | B3 | SSTXn2 | SuperSpeed differential pair #3, TX, negative |
| A11 | SSRXp2 | SuperSpeed differential pair #4, RX, positive | B2 | SSTXp2 | SuperSpeed differential pair #3, TX, positive |
| A12 | GND | Ground return | B1 | GND | Ground return |

a. ^a ^b There is only a single non-SuperSpeed differential pair in the cable. This pin is not connected in the plug/cable.

USB-C Alternate Modes

List of Alternate Mode partner specifications

| Name | Date | Protocol |
|--|---|---|
| DisplayPort Alternate Mode | Published in September 2014 | DisplayPort 1.4 ^{[20][21]} |
| Mobile High-Definition Link (MHL) Alternate Mode | Announced in November 2014 ^[22] | MHL 1.0, 2.0, 3.0 and superMHL 1.0 ^{[9][23][24][25]} |
| Thunderbolt Alternate Mode | Announced in June 2015 ^[26] | Thunderbolt 3 (includes DisplayPort 1.2 Alternate Mode) ^{[26][27][28][29]} |
| HDMI Alternate Mode | Announced in September 2016 ^[30] | HDMI 1.4b ^{[31][32][33][34]} |

Alternate Mode protocol support matrix for USB-C cables and adapters

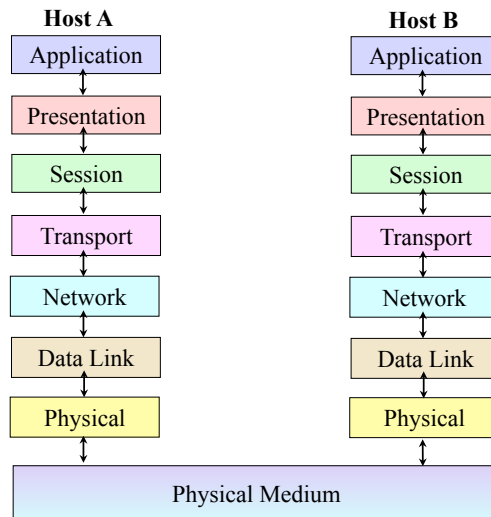
| Mode | USB 3.1 Type-C cable ^[nb 1] | | | | | | Adapter cable or adapter | | | | | Construction | |
|-------------|--|---------------------|-----|-------------|-----------------------|----------|--------------------------|------|------|-------------|-----------|--------------------|-----------------|
| | USB ^[nb 2] | DisplayPort | | Thunderbolt | | superMHL | HDMI | HDMI | | DVI-D | | | Component video |
| | 3.1 | 1.2 | 1.4 | 20 Gbit/s | 40 Gbit/s | | 1.4b | 1.4b | 2.0b | single-link | dual-link | (YPbPr, VGA/DVI-A) | |
| DisplayPort | Yes | Yes | | | | | | | | No | | | Passive |
| | | Optional | | | | | | Yes | Yes | Yes | Yes | | Active |
| Thunderbolt | Yes ^[39] | Yes ^[39] | | Yes | Yes ^[nb 3] | | | | | No | | | Passive |
| | | Optional | | Optional | Yes | | | Yes | Yes | Yes | Yes | | Active |
| MHL | Yes | | | | | Yes | | Yes | No | Yes | No | No | Passive |
| | | | | | | Optional | | | Yes | | | Yes | Active |
| HDMI | | | | | | | Yes | Yes | No | Yes | No | No | Passive |
| | | | | | | | Optional | | | | | Yes | Active |

Ethernet

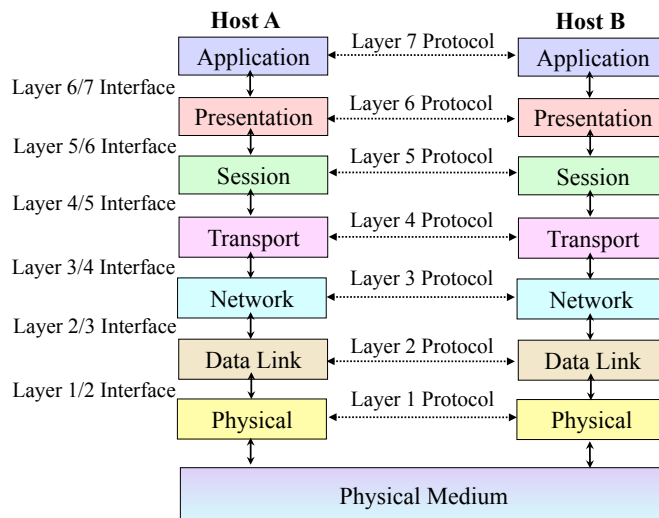
Ethernet (Xerox local area network)

- Local area network
 - up to 1024 stations
 - up to 2.8 km distance
 - 10Mbits/sec serially on shielded co-axial cable
 - 1.5Mbits/sec on twisted pair of copper pair
- Developed by Xerox in late 70s
 - still most common LAN right now
 - being displaced by fiber-optics (can't handle video/audio rates or make required service guarantees)
- High-level protocols to ensure reliable data transmission
- CSMA-CD: carrier sense multiple access with collision detection

The ISO OSI Reference Model



The ISO OSI Reference Model

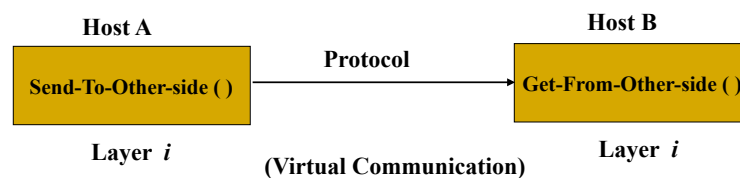


The ISO OSI Reference Model

- **Interface:** It defines which primitive operations and services the lower layer offers to the upper layer.
- **Peer:** The similar layer on a different machine.
- **Protocol:** It is a set of rules and conventions used by the layer to communicate with similar peer layer in another (remote) system.
- The peer processes communicate with each other using a *protocol*.
- A set of layers and protocols is called a **network architecture**.

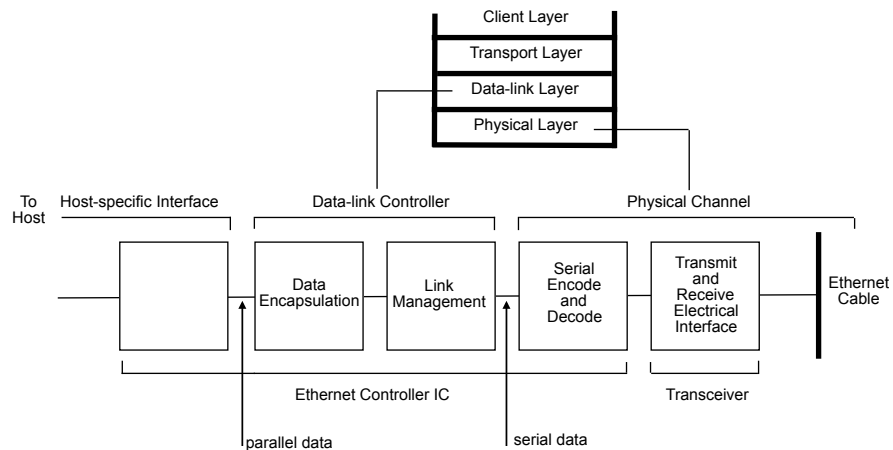
The ISO OSI Reference Model

- In reality, no data are directly transferred from layer i on one machine to layer i on another machine. Instead, each layer passes data and control information to the layer immediately below it, until the lowest layer is reached. Below layer 1 is the physical medium through which actual communication occurs.



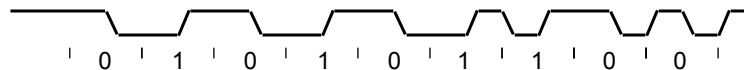
Ethernet layered organization

- Physical and data-link layers are our focus



Serial data format

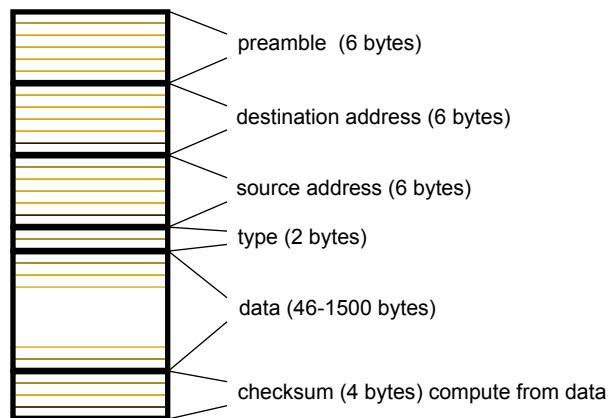
- Manchester encoding
 - signal and clock on one wire (XORed together)
 - "0" = low-going transition
 - "1" = high-going transition



- Extra transitions between 00 and 11 need to be filtered
 - preamble at beginning of data packet contains alternating 1s and 0s
 - allows receivers to get used to where important transitions should be and ignore extra ones (this is how synchronization is achieved)
 - preamble is 48 bits long: 10101...01011

Ethernet packet

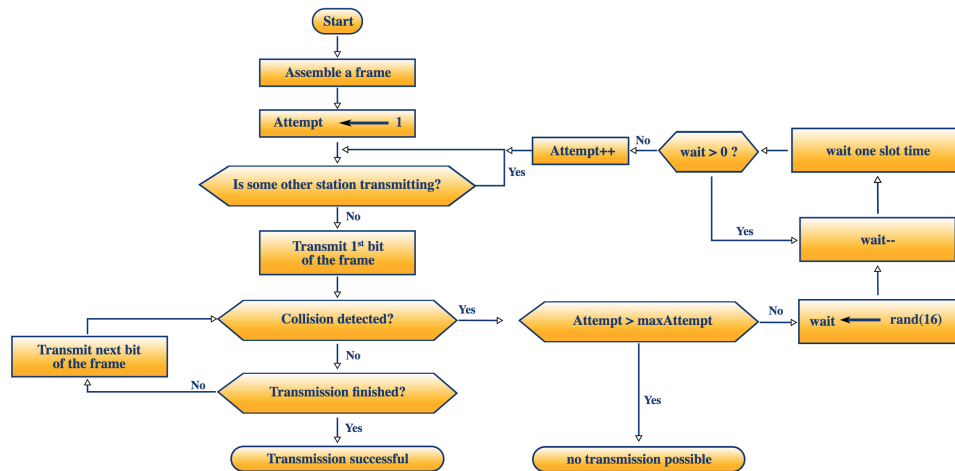
- Packets size: 64 to 1518 bytes + 6 bytes of preamble



Arbitration

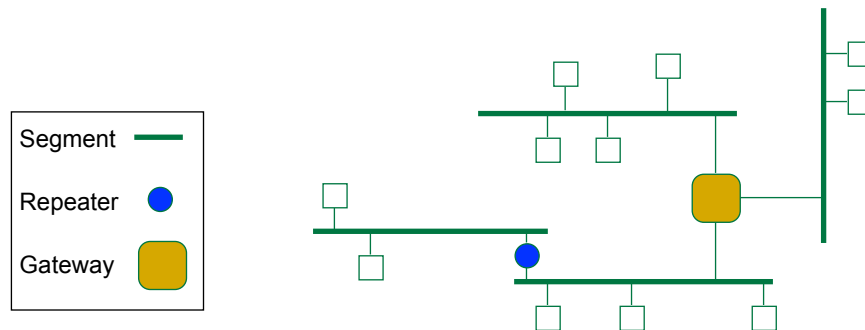
- Wait for line to be quiet for a while then transmit
 - detect collision
 - average value on wire should be exactly between 1 and 0
 - if not, then two transmitters are trying to transmit data
- If collision, stop transmitting
 - wait a random amount of time and try again
 - if collide again, pick a random number from a larger range (2x) and try again
- Exponential backoff on collision detection
- Try up to 16 times before reporting failure

Arbitration



Extending Ethernet

- Segments, repeaters, and gateways
 - segment: a single cable
 - repeater: transfers all messages on one segment to another and vice-versa
 - gateway: selectively forwards messages to other segments and helps isolate traffic



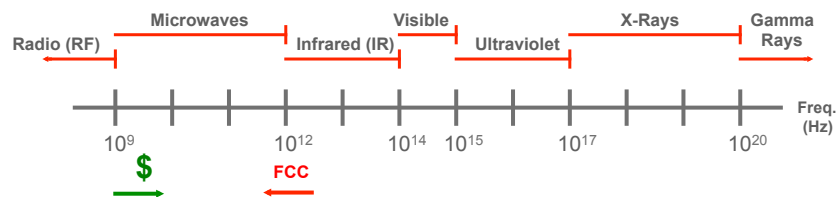
IrDA: Infrared Data Association

Infrared Data Association

- Consortium of over 160 companies
- Meet needs of the “mobile professional”
 - Short interactions with other devices (file transfer, printing)
 - Possibly using others’ peripherals (visiting a customer’ s office)
- Goals:
 - Suitable replacement for cables
 - Interoperability
 - Minimal cost
 - “Point-and-shoot” model (intended use and to reduce interference)
- History:
 - First standard developed in 1994
 - Revisions as recently as late 2011 (i.e., still active)

IrDA: Infrared Data Association

- Characteristics of IR:
 - Implementation costs rise significantly around 1-10 GHz
 - one important exception is IR at around 100 THz – very inexpensive
 - Signals above 100 GHz cannot penetrate walls
 - Most signals below 300 GHz are regulated by the FCC



Speed

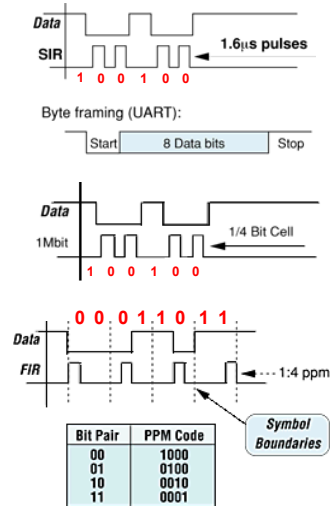
- Components include:
 - Transmitter (LED) and paired receiver (photodiode)
- IrDA supports wide range of speeds
 - 2400 bps to 4 Mbps
 - Exact physical-layer protocol used depends on speed of IrDA connection
 - Uses highest speed available on both devices
 - determined when connection is established
- Future promises even higher speeds:
 - 16-50 Mbps is not too far off
- Comparison to other wireless technologies:
 - Low-power RF (e.g., Bluetooth) slightly slower (.5 - 2 Mbps max)
 - Bound by walls, easy to control, intentional aspect
 - Much lower-power than high-speed RF (e.g., 802.11a at 50Mbps)

Low-speed Modulation

- Speed: 2400 bps - 115 kbps ("Serial Infrared", or SIR)
 - Only 0's require pulse (and thus power) ; pulse < full bit time
 - Standard UART byte framing
 - Pulse is constant 1.6 μ s long (so duty cycle varies with speed)
 - Average duty cycle: $\leq 9\%$

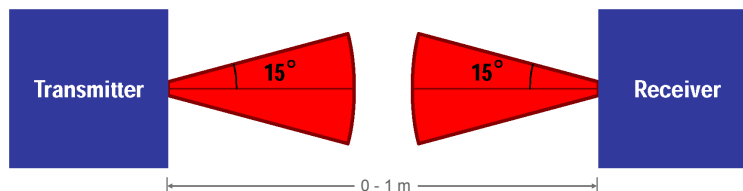
- Speed: 576 kbps - 1 Mbps
 - similar to SIR (pulse only for 0's ; pulse < full bit time)
 - pulse lasts 1/4 of bit time (so pulse varies with speed)
 - Average duty cycle: 12.5%

- Speed: 4 Mbps ("Fast Infrared", or FIR)
 - uses four-pulse-position-modulation scheme (4PPM)
 - pulse during exactly 1/4 of each symbol boundary
 - 4PPM makes synchronization easier to maintain
 - Duty cycle: 25% (independent of data)
 - Lowest power/bit



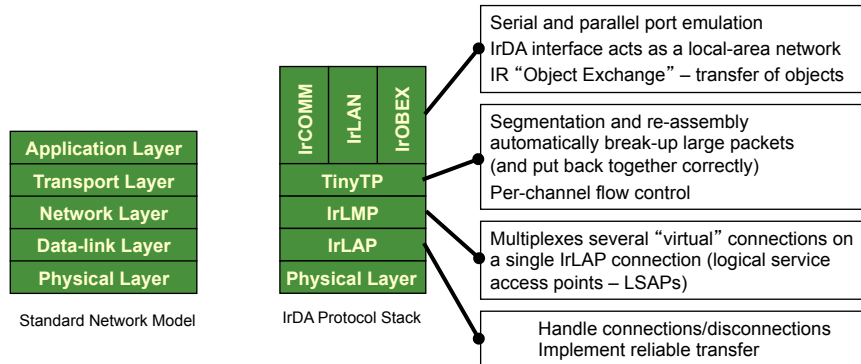
Range

- Linear:
 - IrDA standard requires 0-1 m
 - Realistically, some transceivers work at up to 10 m
- Angular:
 - Limited to a narrow cone (15° half-angle)
 - Done to help reduce interference between devices



IrDA Protocol Stack

- Analogous to the standard layered network model
- Consists of both required and optional components



Protocol Overhead

- Very simple model (point-to-point), so can expect reduced protocol overhead
- For layers in IrDA protocol stack, overhead per packet/frame is:
 - IrLAP = 2 bytes
 - IrLMP = 2 bytes
 - TinyTP = 1 byte

} **Total: 5 bytes**
- For perspective, compare to TCP/IP over Ethernet:
 - Ethernet = 18 bytes minimum
 - IP = 20 bytes
 - TCP = 20 bytes

} **Total: 58 bytes (minimum)**
- IrDA takes advantage of its simpler model, and keeps protocol overhead very low.