- Question:
  1. what is this part of the unit about, and
  2. how does it relate to the rest of the unit?
- Answer: it should be obvious that

$$\text{concurrent systems} \supset \text{computer networks} \Rightarrow \text{concurrency} \cup \text{communication}$$

  but beyond this, we need a motivating example ...

http://en.wikipedia.org/wiki/File:Ken_Thompson_(sitting)_and_Dennis_Ritchie_at_PDP-11_(2876612463).jpg

PDP-11 ↔ Teletype 33 (1)

- **TIA-232-F** [6] specifies a **communication medium** used to connect
  - a **Data Terminal Equipment (DTE)** or *master* device, e.g., a workstation, with
  - a **Data Communication Equipment (DCE)** or *slave* device, e.g., a MODEM

  and hence provide a (more abstract) **communication channel** (or **link**).
- Question: what terms can you think of to characterise this channel?

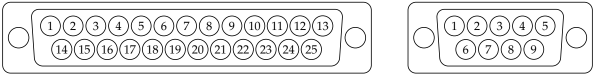
PDP-11 ↔ Teletype 33 (1)

- **TIA-232-F** [6] specifies a **communication medium** used to connect
  - a **Data Terminal Equipment (DTE)** or *master* device, e.g., a workstation, with
  - a **Data Communication Equipment (DCE)** or *slave* device, e.g., a MODEM

  and hence provide a (more abstract) **communication channel** (or **link**).
- Question: what terms can you think of to characterise this channel?
- (Incomplete) answer:
  - **serial** (i.e., can communicate 1-bit per-unit of time),
  - (upto) **full duplex** (i.e., communication can occur simultaneously in both directions),
  - **synchronous** *or* **asynchronous** (i.e., can require *or* avoid shared control wrt. timing),
  - **direct** (i.e., there are no intermediate devices),
  - **unicast** (i.e., a single device recieves any transmitted data),
  - ...

PDP-11 ↔ Teletype 33 (2) – Communication medium

- The medium itself is formed from
  1. an interface (from a choice of two)



PDP-11 ↔ Teletype 33 (2) – Communication medium

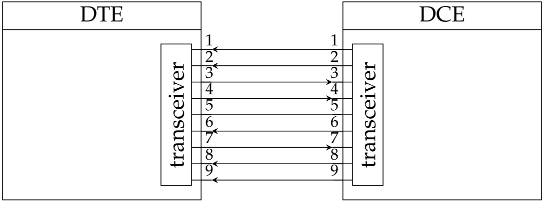
- The medium itself is formed from

2. a pin assignment which dictates how the interface is used

Class	Purpose	Mnemonic	Direction	9-pin	25-pin
Electrical properties	Reference (or signal) ground	GND	common	5	7
	Protective (or shield) ground	PG	common		1
Primary communication channel	Transmitted Data	TxD	DTE → DCE	3	2
	Recieved Data	RxD	DCE → DTE	2	3
	Request To Send	RTS	DTE → DCE	7	4
	Clear To Send	CTS	DCE → DTE	8	5
Secondary communication channel	Secondary Transmitted Data	STxD	DTE → DCE		14
	Secondary Recieved Data	SRxD	DCE → DTE		16
	Secondary Request To Send	SRTS	DTE → DCE		19
	Secondary Clear To Send	SCTS	DCE → DTE		13
Control and status	Data Set Ready	DSR	DCE → DTE	6	6
	Data Terminal Ready	DTR	DTE → DCE	4	20
	Carrier Detect	CD	DCE → DTE	1	8
	Secondary Carrier Detect	SCD	DCE → DTE		12
	Ring Indicator	RI	DCE → DTE	9	22
	Data Signal Rate		DTE → DCE		23
Timing	External Transmitter Clock	ETC	DTE → DCE		24
	Transmitter Clock	TC	DCE → DTE		15
	Reciever Clock	RC	DCE → DTE		17
Test and debug	Local Loop-back	LL	DTE → DCE		18
	Remote Loop-back	RL	DTE → DCE		21
	Test Mode	TM	DCE → DTE		25

PDP-11 ↔ Teletype 33 (3) – Communication medium

- The medium is managed using some form of **transceiver**, i.e.,



- Example: depending on the context, either end-point might
  1. rely on **Universal Asynchronous Receiver/Transmitter (UART)** hardware to provide a higher-level interface, or
  2. directly interface with low-level GPIO pins, ensuring correct voltage levels and timing in software (cf. bit-banging).

- Fact: TIA-232-F does *not* define a **communication protocol** ...
- ... the ~ 100 year old protocol we *still* use actually stems from the design of electronic typewriters (cf. teletype):

Quote

> *In printing telegraphy, the method of selective signaling which consists in operating a transmitter solely under local control to impart to a line character signals of uniform length and each comprising the same number of positive or negative impulses in quick succession and without perceptible spacing intervals between the impulses, initiating the operation of a selecting receiver switch mechanism in response to transmitted impulses at the beginning of each signal, timing the operation of the switch mechanism in synchronism with the transmitted impulses of each signal independently of the line circuit land restoring the same to a condition of rest at the completion of each signal.*
>
> – *Krum [7, Claim 1]*, 1918 *(!)*

- Translation:
  1. all (binary) data transmitted is encoded as discrete voltage levels (cf. "positive and negative impulses"),
  2. the data (cf. "character") has start and end markers added st. the receiver knows when transmission occurs,
  3. this allows asynchronous communication (cf. "local control").

- So ... the DTE and DCE pre-agree a set of **signalling parameters**, e.g.,

$$\begin{array}{lcl} \text{baud rate} & \in & \{110, 300, 600, 1200, 2400, 4800, 9600, 19200, \ldots\} \\ \text{number of data bits} & \in & \{5, 6, 7, 8\} \\ \text{parity type} & \in & \{\text{none}, \text{odd}, \text{even}\} \\ \text{number of stop bits} & \in & \{1, 1.5, 2\} \end{array}$$

  and, as such, agree how data is **framed**, i.e.,

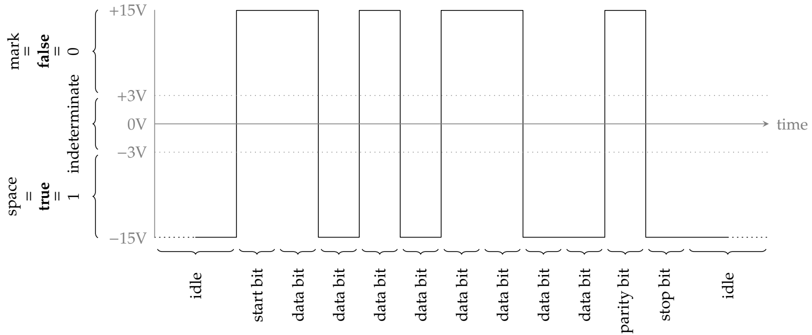
$$D \mapsto F = \text{start bit} \parallel D \parallel \text{parity bit} \parallel \text{stop bits}$$

  where

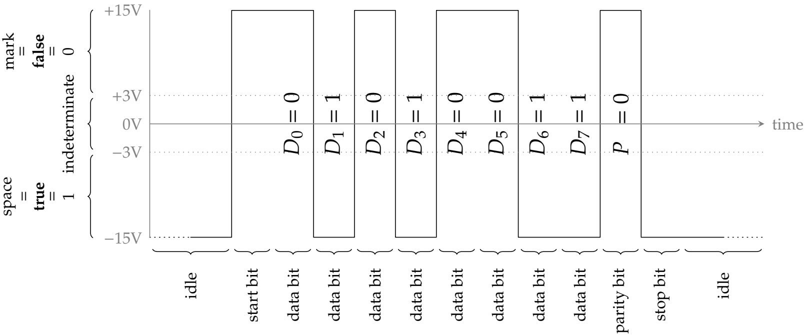
  - the **baud rate** the number of transitions per-second of a given signal (e.g., TxD or RxD),
  - the **start bit** mark the start of a frame,
  - the **stop bit(s)** mark the end of a frame,
  - the **data bits** are the data, and
  - the **parity type** supports (optional) error detection.

PDP-11 ↔ Teletype 33 (6) – Communication protocol

Low-level protocol



PDP-11 ↔ Teletype 33 (6) – Communication protocol
Low-level protocol



- We have

$$D = \langle D_0, D_1, D_2, D_3, D_4, D_5, D_6, D_7 \rangle = \langle 0, 1, 0, 1, 0, 0, 1, 1 \rangle \mapsto 202_{(10)}$$

noting that

$$\left(\bigoplus_{i=0}^{i<8} D_i \right) \oplus P = 0,$$

which we come back to later ...

- TIA-232-F supports (at least) two forms of **flow control**, namely
  1. hardware-based (RTS/CTS), and
  2. software-based (XON/XOFF)

  which you can think of as managing workload of the end-points.
- Example:
  - imagine the DTE is a (fast) workstation, and the DCE is a (slower) printer,
  - if the DCE cannot process the data transmitted fast enough, it will need to either buffer or discard data ...
  - ... *or* it could ask the DTE to slow down, or stop until it has caught up.

PDP-11 ↔ Teletype 33 (8) – Communication protocol

High(er)-level protocol

- RTS/CTS:
  - When the transmitter is ready to transmit, it sets RTS; the receiver primes itself to monitor data signals.
  - When the receiver is ready to receive, it sets CTS; the transmitter commences transmission.
  - If either end-point is unable or unwilling to continue, it clears the associated control signal (i.e., RTS or CTS) so the other end-point stops.

- XON/XOFF:
  - Define two special symbols, e.g., with ASCII

$$\begin{array}{lclclcl} XON & = & 19_{(10)} & = & 13_{(16)} & \mapsto & Ctrl-S \\ XOFF & = & 17_{(10)} & = & 11_{(16)} & \mapsto & Ctrl-Q \end{array}$$

    articulated as "transmit on" and "transmit off".
  - When the receiver is unable to accept more symbols, it sends the XOFF symbol; transmitter suspends transmission.
  - When the receiver is able to accept more symbols, it sends the XON symbol; transmitter resumes transmission.

- Based on some simple coding theory, i.e.,

Definition (**parity**)

The **parity** of an n-bit sequence X is defined as

$$\mathcal{P}(X) = \sum_{i=0}^{i<n} X_i \pmod 2 = \bigoplus_{i=0}^{i<n} X_i,$$

st. X has **even parity** (resp. **odd parity**) if $\mathcal{P}(X) = 0$ (resp. $\mathcal{P}(X) = 1$).

Definition (**parity code**)

An even (resp. odd) **parity code** appends a **parity bit** P to some sequence X st. $X \parallel P$ has even (resp. odd) parity:

$$\text{even parity} \quad \left\{ \begin{array}{lclcl} \mathcal{P}(X) = 0 & \Rightarrow & P = 0 & \rightsquigarrow & \mathcal{P}(X \parallel P) = 0 \\ \mathcal{P}(X) = 1 & \Rightarrow & P = 1 & \rightsquigarrow & \mathcal{P}(X \parallel P) = 0 \end{array} \right.$$

$$\text{odd parity} \quad \left\{ \begin{array}{lclcl} \mathcal{P}(X) = 0 & \Rightarrow & P = 1 & \rightsquigarrow & \mathcal{P}(X \parallel P) = 1 \\ \mathcal{P}(X) = 1 & \Rightarrow & P = 0 & \rightsquigarrow & \mathcal{P}(X \parallel P) = 1 \end{array} \right.$$

a (limited) form of **error detection** is possible

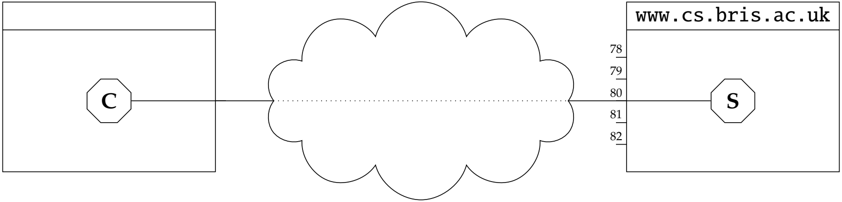
- the transmitter computes $\mathcal{P}(D)$, then appends an appropriate P before transmitting $D \parallel P$,
- the receiver recomputes $\mathcal{P}(D \parallel P)$ and signals an error if this does *not* match expectation.

Conclusions

- Take away points:
  - TIA-232-F is a (very) *specific* example of more general concepts and requirements: the goal is a clear, *general* understanding.
  - Put simply, we want to understand how we can communicate data reliably and efficiently via a computer network ...

Conclusions

- Remit:
  - understand a simple(ish) computer network

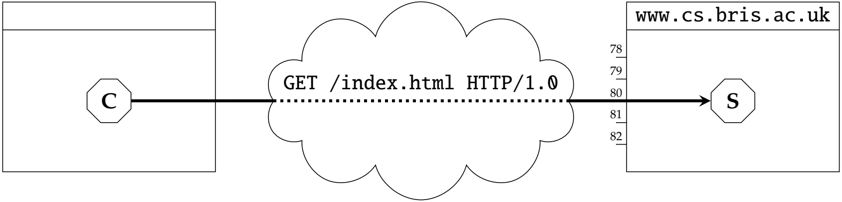


wrt. support for a HTTP transaction, *but*

  - limit the detail and volume of coverage to fit allocated time.
- *Why*?!
  1. some of you may
     - develop network-related hardware or software, or
     - work in network operations,
  2. *most* of you will develop hardware or software that *depends* on a network, and
  3. many of the concepts and techniques *generalise*.

Conclusions

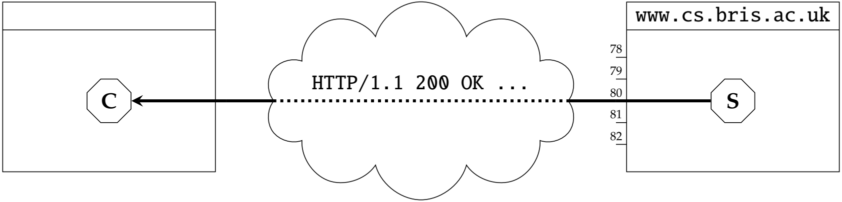
- Remit:
  - understand a simple(ish) computer network



  wrt. support for a HTTP transaction, *but*
  - limit the detail and volume of coverage to fit allocated time.
- *Why*?!
  1. some of you may
     - develop network-related hardware or software, or
     - work in network operations,
  2. *most* of you will develop hardware or software that *depends* on a network, and
  3. many of the concepts and techniques *generalise*.

Conclusions

- Remit:
  - understand a simple(ish) computer network



  wrt. support for a HTTP transaction, *but*
  - limit the detail and volume of coverage to fit allocated time.
- *Why*?!
  1. some of you may
     - develop network-related hardware or software, or
     - work in network operations,
  2. *most* of you will develop hardware or software that *depends* on a network, and
  3. many of the concepts and techniques *generalise*.

References

[1] Wikipedia: Asynchronous serial communication.
http://en.wikipedia.org/wiki/Asynchronous_start-stop.

[2] Wikipedia: Flow control.
http://en.wikipedia.org/wiki/Flow_control_(data).

[3] Wikipedia: Parity bit.
http://en.wikipedia.org/wiki/Parity_bit.

[4] Wikipedia: Serial port.
http://en.wikipedia.org/wiki/Serial_port.

[5] Wikipedia: Teleprinter.
http://en.wikipedia.org/wiki/Teletypewriter.

[6] Interface between data terminal equipment and data circuit terminating equipment employing serial binary data interchange.
Telecommunications Industry Association (TIA) TIA-232-F, 2002.
http://www.tiaonline.org/standards/.

[7] H.L. Krum.
Electric selective system.
U.S. Patent 1,286,351.
http://www.google.com/patents/US1286351.