Asynchronous and Synchronous Transmission
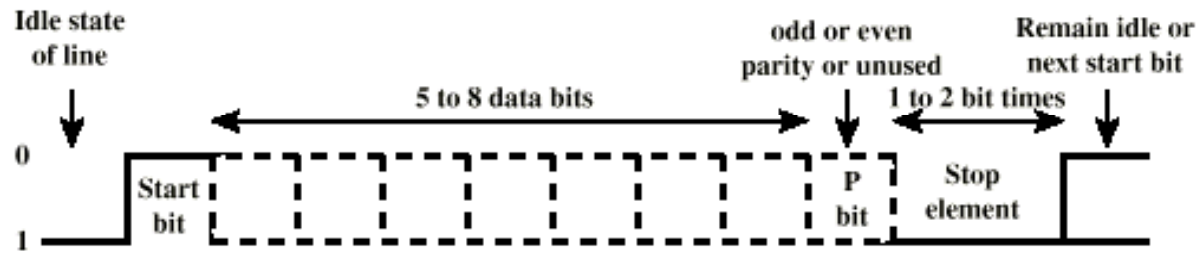
Asynchronous and Synchronous Transmission

⌘ Timing problems require a mechanism to synchronize the transmitter and receiver

⌘ Two solutions
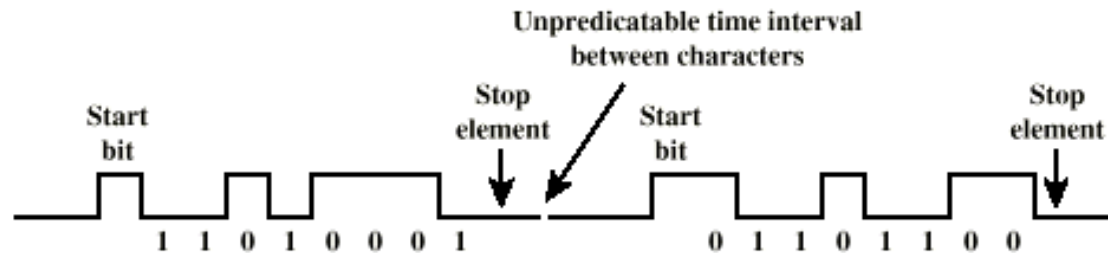- Asynchronous
- Synchronous

Asynchronous

- ⌘ Data transmitted on character at a time
  - ⌃ 5 to 8 bits
- ⌘ Timing only needs maintaining within each character
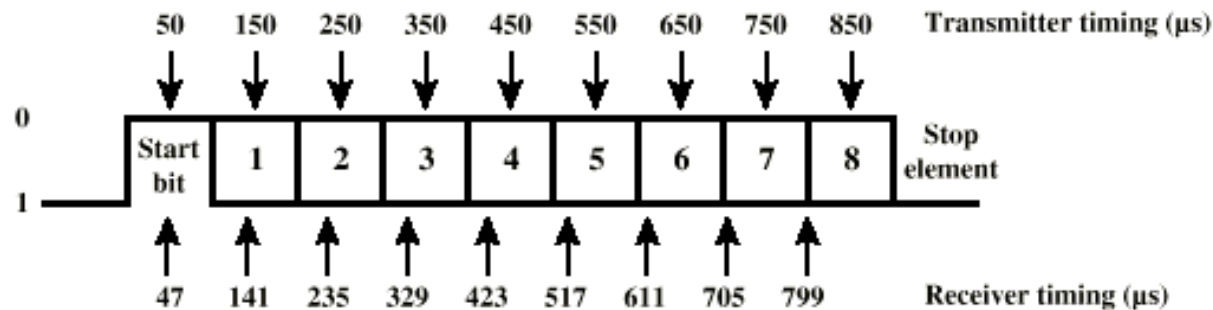- ⌘ Resync with each character

Asynchronous (diagram)

Idle state of line

odd or even parity or unused

Remain idle or next start bit

5 to 8 data bits

1 to 2 bit times

0

Start bit

P bit

Stop element

1

(a) Character format

Unpredicatable time interval between characters

Stop element

Start bit

Start bit

Stop element

1 1 0 1 0 0 0 1

0 1 1 0 1 1 0 0

(b) 8-bit asynchronous character stream

| 50 | 150 | 250 | 350 | 450 | 550 | 650 | 750 | 850 | Transmitter timing (µs) |

0

| Start bit | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | Stop element |

1

| 47 | 141 | 235 | 329 | 423 | 517 | 611 | 705 | 799 | Receiver timing (µs) |

(c) Effect of timing error

Asynchronous - Behavior

- In a steady stream, interval between characters is uniform (length of stop element)
- In idle state, receiver looks for transition 1 to 0
- Then samples next seven intervals (char length)
- Then looks for next 1 to 0 for next char

- Simple
- Cheap
- Overhead of 2 or 3 bits per char (~20%)
- Good for data with large gaps (keyboard)

Synchronous - Bit Level

- Block of data transmitted without start or stop bits
- Clocks must be synchronized
- Can use separate clock line
  - Good over short distances
  - Subject to impairments
- Embed clock signal in data
  - Manchester encoding
  - Carrier frequency (analog)

Synchronous - Block Level

- Need to indicate start and end of block
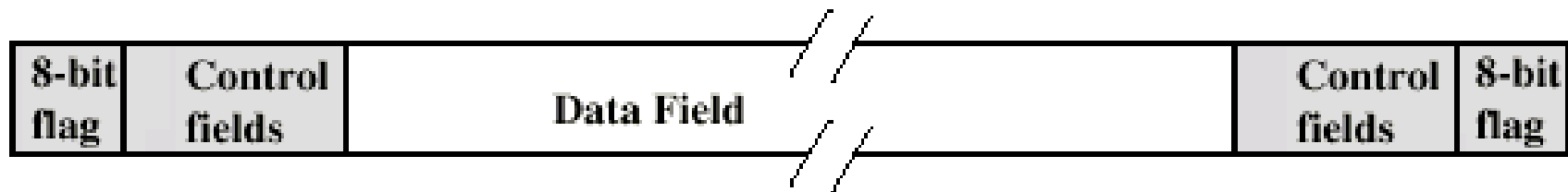- Use preamble and postamble
    - e.g. series of SYN (hex 16) characters
    - e.g. block of 11111111 patterns ending in 11111110

- More efficient (lower overhead) than async

Synchronous (diagram)

8-bit flag	Control fields	Data Field	Control fields	8-bit flag

Line Configuration

- ⌘ Topology
  - ⌃ Physical arrangement of stations on medium
  - ⌃ Point to point
  - ⌃ Multi point
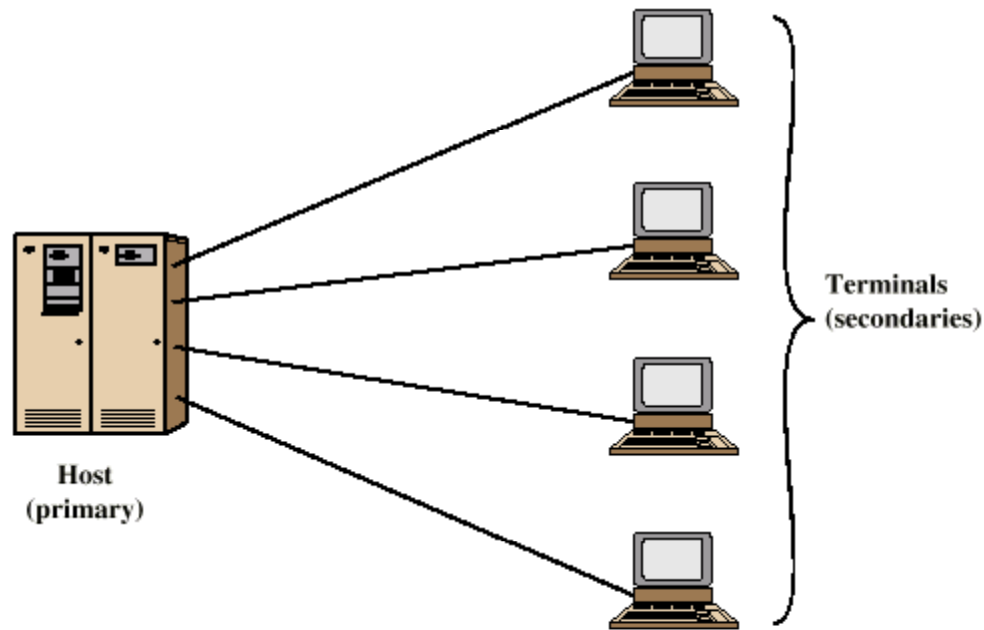   - ☒ Computer and terminals, local area network
- ⌘ Half duplex
  - ⌃ Only one station may transmit at a time
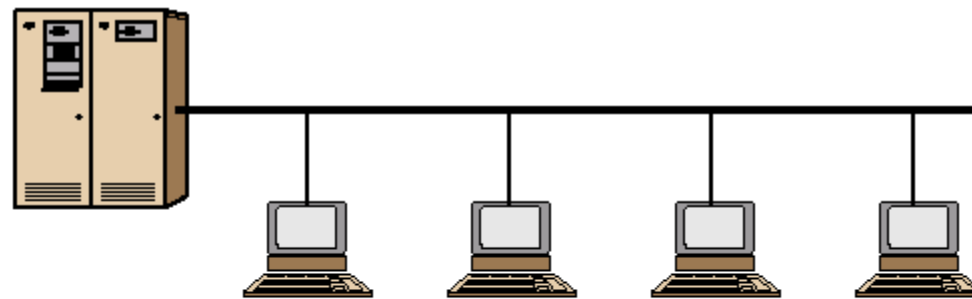 - ⌃ Requires one data path
- ⌘ Full duplex
  - ⌃ Simultaneous transmission and reception between two stations
  - ⌃ Requires two data paths (or echo canceling)

Traditional Configurations

Host
(primary)

Terminals
(secondaries)

(a) Point-to-point

(b) Multipoint

Interfacing

- Data processing devices (or data terminal equipment, DTE) do not (usually) include data transmission facilities

- Need an interface called data circuit terminating equipment (DCE)
  - e.g. modem, NIC

- DCE transmits bits on medium

- DCE communicates data and control info with DTE
  - Done over interchange circuits
  - Clear interface standards required

Characteristics of Interface

⌘ Mechanical
  ⌂ Connection plugs
⌘ Electrical
  ⌂ Voltage, timing, encoding
⌘ Functional
  ⌂ Data, control, timing, grounding
⌘ Procedural
  ⌂ Sequence of events

V.24/EIA-232-F

⌘ ITU-T v.24

⌘ Only specifies functional and procedural

  ⬚ References other standards for electrical and mechanical

⌘ EIA-232-F (USA)

  ⬚ RS-232

  ⬚ Mechanical ISO 2110

  ⬚ Electrical v.28

  ⬚ Functional v.24

  ⬚ Procedural v.24

Mechanical Specification

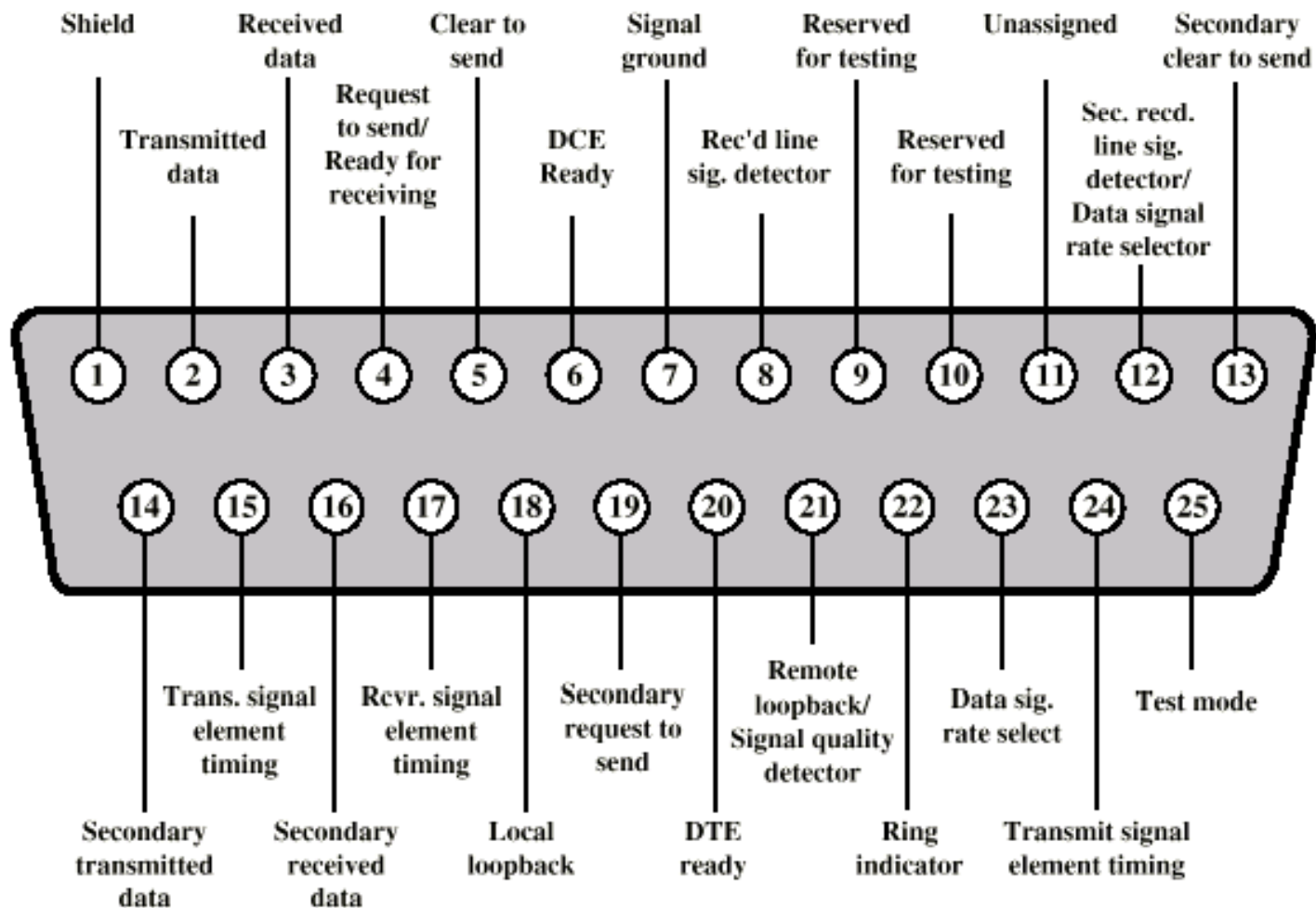Figure 6.5   Pin Assignments for V.24/EIA-232 (DTE Connector Face)

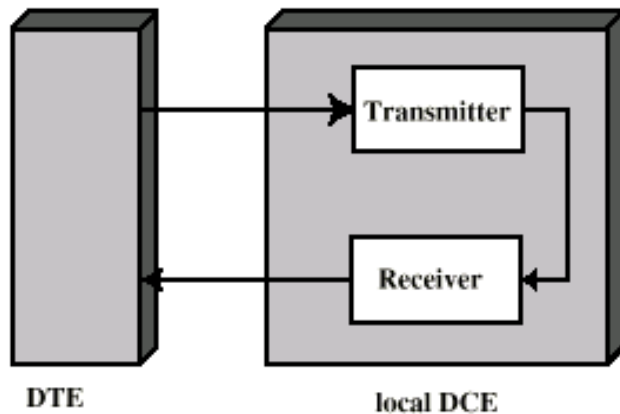# Electrical Specification

- Digital signals
- Values interpreted as data or control, depending on circuit
- More than -3v is binary 1, more than +3v is binary 0 (NRZ-L)
- Signal rate < 20kbps
- Distance <15m
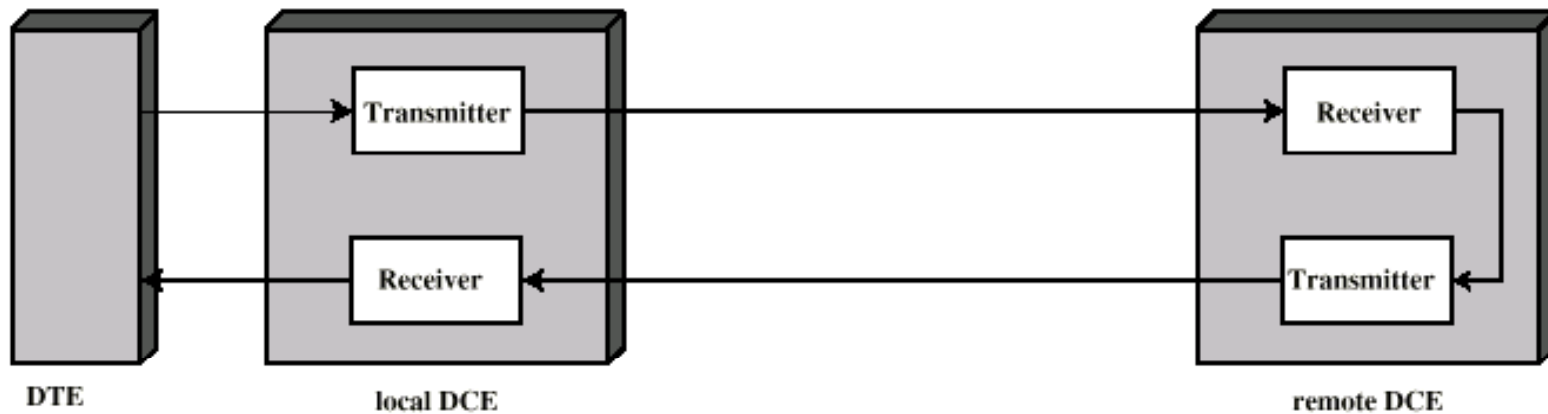- For control, more than-3v is off, +3v is on

Functional Specification

⌘ (See table in Stallings chapter 6)

Local and Remote Loopback

(a) Local loopback Testing

(b) Remote loopback Testing
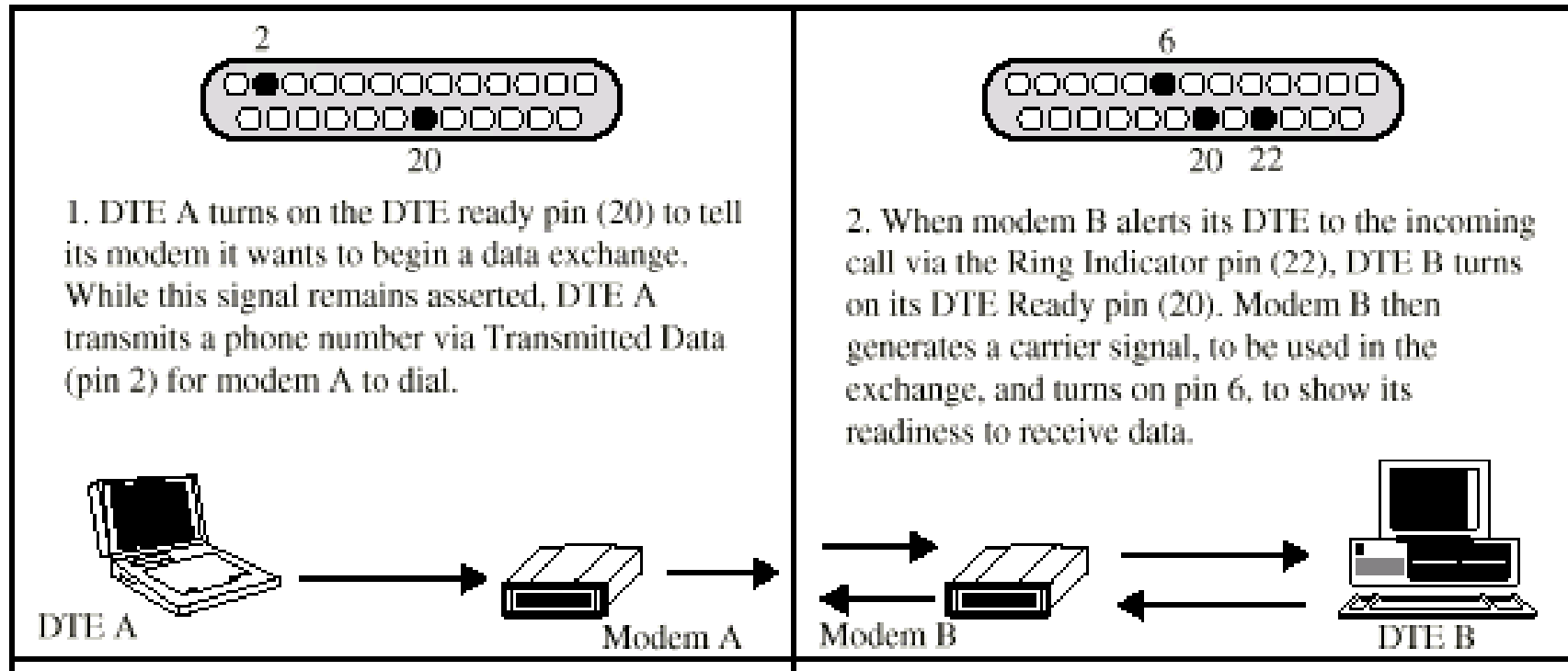
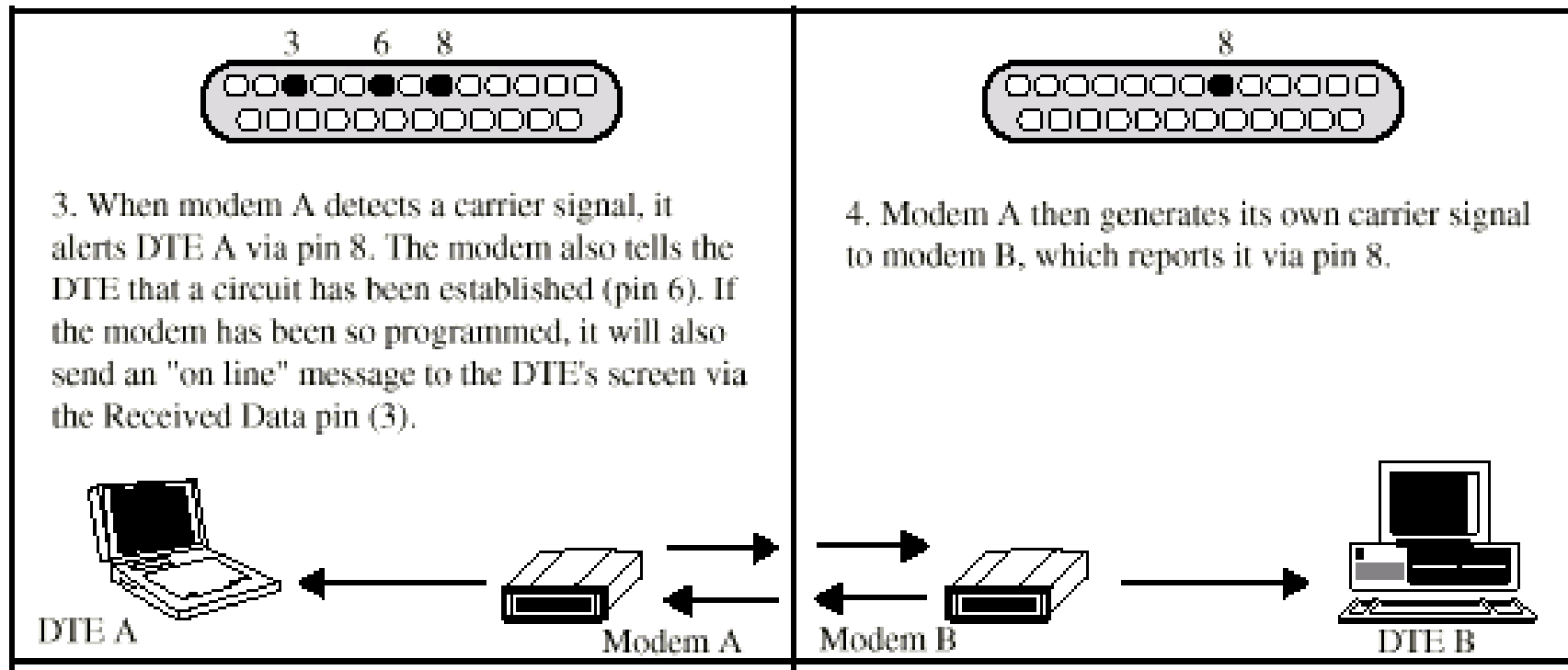# Procedural Specification

- E.g. Asynchronous private line modem
- When turned on and ready, modem (DCE) asserts DCE ready
- When DTE ready to send data, it asserts Request to Send
  - Also inhibits receive mode in half duplex
- Modem responds when ready by asserting Clear to send
- DTE sends data
- When data arrives, local modem asserts Receive Line Signal Detector and delivers data

Dial Up Operation (1)

2

1. DTE A turns on the DTE ready pin (20) to tell its modem it wants to begin a data exchange. While this signal remains asserted, DTE A transmits a phone number via Transmitted Data (pin 2) for modem A to dial.

20

DTE A

Modem A

6

2. When modem B alerts its DTE to the incoming call via the Ring Indicator pin (22), DTE B turns on its DTE Ready pin (20). Modem B then generates a carrier signal, to be used in the exchange, and turns on pin 6, to show its readiness to receive data.

20   22

Modem B

DTE B

Dial Up Operation (2)

3. When modem A detects a carrier signal, it alerts DTE A via pin 8. The modem also tells the DTE that a circuit has been established (pin 6). If the modem has been so programmed, it will also send an "on line" message to the DTE's screen via the Received Data pin (3).

DTE A

Modem A

4. Modem A then generates its own carrier signal to modem B, which reports it via pin 8.

Modem B

DTE B

Dial Up Operation (3)

2   4 5

3

5. When it wishes to send data, DTE A activates Request to Send (pin 4). Modem A responds with Clear to Send (pin 5). DTE A sends data (pulses representing 1s and 0s) to modem A via the Transmitted Data pin (2). Modem A modulates the pulses to send the data over its analog carrier signal.

6. Modem B reconverts the signal to digital form and sends it to DTE B via the Received Data pin (3).

DTE A

Modem A
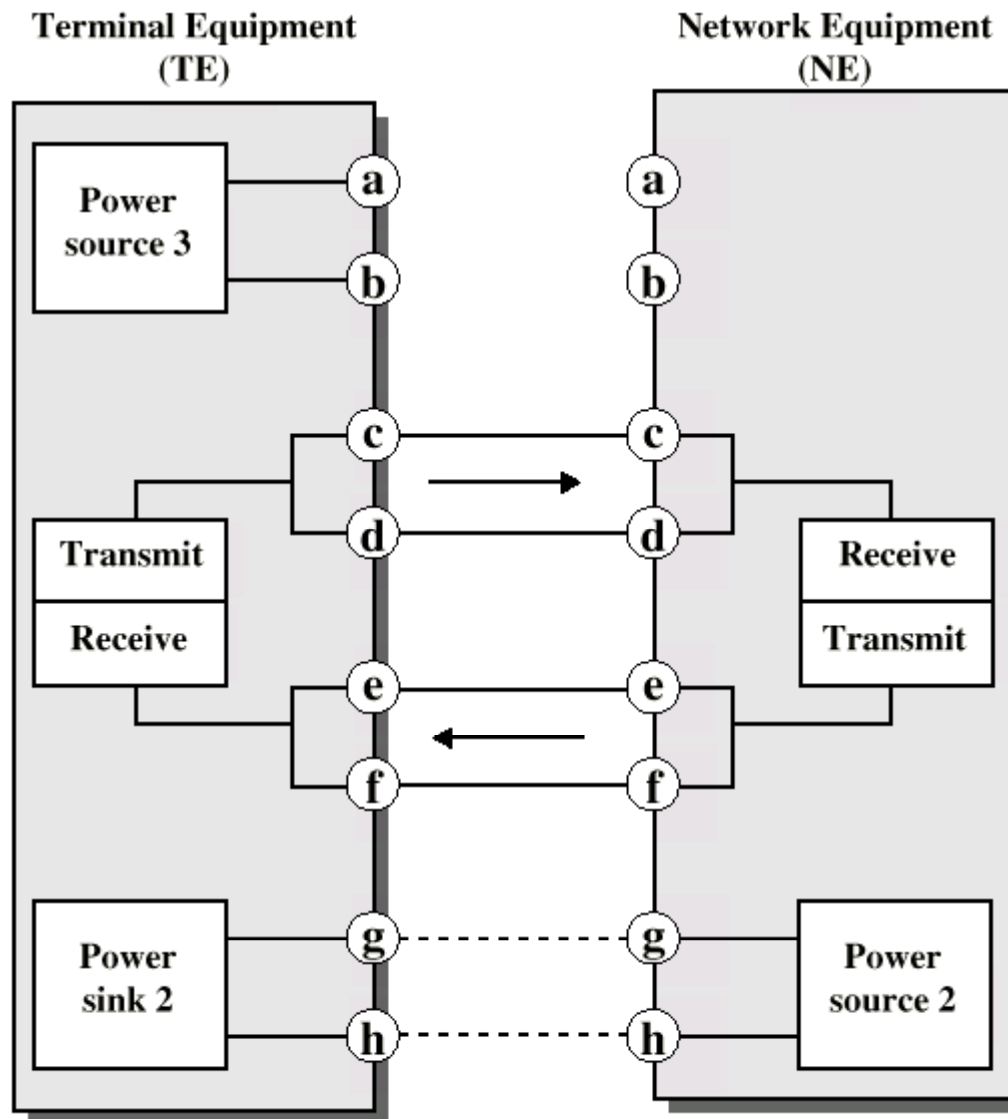
Modem B

DTE B

Null Modem

ISDN Physical Interface Diagram

ISDN Physical Interface

- ⌘ Connection between terminal equipment (c.f. DTE) and network terminating equipment (c.f. DCE)
- ⌘ ISO 8877
- ⌘ Cables terminate in matching connectors with 8 contacts
- ⌘ Transmit/receive carry both data and control

ISDN Electrical Specification

⌘ Balanced transmission

⌃ Carried on two lines, e.g. twisted pair

⌃ Signals as currents down one conductor and up the other

⌃ Differential signaling

⌃ Value depends on direction of voltage

⌃ Tolerates more noise and generates less

⌃ (Unbalanced, e.g. RS-232 uses single signal line and ground)

⌃ Data encoding depends on data rate

⌃ Basic rate 192kbps uses pseudoternary

⌃ Primary rate uses alternative mark inversion (AMI) and B8ZS or HDB3

Foreground Reading

- Stallings chapter 6
- Web pages from ITU-T on v. specification
- Web pages on ISDN