Internet Protocols EBU5403
The Data Link Layer Part I
C4

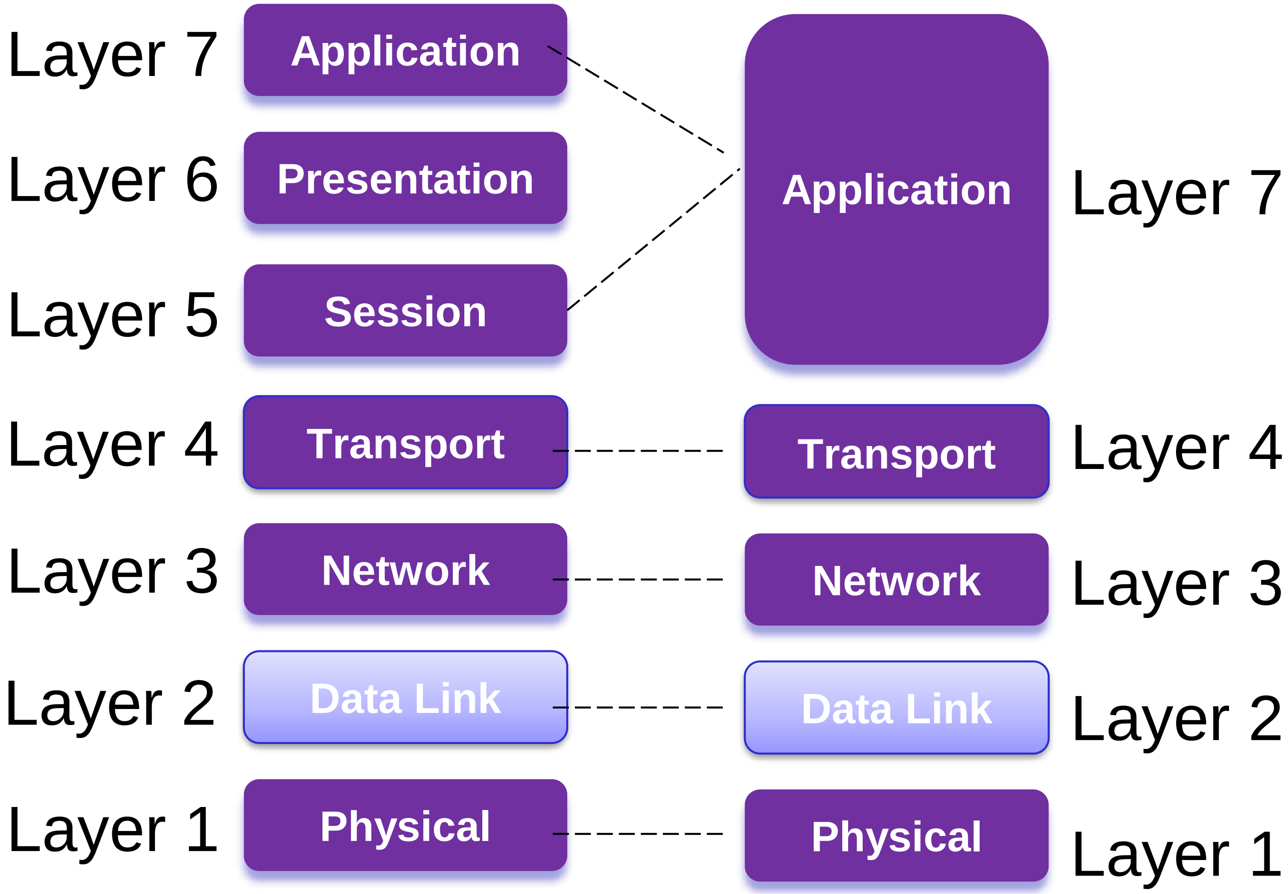
Michael Chai (michael.chai@qmul.ac.uk)
Richard Clegg (r.clegg@qmul.ac.uk)
Cunhua Pan (c.pan@qmul.ac.uk)

<table>
<tr><th></th><th>Part I</th><th>Part 2</th><th>Part 3</th><th>Part 4</th></tr>
<tr><td>Ecommerce + Telecoms I</td><td colspan="2">Richard Clegg</td><td colspan="2" rowspan="2">Cunhua Pan</td></tr>
<tr><td>Telecoms 2</td><td colspan="2">Michael Chai</td></tr>
</table>

Structure of course

- Part A
  - Introduction to IP Networks
  - The Transport layer (part 1)
- Part B
  - The Transport layer (part II)
  - The Network layer (part I)
  - Class test
- Part C
  - The Network layer (part II)
  - The Data link layer (part I)
- Part D
  - The Data link layer (part II)
  - Network management and security
  - Class test

Data Link Layer



Link layer, LANs: outline

6.1 introduction, services

6.2 error detection, correction

6.3 multiple access protocols

6.4 LANs
- addressing, ARP
- Ethernet
- switches
- VLANS

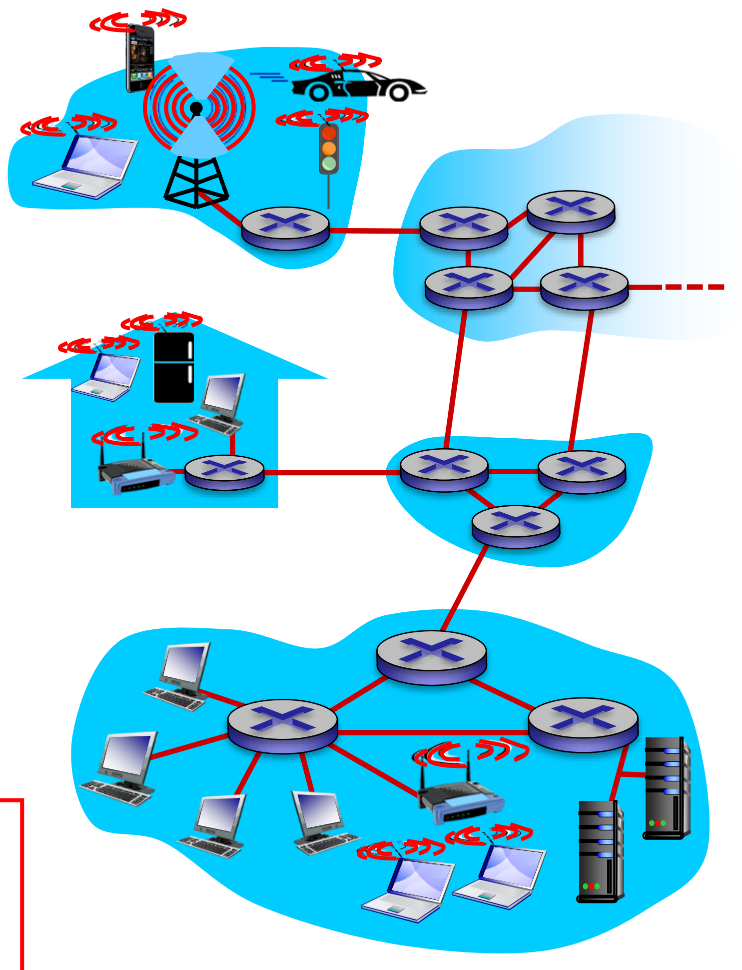
6.5 link virtualization: MPLS

6.6 a day in the life of a web request

Link layer: introduction

terminology:

- hosts and routers: nodes
- communication channels that connect adjacent nodes along communication path: links
  - wired links
  - wireless links
  - LANs
- layer-2 packet: *frame,* encapsulates datagram



data-link layer has responsibility of transferring datagram from one node to *physically adjacent* node over a link

Link layer: context

- datagram transferred by different link protocols over different links:
  - e.g., Ethernet on first link, frame relay on intermediate links, 802.11 on last link
- each link protocol provides different services
  - e.g., may or may not provide rdt over link

Link layer services

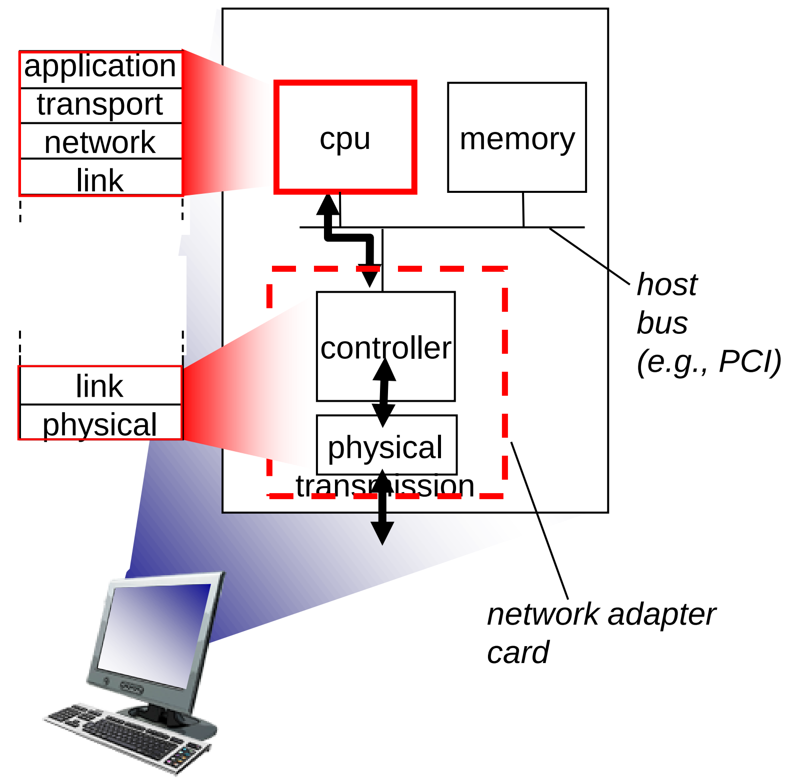
- *framing, link access:*
  - encapsulate datagram into frame, adding header, trailer
  - channel access if shared medium
  - "MAC" addresses used in frame headers to identify source, destination
    - different from IP address!
- *reliable delivery between adjacent nodes*
  - we learned how to do this already (transport layer)!
  - seldom used on low bit-error link (fiber, some twisted pair)
  - wireless links: high error rates
    - *Q:* why both link-level and end-end reliability?

Link layer services (more)

- *flow control:*
  - pacing between adjacent sending and receiving nodes
- *error detection:*
  - errors caused by signal attenuation, noise.
  - receiver detects presence of errors:
    - signals sender for retransmission or drops frame
- *error correction:*
  - receiver identifies *and corrects* bit error(s) without resorting to retransmission
- *half-duplex and full-duplex*
  - with half duplex, nodes at both ends of link can transmit, but not at same time

Where is the link layer implemented?

- in each and every host
- link layer implemented in "adaptor" (aka *network interface card* NIC) or on a chip
  - Ethernet card, 802.11 card; Ethernet chipset
  - implements link, physical layer
- attaches into host's system buses
- combination of hardware, software, firmware



Link layer, LANs: outline

6.1 introduction, services

6.2 error detection, correction

6.3 multiple access protocols

6.4 LANs
- addressing, ARP
- Ethernet
- switches
- VLANS

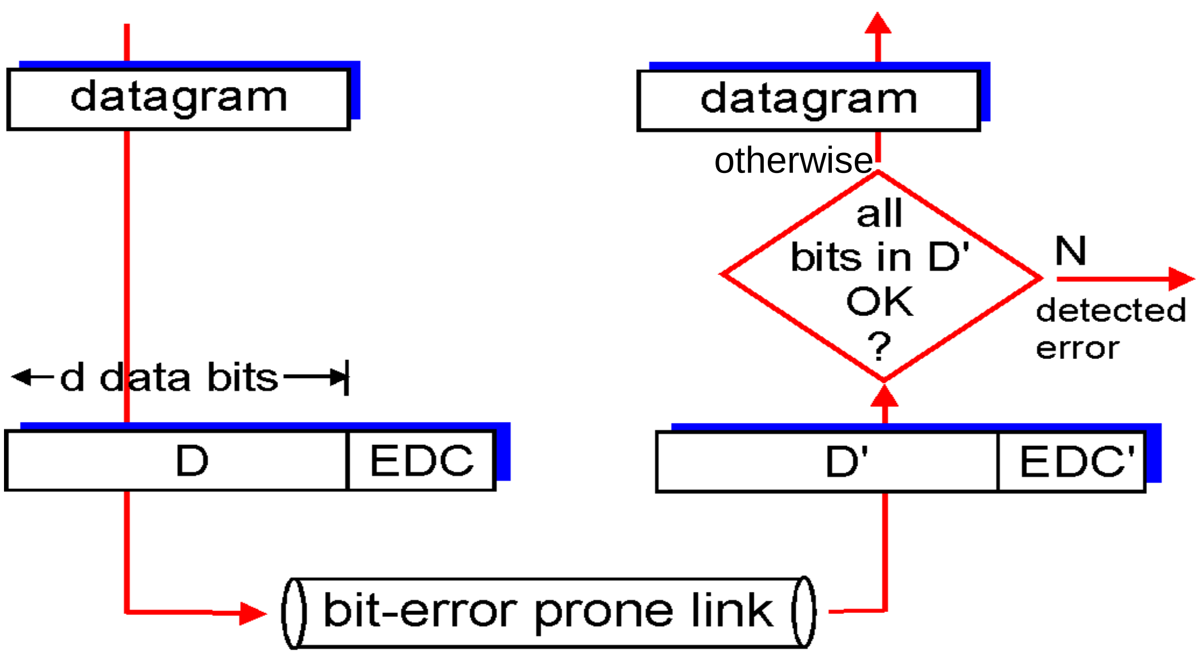
6.5 link virtualization: MPLS

6.6 a day in the life of a web request

Error detection

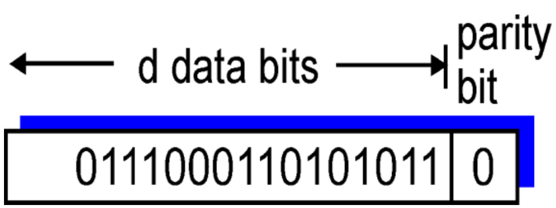
EDC= Error Detection and Correction bits (redundancy)
D = Data protected by error checking, may include header fields

- Error detection not 100% reliable!
  - protocol may miss some errors, but rarely
  - larger EDC field yields better detection and correction

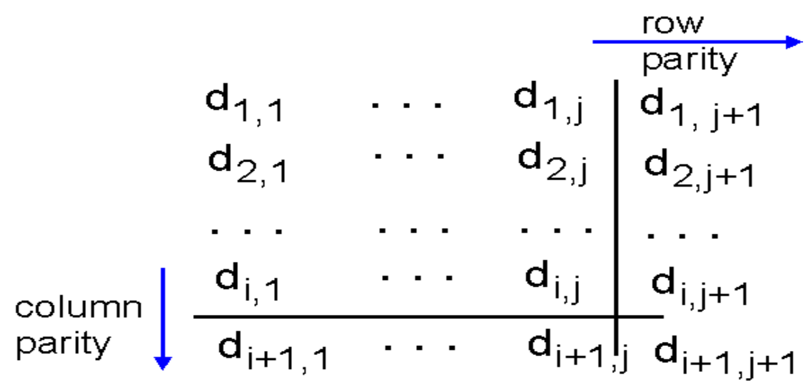


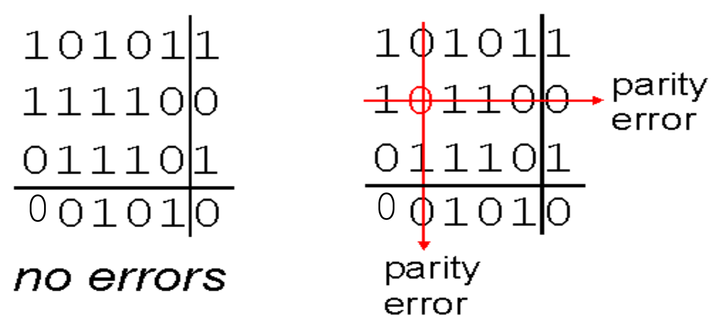
Parity checking

single bit parity:
- *d*etect single bit errors



two-dimensional bit parity:
- detect and correct single bit errors





no errors

correctable single bit error

* Check out the online interactive exercises for more examples: http://gaia.cs.umass.edu/kurose_ross/interactive/

Internet checksum (review)

goal: detect "errors" (e.g., flipped bits) in transmitted packet (note: used at transport layer only)

sender:

- treat segment contents as sequence of 16-bit integers
- checksum: addition (1's complement sum) of segment contents
- sender puts checksum value into UDP checksum field

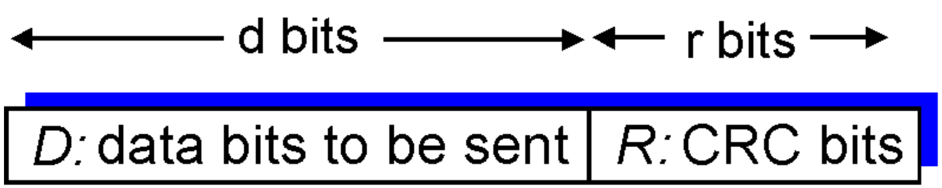
receiver:

- compute checksum of received segment
- check if computed checksum equals checksum field value:
  - NO - error detected
  - YES - no error detected. *But maybe errors nonetheless?*

Real layer 2 systems: Cyclic Redundancy Check (CRC) algorithm has good protection against bursts of errors.

Cyclic redundancy check

- more powerful error-detection coding
- view data bits, D, as a binary number
- choose r+1 bit pattern (generator), G
- goal: choose r CRC bits, R, such that
  - <D,R> exactly divisible by G (modulo 2)
  - receiver knows G, divides <D,R> by G. If non-zero remainder: error detected!
- widely used in practice (Ethernet, 802.11 WiFi, ATM)



bit pattern

$D * 2^r$ XOR R

mathematical formula

CRC example

want:

$D \cdot 2^r \text{ XOR } R = nG$

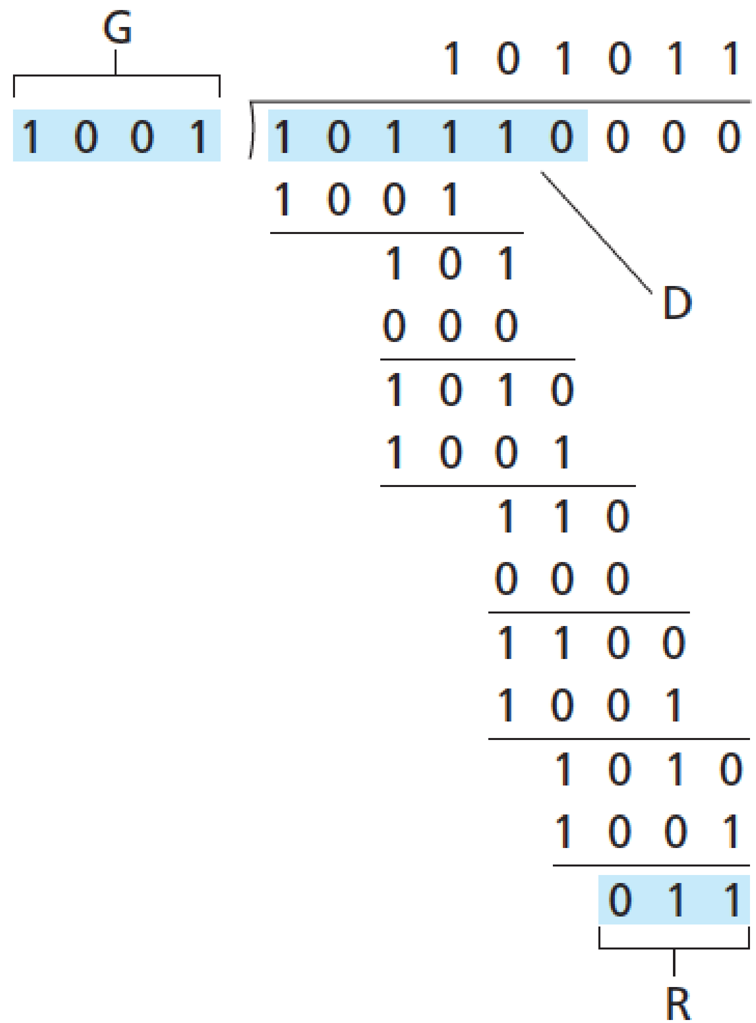
equivalently:

$D \cdot 2^r = nG \text{ XOR } R$

equivalently:

if we divide $D \cdot 2^r$ by G, want remainder R to satisfy:

$$R = remainder\left[\frac{D \cdot 2^r}{G}\right]$$



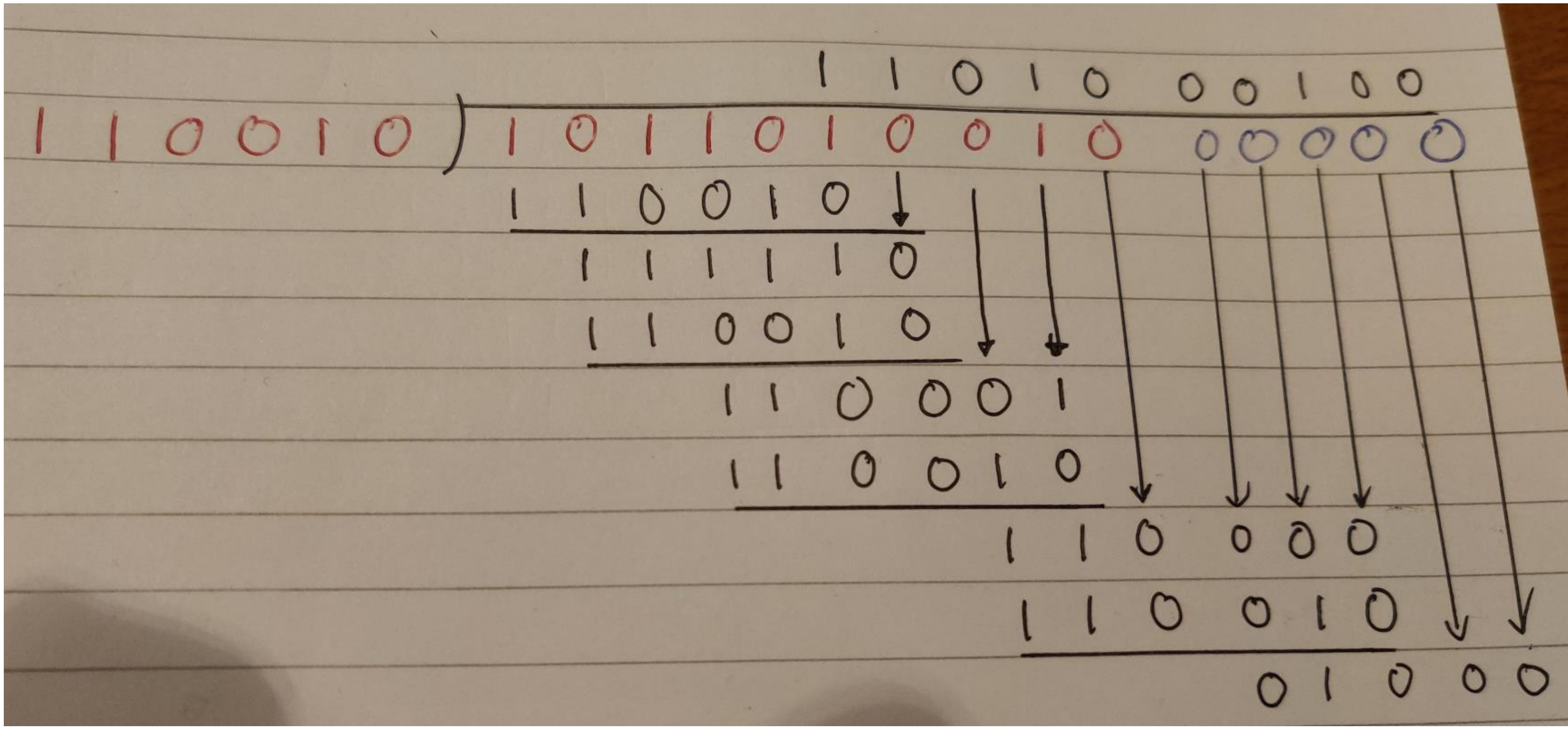
* Check out the online interactive exercises for more examples: http://gaia.cs.umass.edu/kurose_ross/interactive/

Test your understanding

Using CRC, find CRC bit, **R**.

- Data bits, **D**= 1011010010
- Generator bits, **G**=110010
- CRC bits, **R** =?
- Frame =?

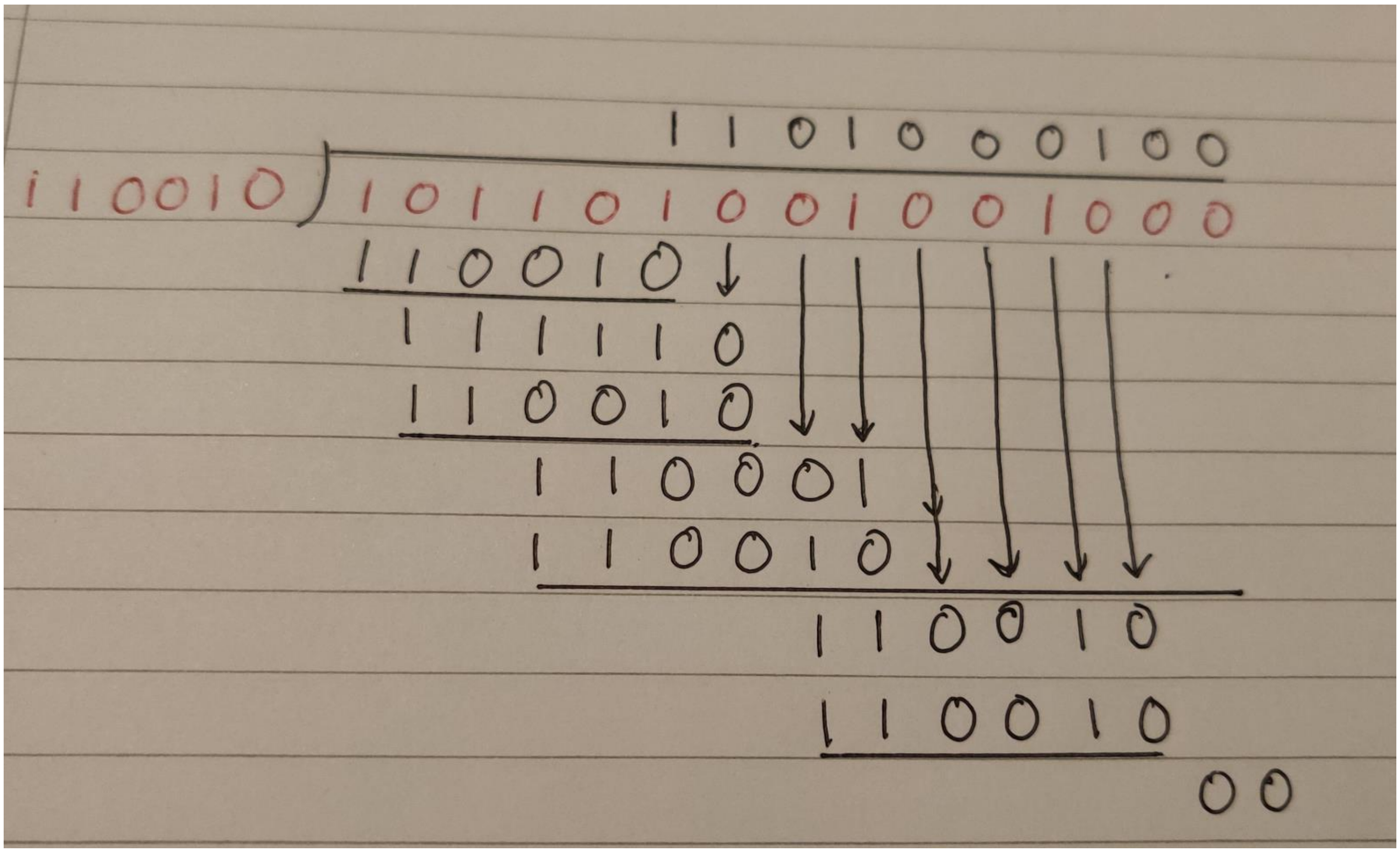
Test your understanding (Solution)



Frame: 101110100101000

Test your understanding (Solution)

Frame: 1011101001001000; G = 110010



What have we learned?

- Data link layer has responsibility of transferring datagram from one node to physically adjacent node over a link.
- Key topics we learnt in this part:
  - Data link services: framing, link access and reliable delivery between adjacent nodes.
  - Error detection methods: Error Detection and Correction, Parity Check and CRC