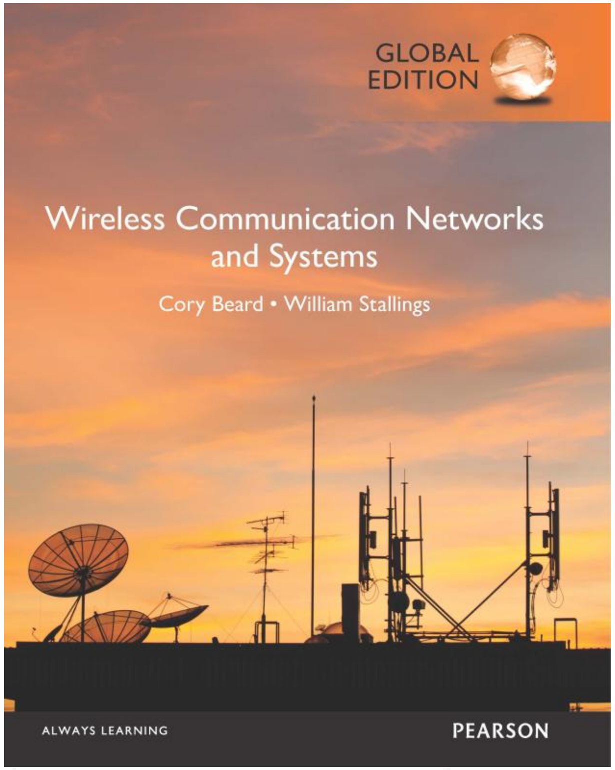
MOBILE SYSTEM-HT25 LECTURE 11: CODING AND ERROR CONTROL

Azra Abtahi

Email: azra.abtahi-fahliani@mau.se

Faculty of Technology and Society Department of Computer Science and Media Technology Malmö University

Most slides are primarily adapted from Beard & Stallings (2016), Wireless Communication Networks and Systems (Chapter 10)


GLOBAL EDITION
Wireless Communication Networks and Systems
Cory Beard • William Stallings
ALWAYS LEARNING
PEARSON

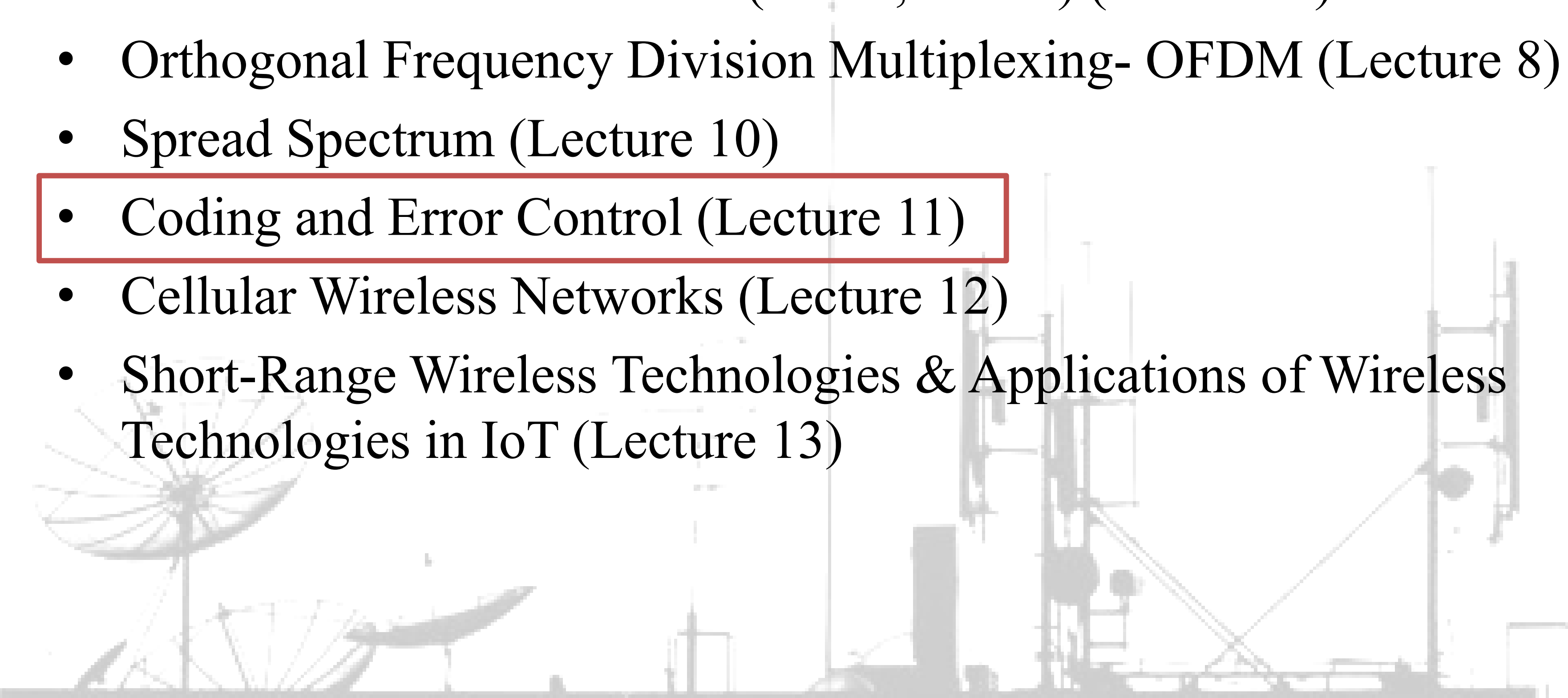
Wireless Communication Networks and Systems

1st edition, Global edition

Cory Beard, William Stallings

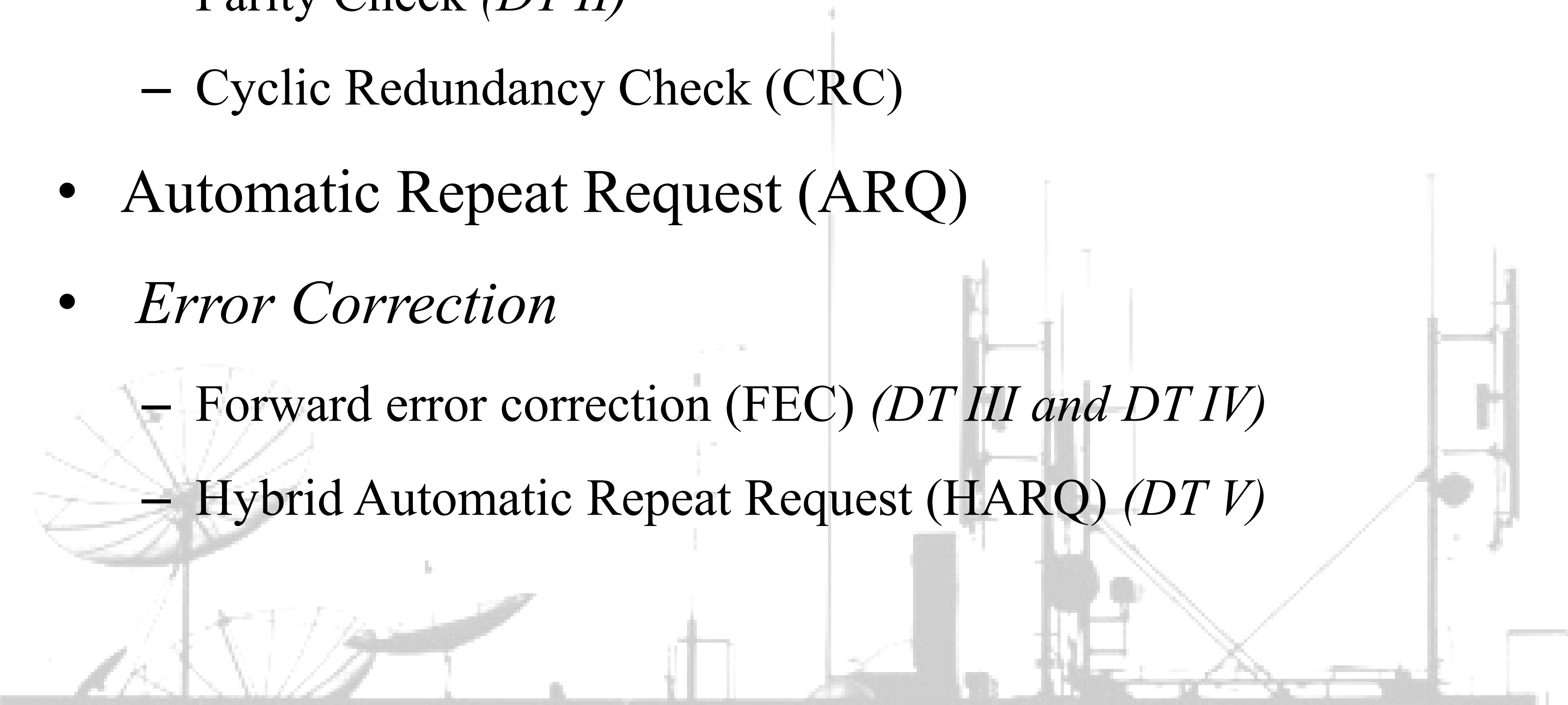
WHERE WE ARE IN THE COURSE

- Evolution of Wireless Communication, Transmission fundamentals, Analog and Digital Modulations (Lectures 2-4)
- The Wireless Channel (Lectures 5 and 6)
- Transmission Fundamentals (CTFT, DTFT) (Lecture 7)
- Orthogonal Frequency Division Multiplexing- OFDM (Lecture 8)
- Spread Spectrum (Lecture 10)
- Coding and Error Control (Lecture 11)
- Cellular Wireless Networks (Lecture 12)
- Short-Range Wireless Technologies & Applications of Wireless Technologies in IoT (Lecture 13)

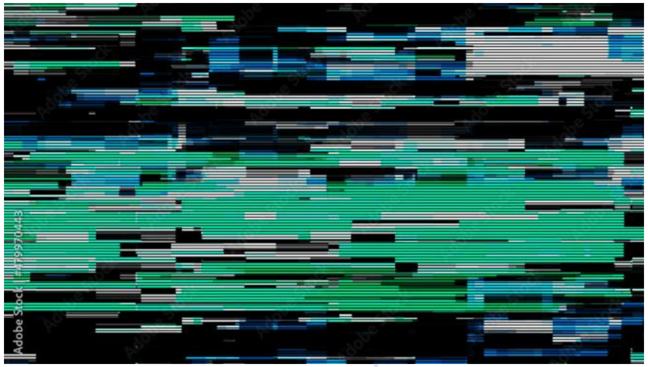


OUTLINE

- Introduction to “*Coding and Error Control*” *(DT I)*
- *Error Detection*
  - Parity Check *(DT II)*
  - Cyclic Redundancy Check (CRC)
- Automatic Repeat Request (ARQ)
- *Error Correction*
  - Forward error correction (FEC) *(DT III and DT IV)*
  - Hybrid Automatic Repeat Request (HARQ) *(DT V)*



CODING AND ERROR CONTROL – THE ART OF FIXING MISTAKES



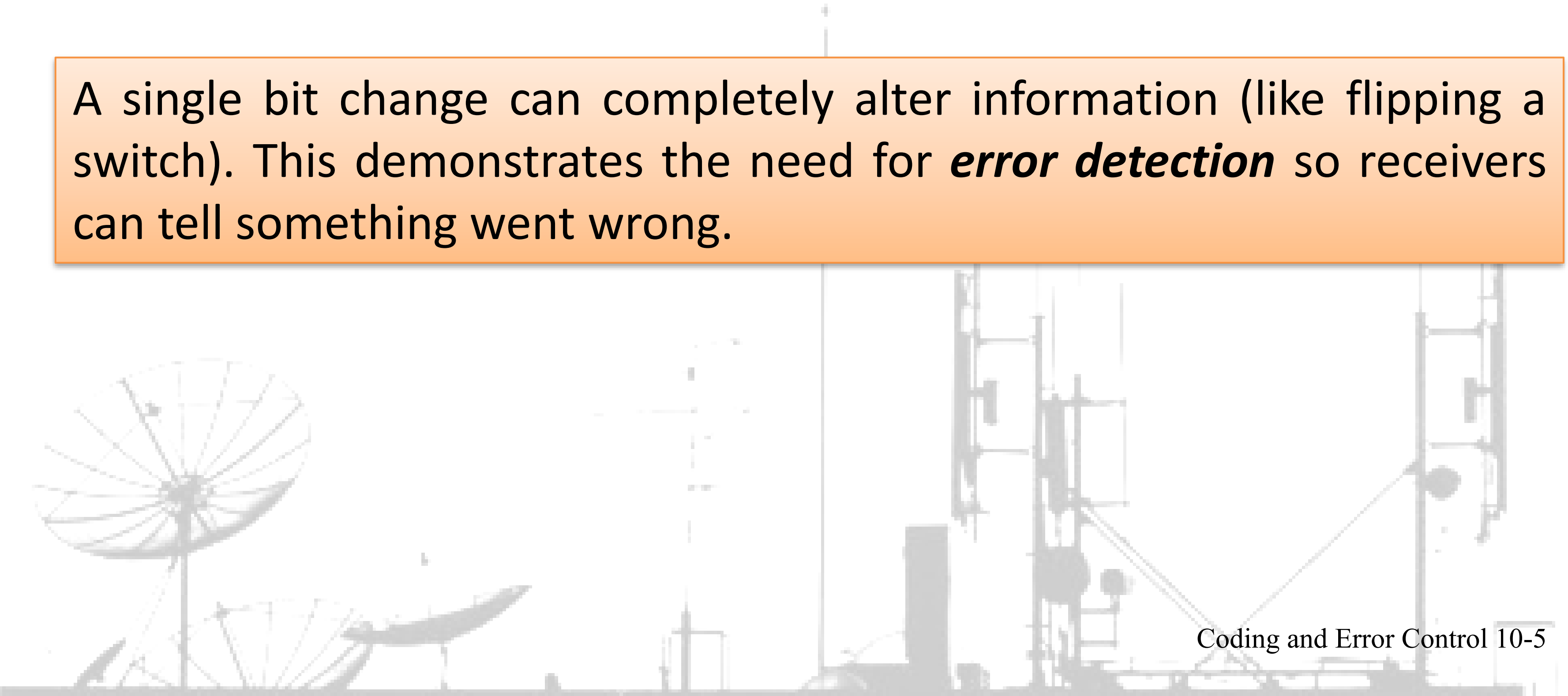
- Data may arrive incorrectly or not at all.
- Error control ensures data is either *detected and corrected* or *detected and retransmitted,* so the system remains reliable.

WHY DOES IT MATTER?

$$101\boxed{0}110 \rightarrow 1011110$$

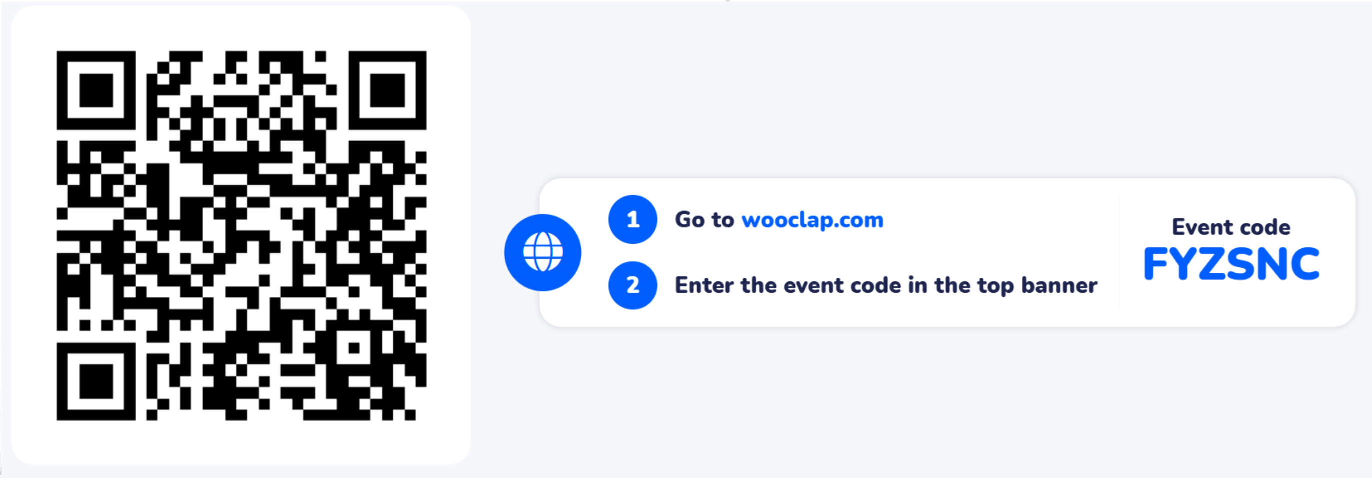
Can you spot the error? Why does it matter?

A single bit change can completely alter information (like flipping a switch). This demonstrates the need for ***error detection*** so receivers can tell something went wrong.



DISCUSSION TIME I: WHY ERRORS HAPPEN?

What could cause data errors in wireless communications?



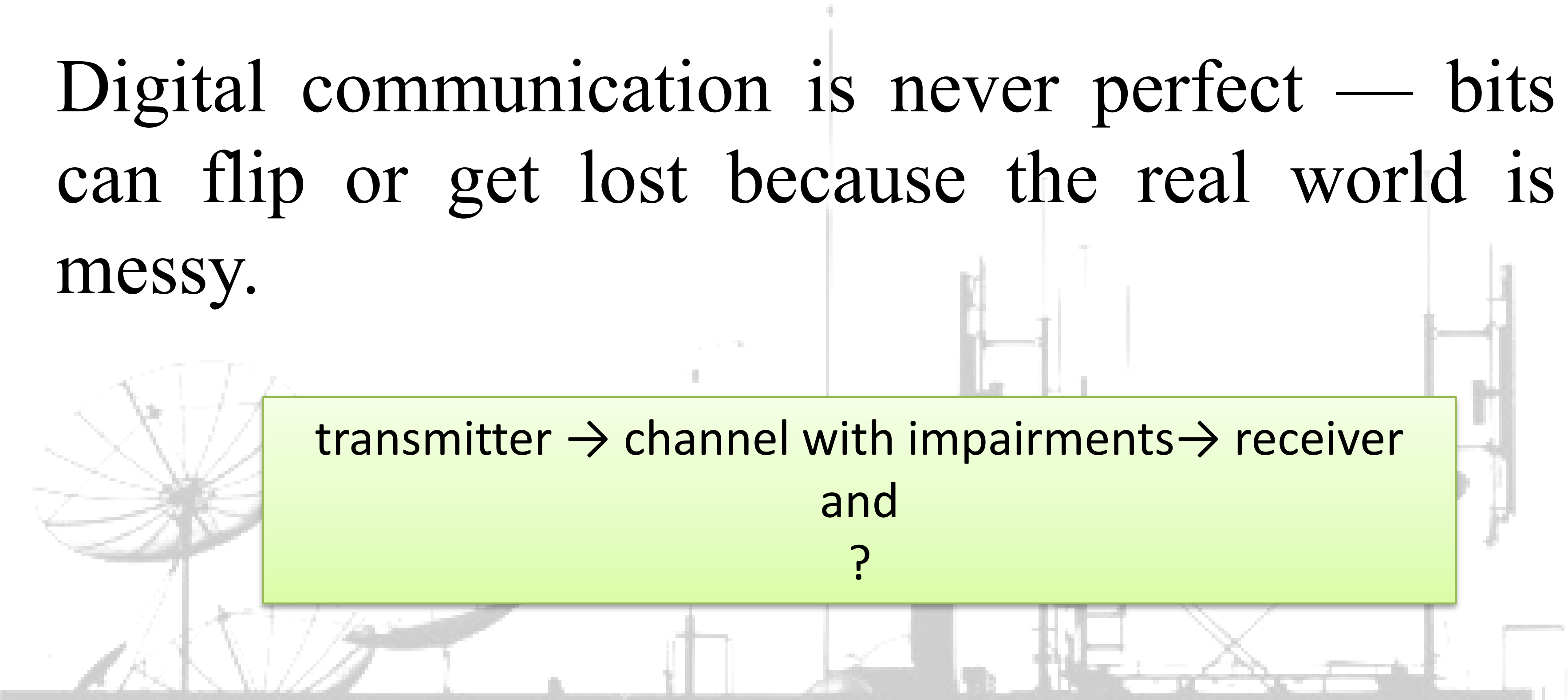
1. Go to wooclap.com
2. Enter the event code in the top banner

Event code
FYZSNC

WHY ERRORS HAPPEN?

What could cause data errors in wireless communications?

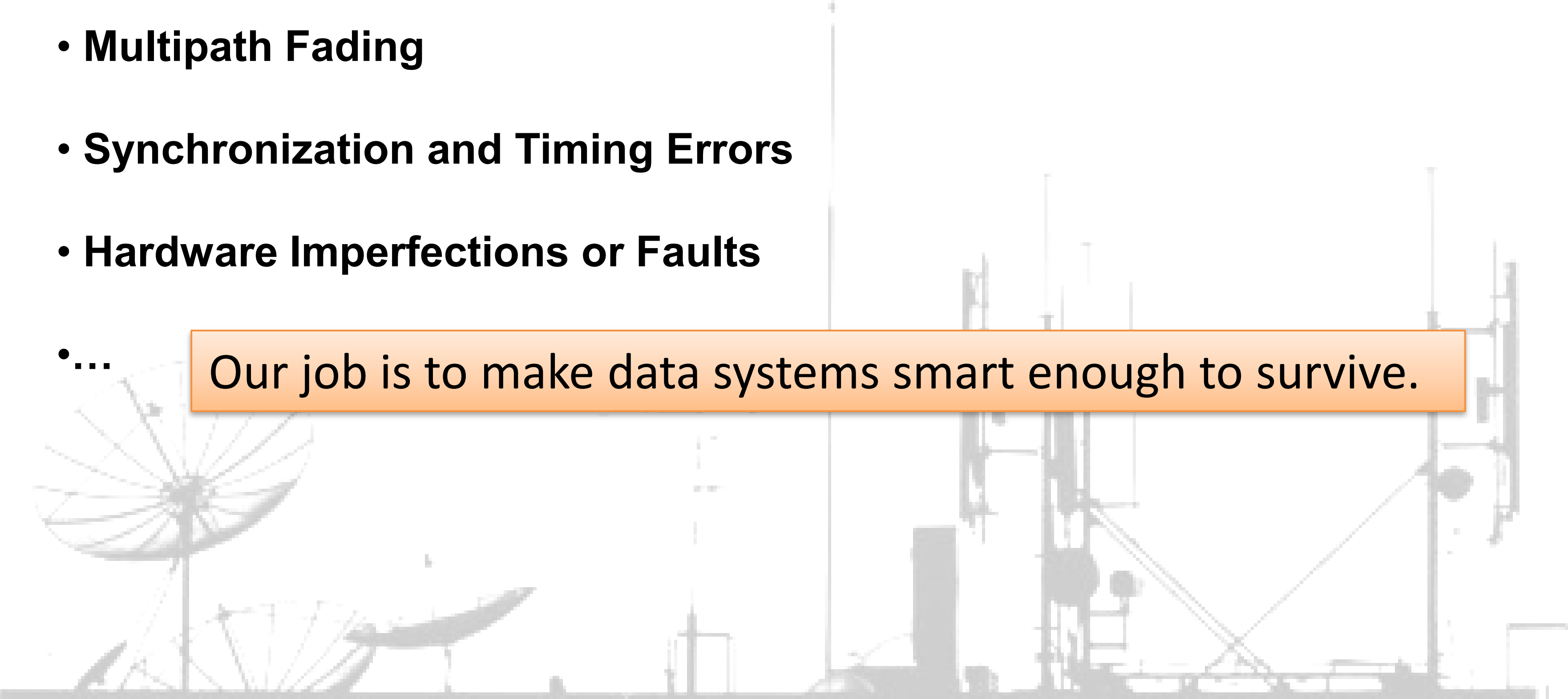
Digital communication is never perfect — bits can flip or get lost because the real world is messy.



transmitter → channel with impairments→ receiver
and
?

WHY ERRORS HAPPEN?

- Various types of noise (including thermal noise) and interference (including jamming)
- Signal Attenuation (Weakening Over Distance)
- Multipath Fading
- Synchronization and Timing Errors
- Hardware Imperfections or Faults
- …



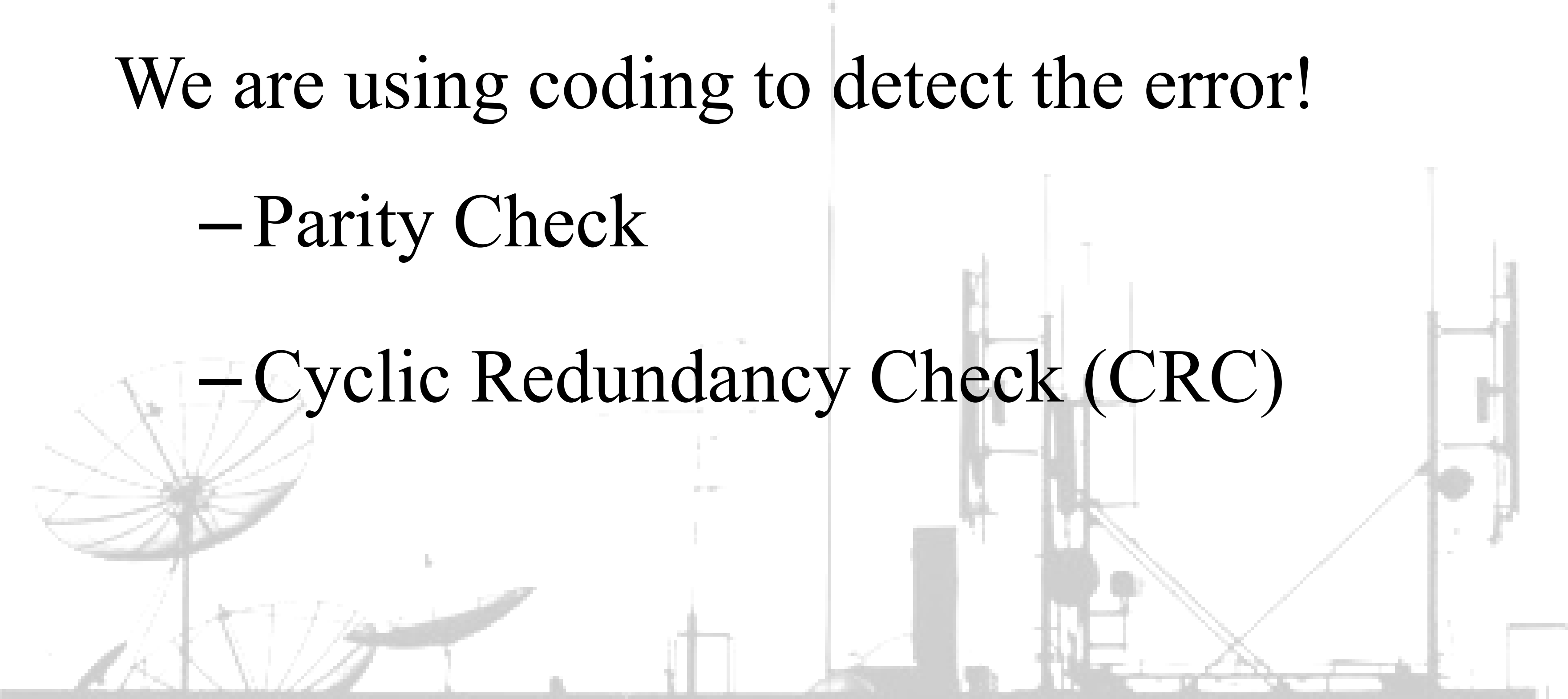
Our job is to make data systems smart enough to survive.

ERROR DETECTION?

A way to notice when something went wrong with the received data.

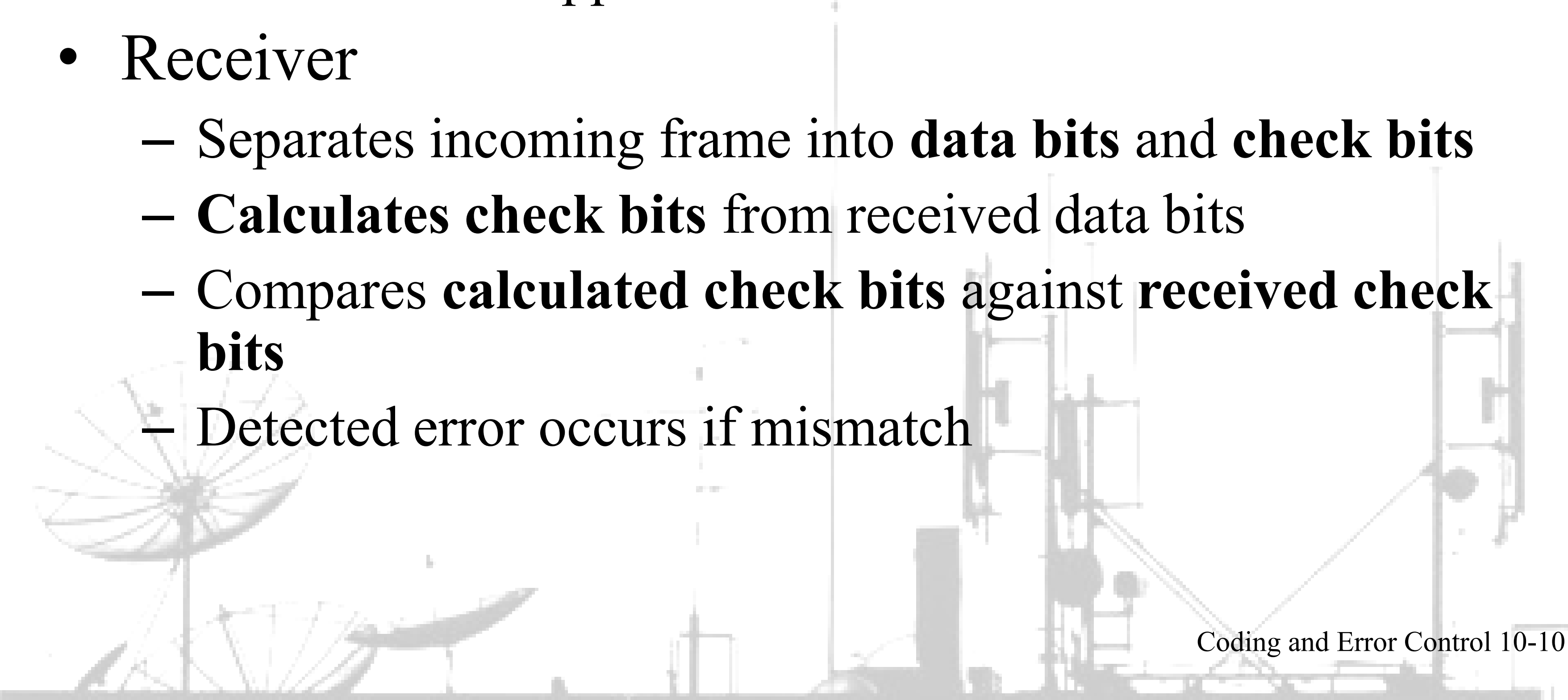
We are using coding to detect the error!

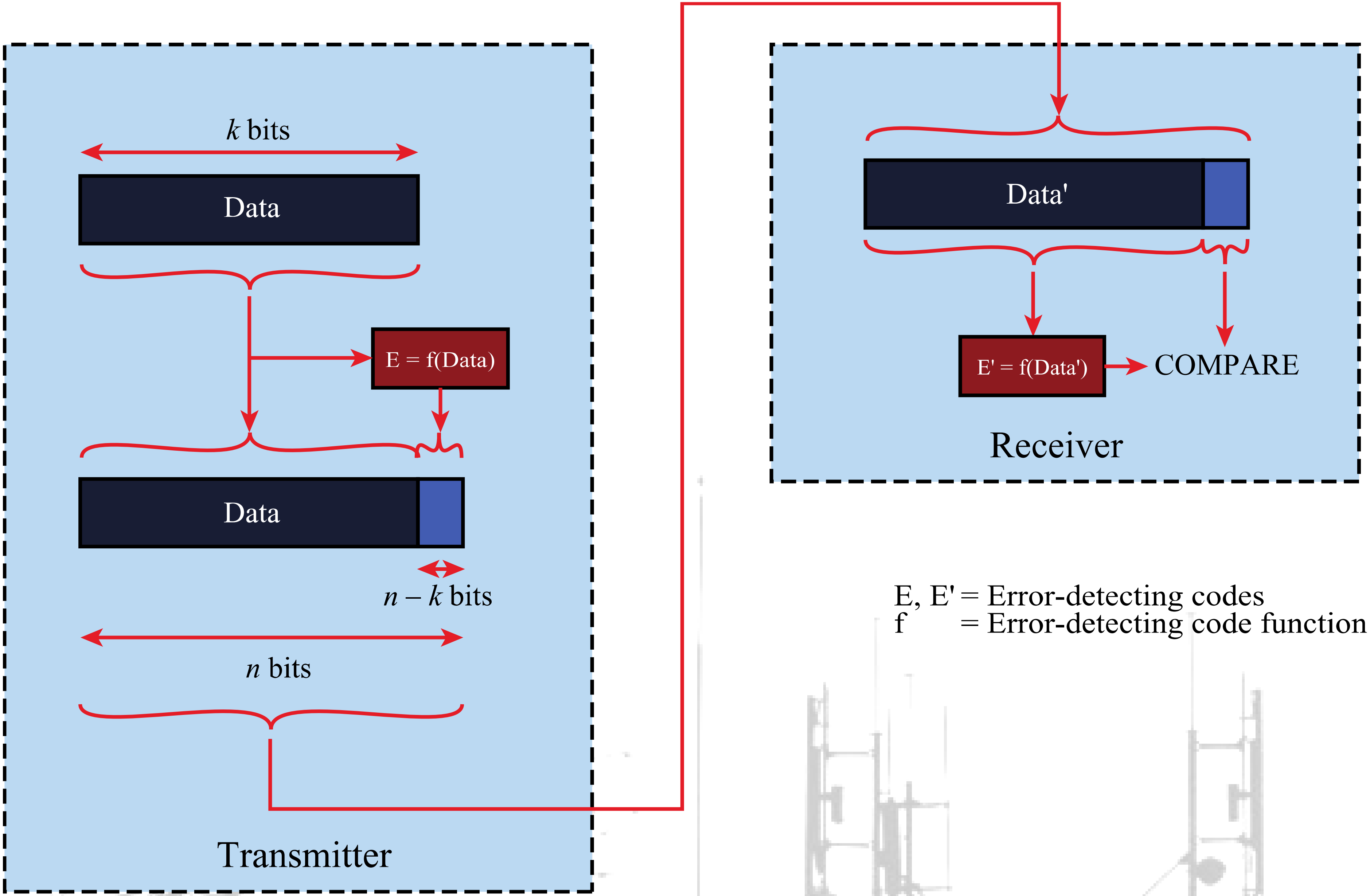
- Parity Check
- Cyclic Redundancy Check (CRC)



ERROR DETECTION PROCESS

- Transmitter
  - For a given frame, an error-detecting code (**check bits**) is calculated from data bits
  - Check bits are appended to data bits
- Receiver
  - Separates incoming frame into **data bits** and **check bits**
  - **Calculates check bits** from received data bits
  - Compares **calculated check bits** against **received check bits**
  - Detected error occurs if mismatch





E, E' = Error-detecting codes
f = Error-detecting code function

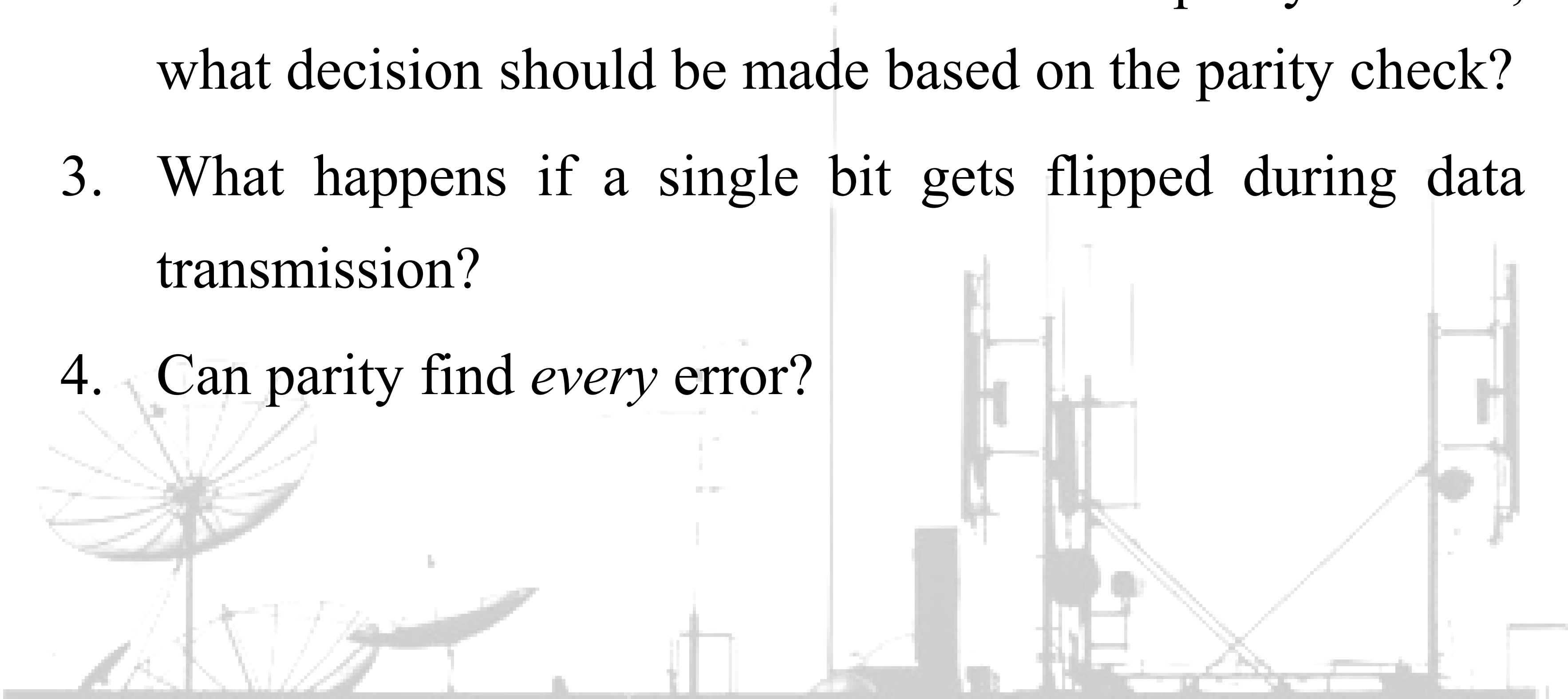
10.1 ERROR DETECTION PROCESS

PARITY CHECK

- Parity bit appended to a block of data
  - **Even parity**
    - Added bit ensures an even number of 1s
  - **Odd parity**
    - Added bit ensures an odd number of 1s
- Example, 7-bit character [1110001]
  - Even parity [1110001**0**]
  - Odd parity [1110001**1**]

DISCUSSION TIME II: PARITY DETECTIVES

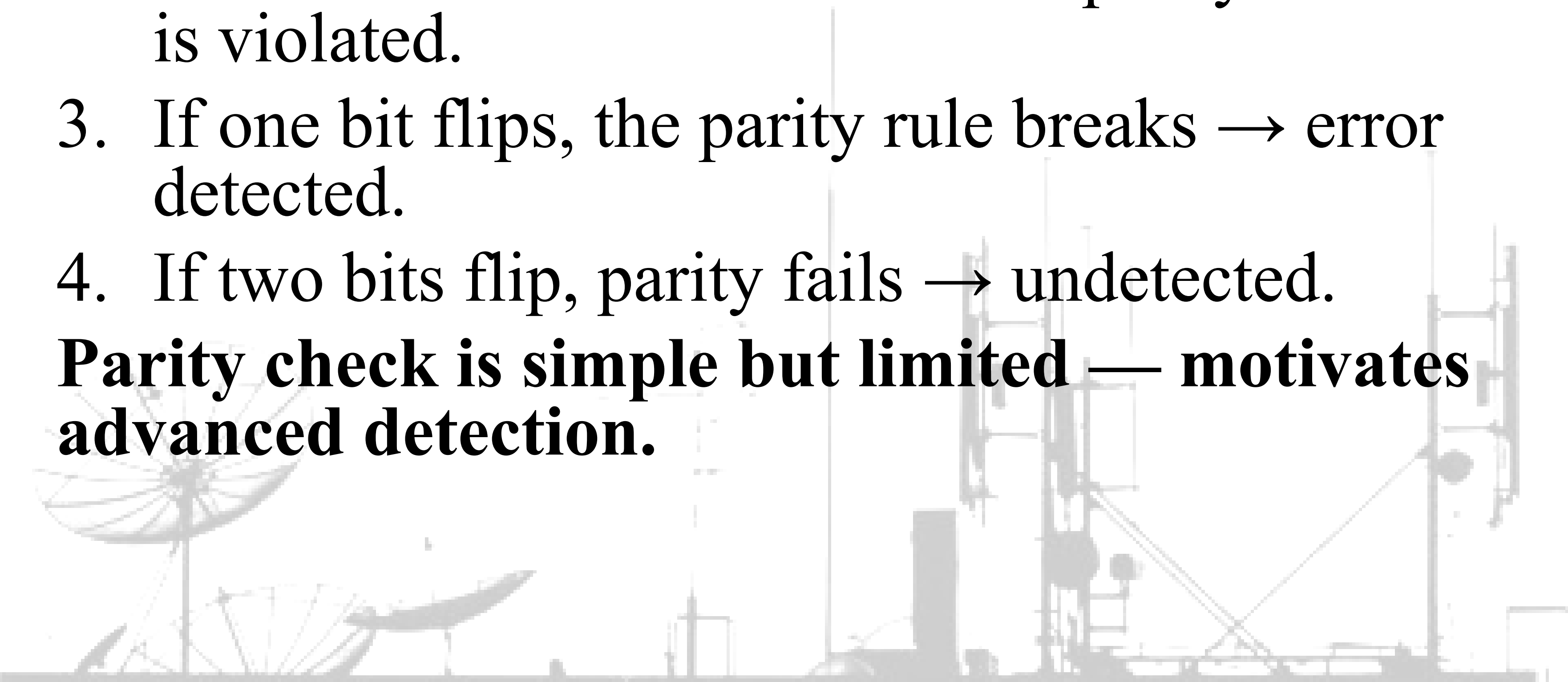
1. Try encoding 11001 with even and odd parity.
2. If the receiver receives 110011 and odd parity is used, what decision should be made based on the parity check?
3. What happens if a single bit gets flipped during data transmission?
4. Can parity find *every* error?



CLASS DISCUSSION

1.
    - Even parity: 11001**1**
    - Odd parity: 11001**0**
2. An error is detected because the parity condition is violated.
3. If one bit flips, the parity rule breaks → error detected.
4. If two bits flip, parity fails → undetected.

Parity check is simple but limited — motivates advanced detection.



CYCLIC REDUNDANCY CHECK (CRC)

- Transmitter
  - For a *k*-bit block, transmitter generates an (*n*-*k*)-bit **frame check sequence (FCS)**
  - Resulting frame of *n* bits is exactly divisible by **predetermined number**
- Receiver
  - Divides incoming frame by predetermined number
  - If no remainder, assumes no error

Data + **FCS** bits → divisible by a known number.

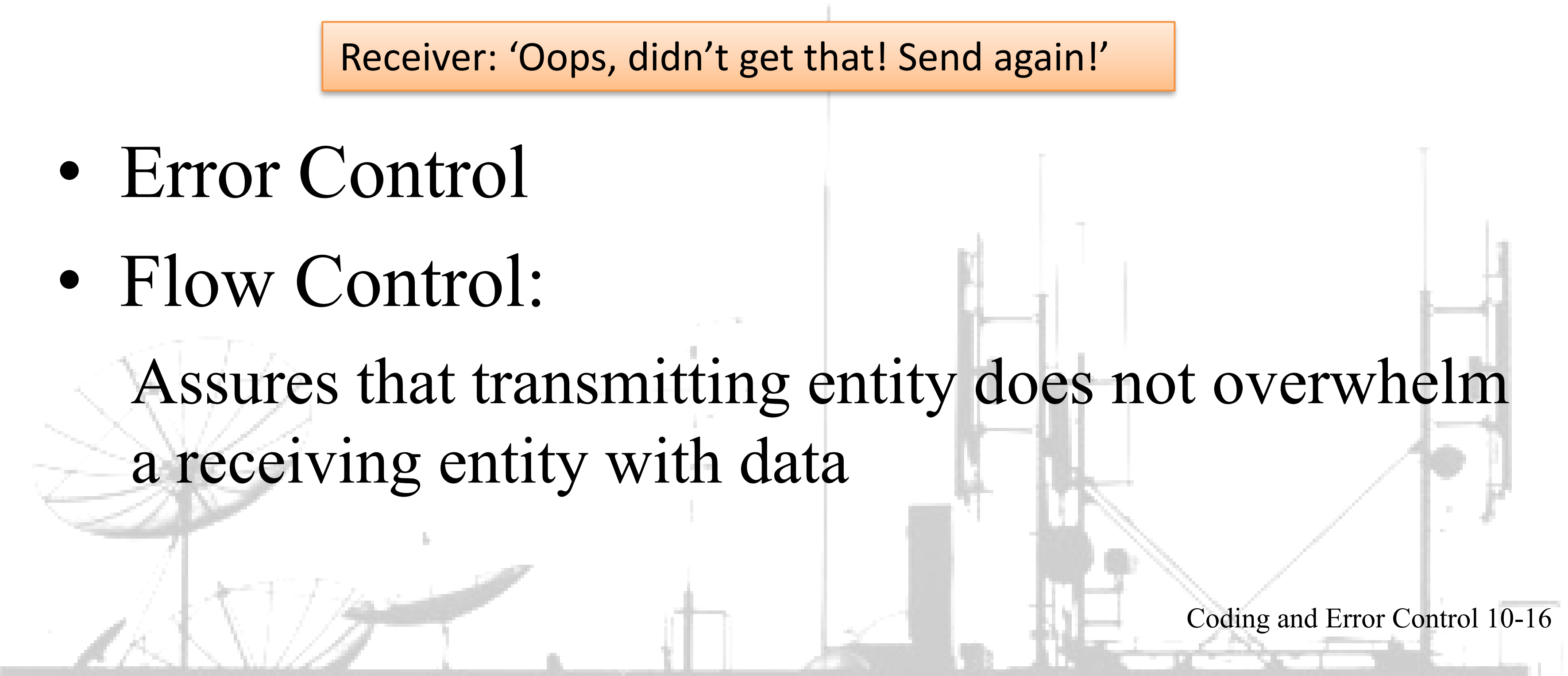
AUTOMATIC REPEAT REQUEST

- Error detection requires retransmission
- Automatic Repeat reQuest (ARQ) relies on use of an error detection code (such as CRC)

Receiver: ‘Oops, didn’t get that! Send again!’

- Error Control
- Flow Control:

  Assures that transmitting entity does not overwhelm a receiving entity with data



ARQ FOR WIRELESS COMMUNICATIONS

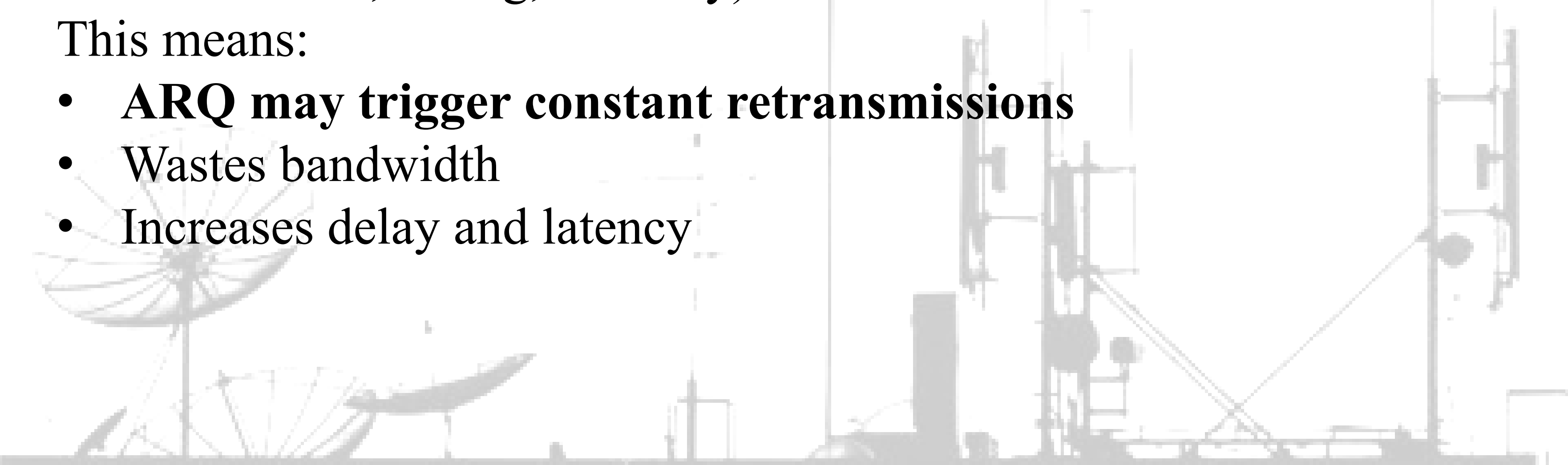
Is Error detection + ARQ good for:

- **cables?**
- **wireless?**

- In wired systems (like Ethernet or fiber), channel quality is high, so ARQ works very well — errors are infrequent and retransmitting occasionally is no problem.
- But in wireless systems, the environment causes many errors (due to interference, fading, mobility).

This means:

- **ARQ may trigger constant retransmissions**
- Wastes bandwidth
- Increases delay and latency



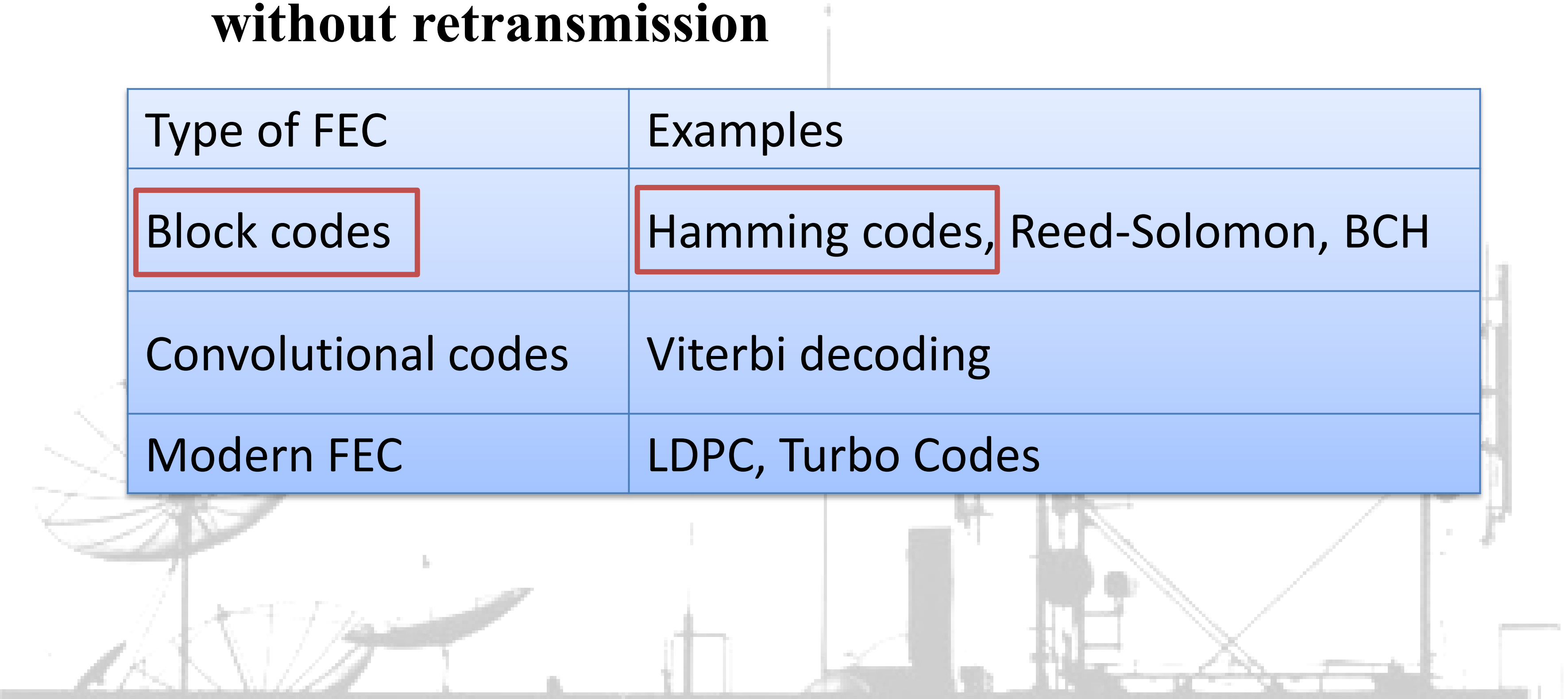
ERROR CORRECTION

Forward Error Correction

A technique where:

- Redundant bits are added at the sender
- Receiver can **detect and correct** some bit errors **without retransmission**

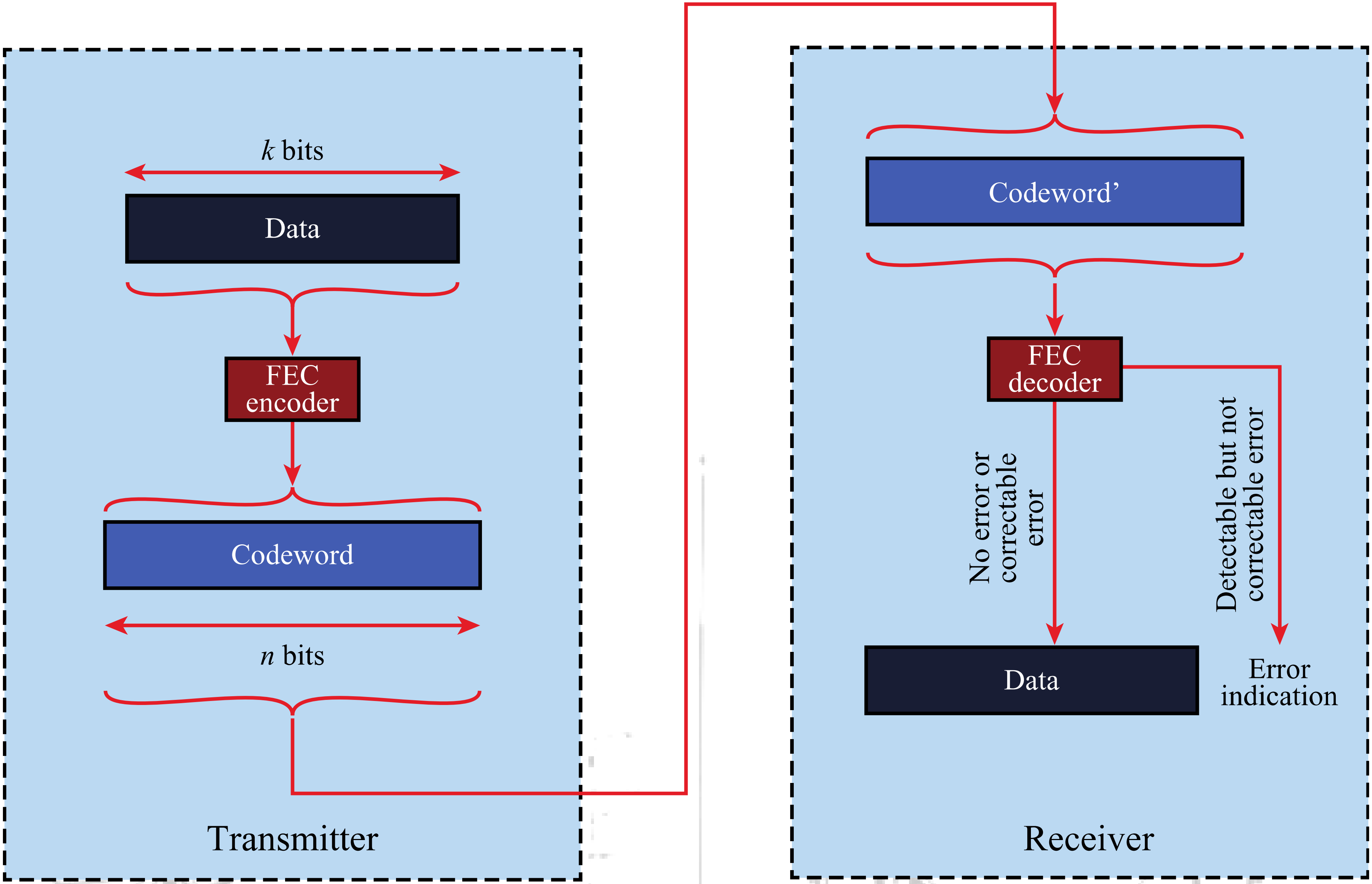
Type of FEC	Examples
Block codes	Hamming codes, Reed-Solomon, BCH
Convolutional codes	Viterbi decoding
Modern FEC	LDPC, Turbo Codes



BLOCK FORWARD ERROR CORRECTION

- Transmitter
  - **Forward error correction (FEC)** encoder maps each k-bit block into an n-bit block **codeword**
  - Codeword is transmitted

- Receiver
  - Incoming signal is demodulated
  - Block passed through an **FEC decoder**
  - **FEC decoder fixes certain errors without asking for retransmission.**



10.5 FORWARD ERROR CORRECTION PROCESS

HAMMING CODE

- Designed to correct single bit errors

- Family of (n, k) block error-correcting codes with parameters:
  - Block length: $n = 2^m - 1$
  - Number of data bits: $k = 2^m - m - 1$
  - Number of check bits: $n - k = m$
  - Minimum distance: $d_{\min} = 3$

Hamming can fix single-bit errors by comparing distance between codewords.

DECODING PROCESS

- Coding table

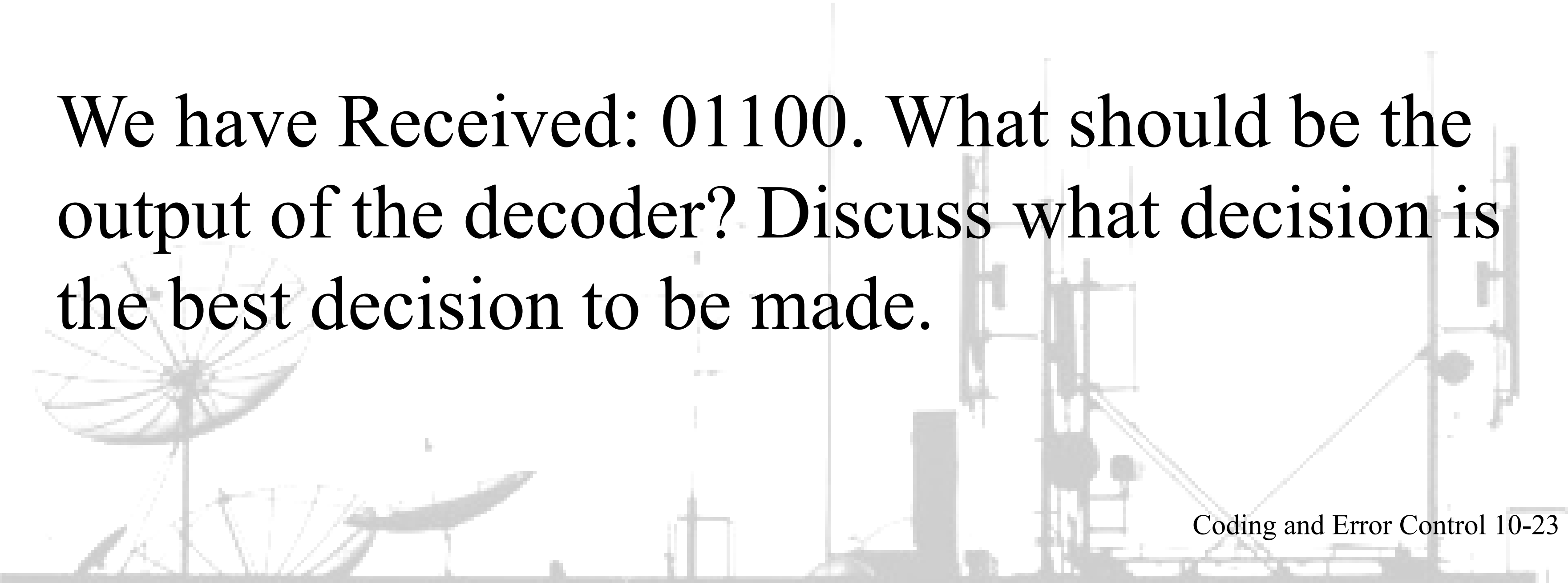
Data block	Codeword
00	00000
01	00111
10	11001
11	11110

- Received: 00100
  - Not valid, error is detected
  - Correction?
    - One bit away from 00000
    - Two bits away from 00111
    - Three bits away from 11110
    - Four bits away from 11001
  - Most likely 00000 was sent, assume data was 00
    - But others could have been sent, albeit much less likely

DISCUSSION TIME III: DECODING PROCESS IN BLOCK ERROR CORRECTION

Data block	Codeword
00	00000
01	00111
10	11001
11	11110

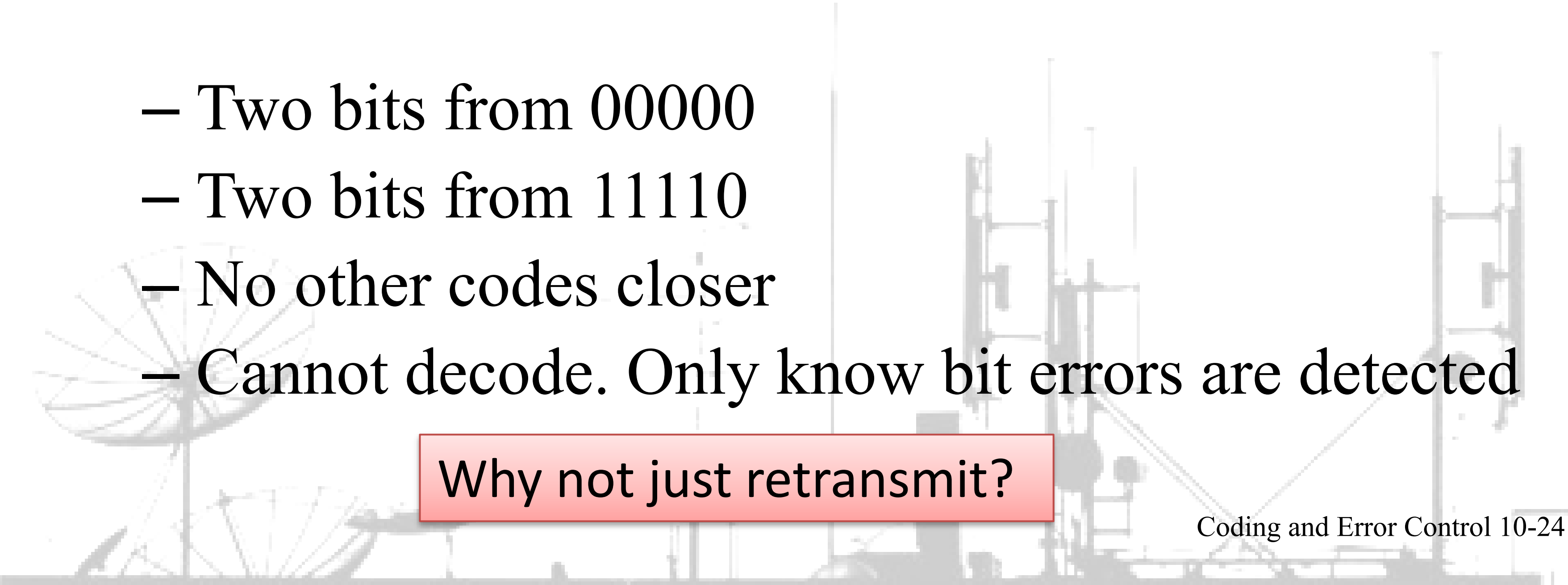
We have Received: 01100. What should be the output of the decoder? Discuss what decision is the best decision to be made.



DISCUSSION TIME III: DECODING PROCESS IN BLOCK ERROR CORRECTION

Data block	Codeword
00	00000
01	00111
10	11001
11	11110

- – Two bits from 00000
- – Two bits from 11110
- – No other codes closer
- – Cannot decode. Only know bit errors are detected



Why not just retransmit?

DISCUSSION TIME IV

What are the advantages and disadvantages of the FEC and ARQ. Is any of them adequate in practical situations?

DISCUSSION TIME IV

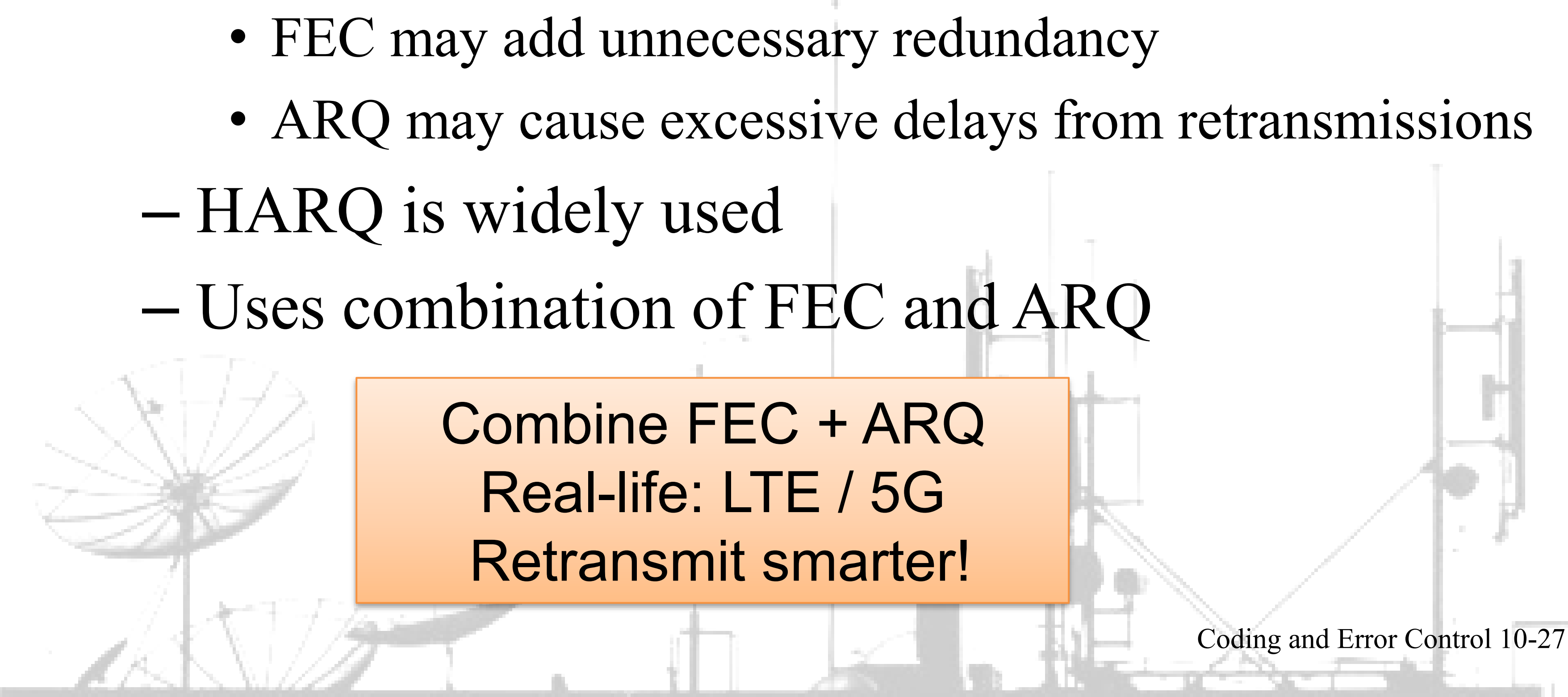
What are the advantages and disadvantages of the FEC and ARQ. Is any of them adequate in practical situations?

Neither FEC or ARQ is adequate in practical situations

- FEC may add unnecessary redundancy, but it is useful when real-time communication is needed.
- ARQ may cause excessive delays from retransmissions, but it is useful in low error rate environments.

HYBRID ARQ

- Hybrid Automatic Repeat Request (HARQ)
  - Neither FEC or ARQ is adequate in practical situations
    - FEC may add unnecessary redundancy
    - ARQ may cause excessive delays from retransmissions
  - HARQ is widely used
  - Uses combination of FEC and ARQ



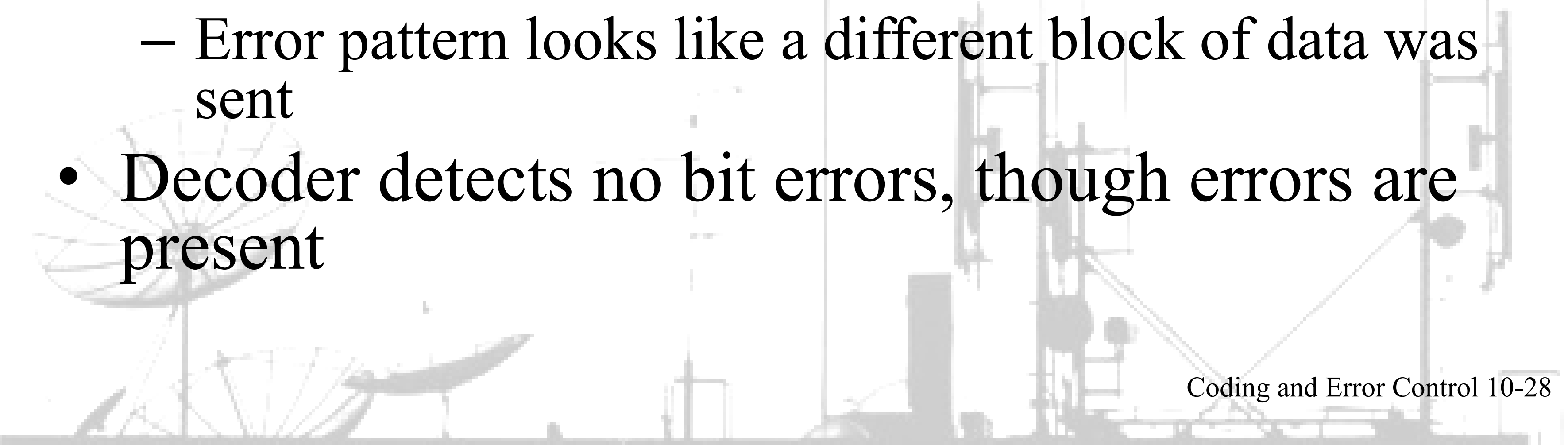
Combine FEC + ARQ
Real-life: LTE / 5G
Retransmit smarter!

FEC DECODER OUTCOMES

- No errors present
  - Codeword produced by decoder matches original codeword
- Decoder detects and corrects bit errors
- Decoder detects but cannot correct bit errors; reports uncorrectable error

ARQ

- Decoder incorrectly corrects bit errors
  - Error pattern looks like a different block of data was sent
- Decoder detects no bit errors, though errors are present



DISCUSSION V: REFLECTION DISCUSSION

If you were designing a communication system, which error control technique would you choose — and why?

DISCUSSION V: REFLECTION DISCUSSION

If you were designing a communication system, which error control technique would you choose — and why?

- ARQ based on Parity Check: simplest, low overhead (for low-error environments)
- ARQ based on CRC: strong detection, light computation can cause delays
- FEC: best when retransmission is slow (we have high error rates) and we need real-time communication
- HARQ: best modern approach — balances speed & reliability

SUMMARY: KEY TAKEAWAYS

Technique	Goal	Example
Parity	Detect simple errors	Early computers
CRC	Detect multiple errors	Ethernet, ZIP
FEC	Correct errors	Satellite, CDs
HARQ	Mix detect + correct	4G/5G