

“I called my brothel Friday night.”

“I called my brothel Friday night.”

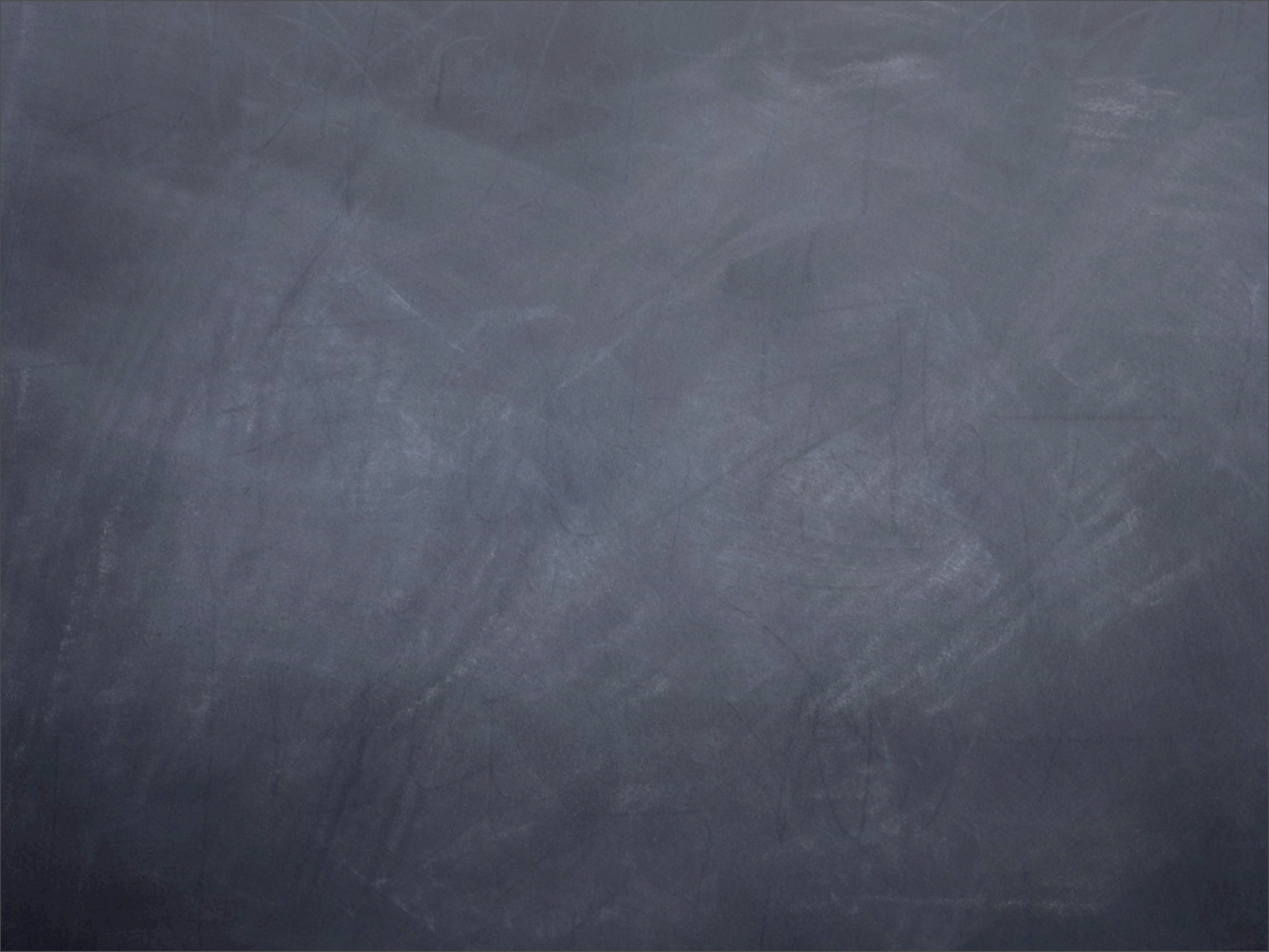
“I wanted to see how he was doing
after his surgery.”

“I called my brothel Friday night.”

ASCII codes for r and l:

l = 01101100

r = 01101010



Transmitting Large Messages

0 1 1 0 1 0 0 1 0 1 0 1 0 0 0 1 0 1 0 0 0 0 1 1

becomes

0 1 1 0 1 0 0 1

0 1 0 1 0 0 0 1

0 1 0 0 0 0 1 1

becomes

0 1 1 0 1 0 0 1 0

0 1 0 1 0 0 0 1 1

0 1 0 0 0 0 1 1 1

becomes

0 1 1 0 1 0 0 1 0 0 1 0 1 0 0 0 1 1 0 1 0 0 0 0 1 1 1

Transmitting Large Messages

0 1 1 0 1 0 0 1 0 1 0 1 0 0 0 1 0 1 0 0 0 0 1 1

0 1 1 0 1 0 0 1
0 1 0 1 0 0 0 1
0 1 0 0 0 0 1 1

0 1 1 0 1 0 0 1 0
0 1 0 1 0 0 0 1 1
0 1 0 0 0 0 1 1 1

0 1 1 0 1 0 0 1 0 0 1 0 1 0 0 0 1 1 0 1 0 0 0 0 1 1 1

Burst Errors

0 1 1 0 1 0 0 1 0 0 1 0 1 0 0 0 1 1 0 1 0 0 0 0 1 1 1

becomes

0 1 1 ? 1 1 1

Vertical Parity

0 1 1 0 1 0 0 1 0 1 0 1 0 0 0 1 0 1 0 0 0 0 1 1

0 1 1 0 1 0 0 1
0 1 0 1 0 0 0 1
0 1 0 0 0 0 1 1

0 1 1 0 1 0 0 1
0 1 0 1 0 0 0 1
0 1 0 0 0 0 1 1
0 1 1 1 1 0 1 1

0 1 1 0 1 0 0 1 0 1 0 1 0 0 0 1 0 1 0 0 0 0 1 1 0 1 1 1 1 0 1 1

Vertical Parity & Burst Errors

Short burst?
(\leq width)

0	1	1	0	1	0	0	1
0	1	0	?	?	?	?	?
?	?	?	0	0	0	1	1
0	1	1	1	1	0	1	1

$P(\text{undetected error})$

=

?

Long burst?
($\geq 2 * \text{width}$)

0	1	1	0	1	0	?	?
?	?	?	?	?	?	?	?
?	?	?	?	?	?	?	1
0	1	1	1	1	0	1	1

$P(\text{undetected error})$

\cong

?

Vertical Parity & Burst Errors

Short burst?
($\leq \text{width}$)

0	1	1	0	1	0	0	1
0	1	0	?	?	?	?	?
?	?	?	0	0	0	1	1
0	1	1	1	1	0	1	1

$P(\text{undetected error})$

=

0

Long burst?
($\geq 2 * \text{width}$)

0	1	1	0	1	0	?	?
?	?	?	?	?	?	?	?
?	?	?	?	?	?	?	1
0	1	1	1	1	0	1	1

$P(\text{undetected error})$

\cong

?

Vertical Parity & Burst Errors

Short burst?
($\leq \text{width}$)

0	1	1	0	1	0	0	1
0	1	0	?	?	?	?	?
?	?	?	0	0	0	1	1
0	1	1	1	1	0	1	1

$P(\text{undetected error})$

=

0

Long burst?
($\geq 2 * \text{width}$)

0	1	1	0	1	0	?	?
?	?	?	?	?	?	?	?
?	?	?	?	?	?	?	1
0	1	1	1	1	0	1	1

$P(\text{undetected error})$

\cong

$1/2^{\text{width}}$

Vertical Parity & Burst Errors

Medium burst?
(width < burst < 2*width)

0	1	1	0	1	0	0	1
0	1	0	?	?	?	?	?
?	?	?	?	0	0	1	1
0	1	1	1	1	0	1	1

$P(\text{undetected error})$

=

?

Vertical Parity & Burst Errors

Medium burst?
(width < burst < 2*width)

0	1	1	0	1	0	0	1
0	1	0	?	?	?	?	?
?	?	?	?	?	?	1	1
0	1	1	1	1	0	1	1

$P(\text{undetected error})$

=

?

Vertical Parity & Burst Errors

Medium burst?

(width < burst < 2*width)

0	1	1	0	1	0	0	1
0	1	0	?	?	?	?	?
?	?	?	?	?	?	1	1
0	1	1	1	1	0	1	1

$P(\text{undetected error})$

=

$P(\text{no error detected}) - P(\text{no error})$

Vertical Parity & Burst Errors

Medium burst?

(width < burst < 2*width)

0	1	1	0	1	0	0	1
0	1	0	?	?	?	?	?
?	?	?	?	?	?	1	1
0	1	1	1	1	0	1	1

$P(\text{undetected error})$

=

$P(\text{no error detected}) - P(\text{corruption} == \text{original seq.})$

=

$1/2^{\text{width}} - 1/2^{\text{burst}}$

2 Dimensional Parity

0 1 1 0 1 0 0 1 0 1 0 1 0 0 0 1 0 1 0 0 0 0 1 1

0 1 1 0 1 0 0 1
0 1 0 1 0 0 0 1
0 1 0 0 0 0 1 1

0 1 1 0 1 0 0 1 0
0 1 0 1 0 0 0 1 1
0 1 0 0 0 0 1 1 1
0 1 1 1 1 0 1 1 0

0 1 1 0 1 0 0 1 0 0 1 0 1 0 0 0 1 1 0 1 0 0 0 0 1 1 1 0 1 1 1 1 0 1 1 0

Error Correction

0	1	1	0	1	0	0	1	0
0	1	0	1	0	0	0	1	1
0	1	0	0	0	0	1	1	1
0	1	1	1	1	0	1	1	0

Error Correction

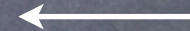
0	1	1	0	1	0	0	1	0
0	1	0	X	0	0	0	1	1
0	1	0	0	0	0	1	1	1
0	1	1	1	1	0	1	1	0

Error Correction

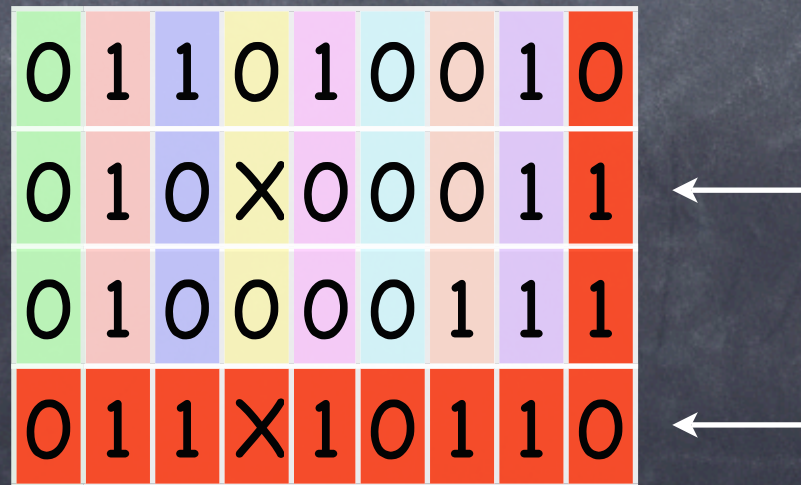
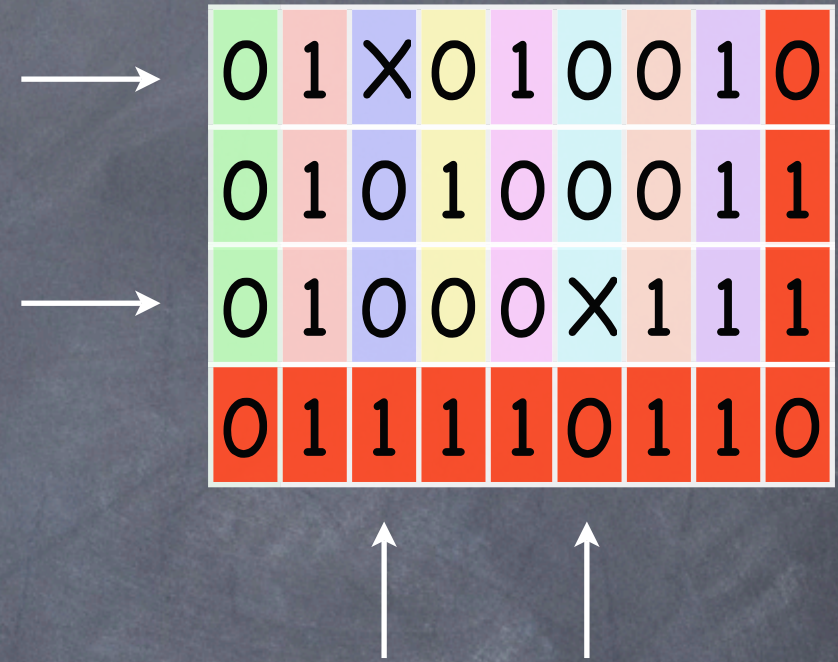
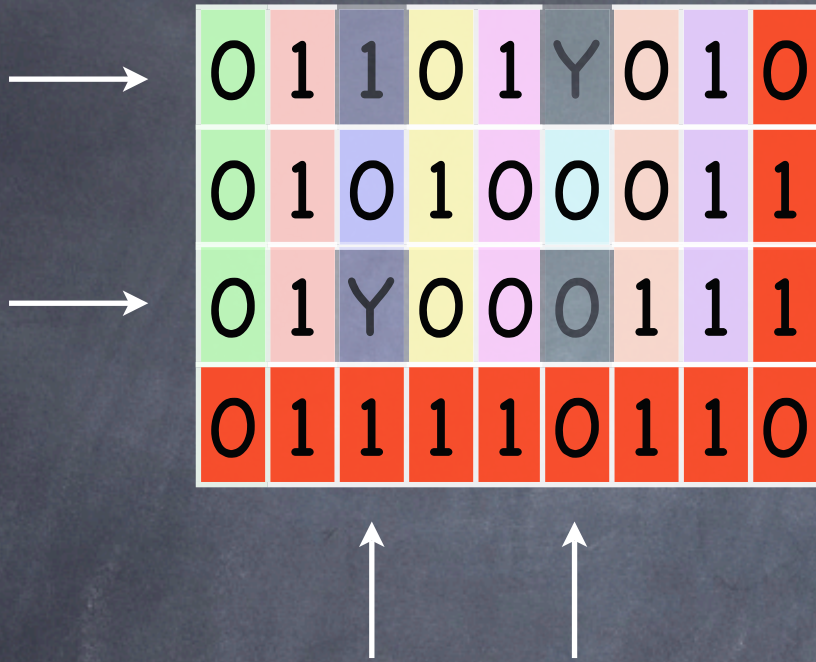
0	1	1	0	1	0	0	1	0
0	1	0	0	0	0	0	1	1
0	1	0	0	0	0	1	1	1
0	1	1	1	1	0	1	1	0

Error Correction

0	1	1	0	1	0	0	1	0
0	1	0	X	0	0	0	1	1
0	1	0	0	0	0	1	1	1
0	1	1	1	1	0	1	1	0



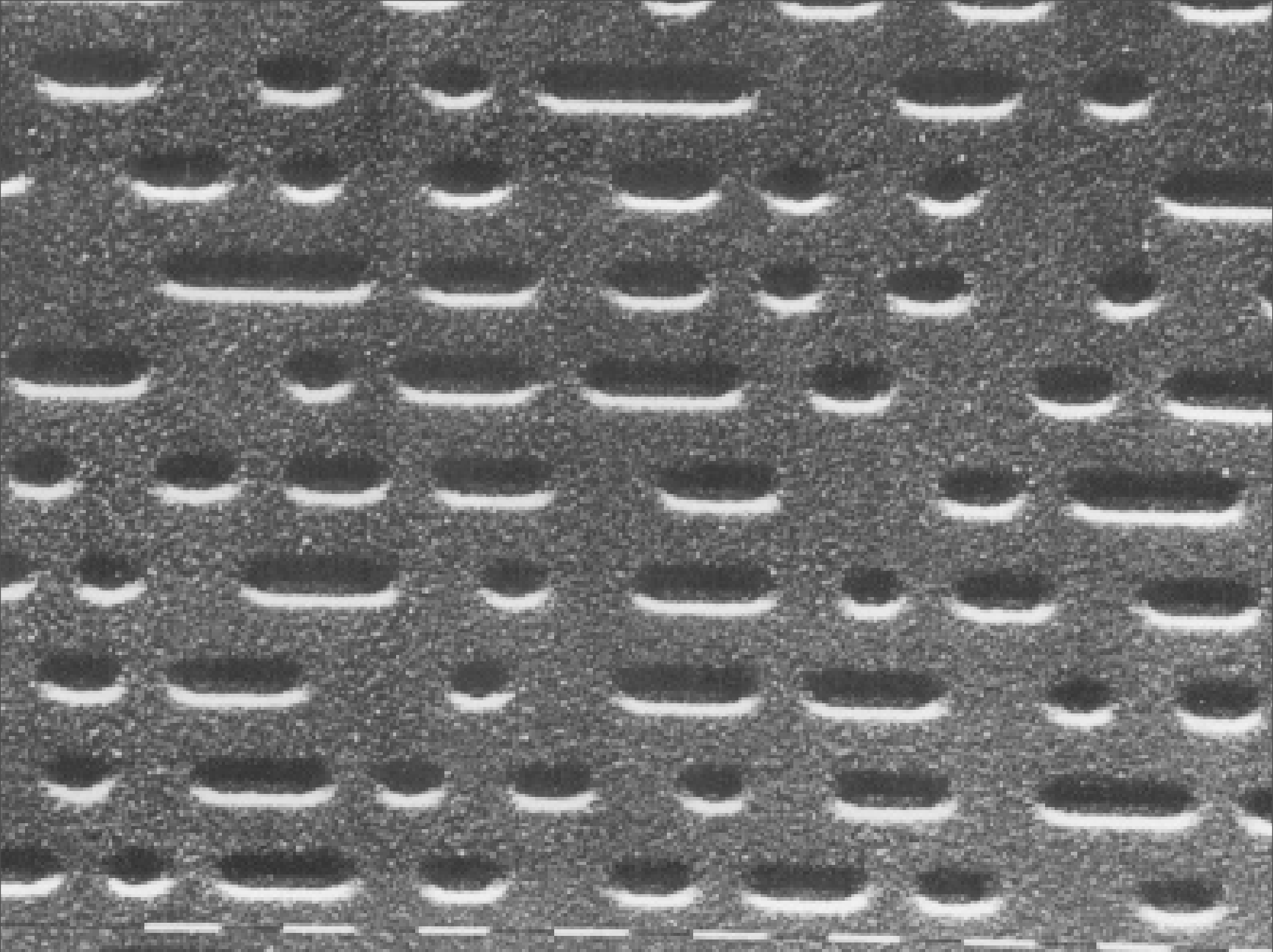
Error Detection



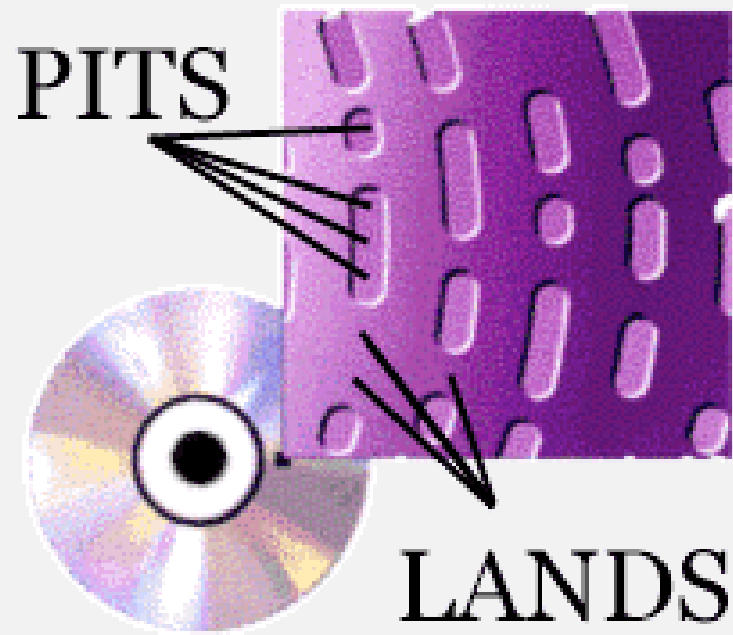
Whoops!

A 4x9 grid of numbers and symbols. The cells are colored as follows: Row 1: (1,1) green, (1,2) light red, (1,3) light blue, (1,4) yellow, (1,5) light pink, (1,6) light cyan, (1,7) light orange, (1,8) light purple, (1,9) red. Row 2: (2,1) green, (2,2) light red, (2,3) light blue, (2,4) yellow, (2,5) light pink, (2,6) light cyan, (2,7) light orange, (2,8) light purple, (2,9) red. Row 3: (3,1) green, (3,2) light red, (3,3) light blue, (3,4) yellow, (3,5) light pink, (3,6) light cyan, (3,7) light orange, (3,8) light purple, (3,9) red. Row 4: (4,1) red, (4,2) red, (4,3) red, (4,4) red, (4,5) red, (4,6) red, (4,7) red, (4,8) red, (4,9) red. A white arrow points to the cell at (3,1) containing '0'. Another white arrow points to the cell at (4,3) containing '1'.

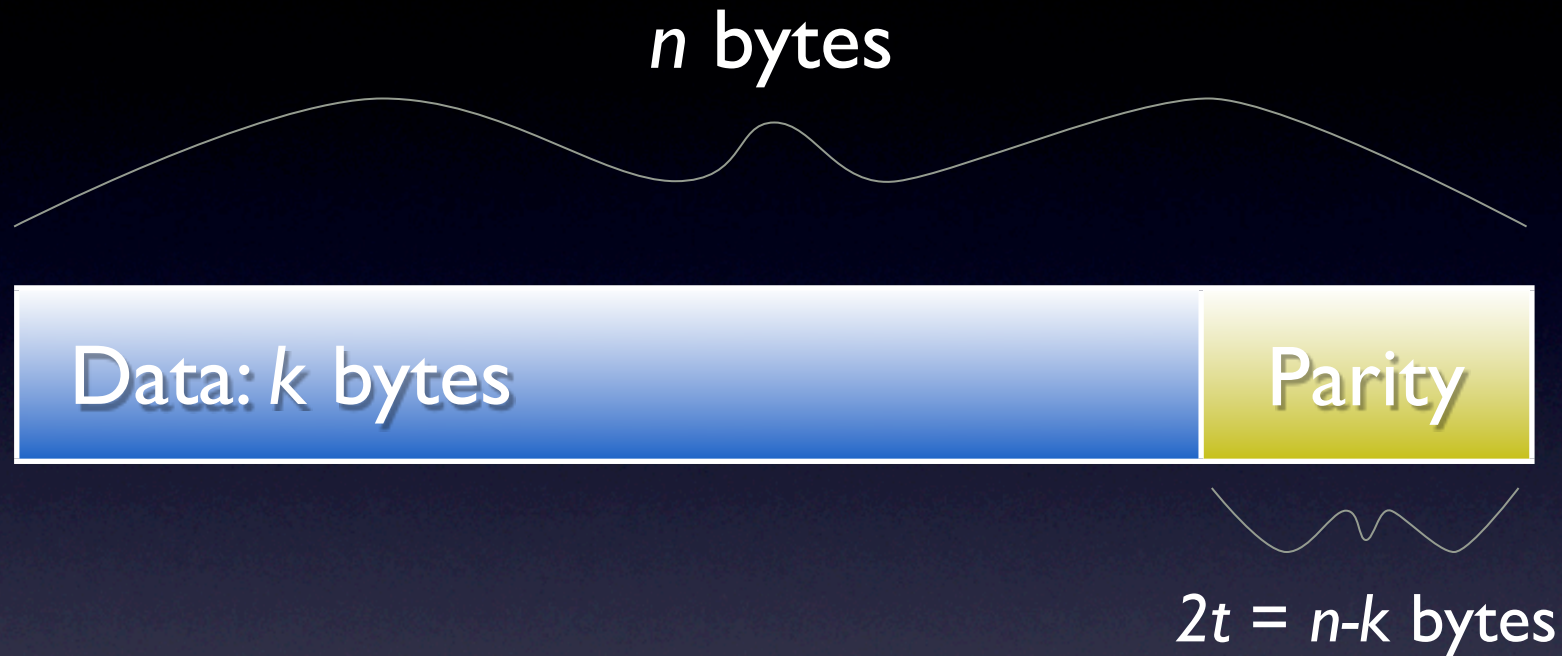
0	1	X	0	1	X	0	1	0
0	1	0	1	0	0	0	1	1
0	1	0	0	0	X	1	1	1
0	1	1	1	1	0	1	1	0



CD Encoding



Reed-Solomon Codes



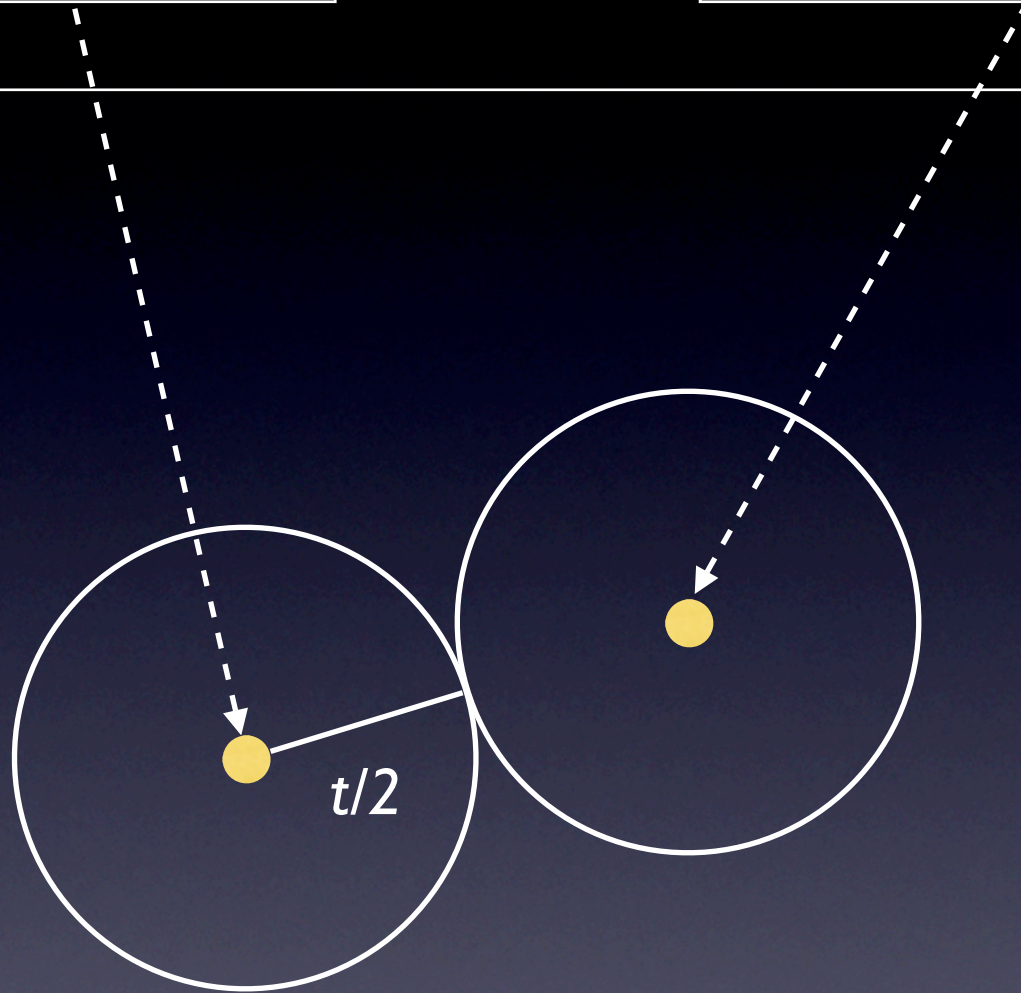
$2t = n - k$ bytes
correct errors in
up to t bytes

Example

- $n: 255$ $k: 223$ $t: 16$
- $16 \leq \# \text{ bits corrected} \leq 16 \times 8 = 128$

10101001110111

00111011100110



Space o' Codewords



Checksums









IP Datagram Format

4	4	8	16
IP version	Hdr len	Service class	Packet Length
Packet Number			Fragment Number
TTL	Protocol	Error Check	
From Addr			
To Addr			
DATA (up to 65516 bytes)			

TCP Segment Format

16		16	
Source Port		Destination Port	
Sequence Number			
Acknowledgment Number			
Hdr Len		Flags	Receiver window
Error Check		Urgent Pointer	
DATA			

Decimal Sum

$$\begin{array}{r} 22319423 \\ 1407438 \\ 39956194 \\ + 61708302 \\ \hline \end{array}$$

Decimal Sum

	2	2	3	1	9	4	2	3
	0	1	4	0	7	4	3	8
	3	9	9	5	6	1	9	4
+	6	1	7	0	8	3	0	2

Decimal Sum

$$\begin{array}{r} 1 \\ 22319423 \\ 01407438 \\ 39956194 \\ + 61708302 \\ \hline 7 \end{array}$$

Decimal Sum

$$\begin{array}{r} 11 \\ 22319423 \\ 01407438 \\ 39956194 \\ + 61708302 \\ \hline 57 \end{array}$$

Decimal Sum

2 0 3 1 1 1

2	2	3	1	9	4	2	3	
0	1	4	0	7	4	3	8	
3	9	9	5	6	1	9	4	
+	6	1	7	0	8	3	0	2

3 9 1 3 5 7

Decimal Sum

$$\begin{array}{r} 1203111 \\ 22319423 \\ 01407438 \\ 39956194 \\ + 61708302 \\ \hline 5391357 \end{array}$$

Decimal Sum

$$\begin{array}{r} 11203111 \\ \begin{array}{|c|c|c|c|c|c|c|c|} \hline 2 & 2 & 3 & 1 & 9 & 4 & 2 & 3 \\ \hline 0 & 1 & 4 & 0 & 7 & 4 & 3 & 8 \\ \hline 3 & 9 & 9 & 5 & 6 & 1 & 9 & 4 \\ \hline 6 & 1 & 7 & 0 & 8 & 3 & 0 & 2 \\ \hline \end{array} \\ + \\ \begin{array}{|c|c|c|c|c|c|c|c|} \hline 2 & 5 & 3 & 9 & 1 & 3 & 5 & 7 \\ \hline \end{array} \end{array}$$

Decimal Sum

$$\begin{array}{r} 1\ 1\ 2\ 0\ 3\ 1\ 1\ 1 \\ 2\ 2\ 3\ 1\ 9\ 4\ 2\ 3 \\ 0\ 1\ 4\ 0\ 7\ 4\ 3\ 8 \\ 3\ 9\ 9\ 5\ 6\ 1\ 9\ 4 \\ +\ 6\ 1\ 7\ 0\ 8\ 3\ 0\ 2 \\ \hline 1\ 2\ 5\ 3\ 9\ 1\ 3\ 5\ 7 \end{array}$$

Binary Sum

0	1	1	0	1	0	0	1	0	1	1	1	0	0	0	0
0	1	0	1	0	0	1	0	0	1	1	0	1	1	0	1
0	1	1	0	1	0	0	1	0	0	0	1	1	0	0	1
0	1	1	0	1	0	0	0	0	1	0	0	0	1	0	1

Binary Sum

																1
0	1	1	0	1	0	0	1	0	1	1	1	0	0	0	0	
0	1	0	1	0	0	1	0	0	1	1	0	1	1	0	1	
0	1	1	0	1	0	0	1	0	0	0	1	1	0	0	1	
0	1	1	0	1	0	0	0	0	1	0	0	0	1	0	1	1

Binary Sum

																0	1
0	1	1	0	1	0	0	1	0	1	1	1	0	0	0	0		
0	1	0	1	0	0	1	0	0	1	1	0	1	1	0	1		
0	1	1	0	1	0	0	1	0	0	0	1	1	0	0	1		
0	1	1	0	1	0	0	0	0	1	0	0	0	1	0	1		
																1	1

Binary Sum

1 1 1 1 0 1

0	1	1	0	1	0	0	1	0	1	1	1	0	0	0	0
0	1	0	1	0	0	1	0	0	1	1	0	1	1	0	1
0	1	1	0	1	0	0	1	0	0	0	1	1	0	0	1
0	1	1	0	1	0	0	0	0	1	0	0	0	1	0	1

1 1 1 0 1 1

Binary Sum

1 0 1 1 1 1 0 1

0	1	1	0	1	0	0	1	0	1	1	1	0	0	0	0
0	1	0	1	0	0	1	0	0	1	1	0	1	1	0	1
0	1	1	0	1	0	0	1	0	0	0	1	1	0	0	1
0	1	1	0	1	0	0	0	0	1	0	0	0	1	0	1

0 0 1 1 1 0 1 1

Binary Sum

1 1 0 1 1 1 0 1 1 1 1 0 1

0	1	1	0	1	0	0	1	0	1	1	1	0	0	0	0
0	1	0	1	0	0	1	0	0	1	1	0	1	1	0	1
0	1	1	0	1	0	0	1	0	0	0	1	1	0	0	1
0	1	1	0	1	0	0	0	0	1	0	0	0	1	0	1

0 1 1 0 1 0 0 1 1 1 0 1 1

Binary Sum

1 0 1 1 0 1 1 1 0 1 1 1 1 0 1

0	1	1	0	1	0	0	1	0	1	1	1	0	0	0	0
0	1	0	1	0	0	1	0	0	1	1	0	1	1	0	1
0	1	1	0	1	0	0	1	0	0	0	1	1	0	0	1
0	1	1	0	1	0	0	0	0	1	0	0	0	1	0	1

0 0 0 1 1 0 1 0 0 1 1 1 0 1 1

Binary Sum

1 1 0 1 1 0 1 1 1 0 1 1 1 1 0 1

0	1	1	0	1	0	0	1	0	1	1	1	0	0	0	0
0	1	0	1	0	0	1	0	0	1	1	0	1	1	0	1
0	1	1	0	1	0	0	1	0	0	0	1	1	0	0	1
0	1	1	0	1	0	0	0	0	1	0	0	0	1	0	1

1 0 0 0 1 1 0 1 0 0 1 1 1 0 1 1

Binary Sum

1 1 0 1 1 0 1 1 1 0 1 1 1 1 0 1

0	1	1	0	1	0	0	1	0	1	1	1	0	0	0	0
0	1	0	1	0	0	1	0	0	1	1	0	1	1	0	1
0	1	1	0	1	0	0	1	0	0	0	1	1	0	0	1
0	1	1	0	1	0	0	0	0	1	0	0	0	1	0	1

1 1 0 0 0 1 1 0 1 0 0 1 1 1 0 1 1

Modular Systems



TCP Segment Format

16		16	
Source Port		Destination Port	
Sequence Number			
Acknowledgment Number			
Hdr Len		Flags	Receiver window
Error Check		Urgent Pointer	
DATA			

Binary Sum

0	1	1	0	1	0	0	1	0	1	1	1	0	0	0	0
0	1	0	1	0	0	1	0	0	1	1	0	1	1	0	1
0	1	1	0	1	0	0	1	0	0	0	1	1	0	0	1
0	1	1	0	1	0	0	0	0	1	0	0	0	1	0	1

1 1 0 0 0 1 1 0 1 0 0 1 1 1 0 1 1
1

1 0 0 0 1 1 0 1 0 0 1 1 1 1 0 0

0 1 1 1 0 0 1 0 1 1 0 0 0 0 1 1