INFORMATION THEORY

Marco Dalai

DII - Office 6

Email: marco.dalai@unibs.it
Tel.: 0303715847
Office hours: send me an email

Teaching Materials

- **Textbook**:

  Elements of Information Theory, Thomas M. Cover, Joy A. Thomas, ed. John Wiley & Sons. (1-st or 2-nd edition)

  - It covers “almost” everything
  - Available in our library
  - Source of many problems/handouts available online
  - Additional very interesting topics included (Gambling, Stock Market, Statistics, Kolmogorov Theory...)

Teaching Materials

- **Other books:**
  - Robert. G. Gallager, *Information Theory and Reliable Communication*
  - Robert M. Fano. *Transmission of Information: A Statistical Theory of Communications*.
  - Andrew J. Viterbi, Jim K. Omura. *Principles of Digital Communication and Coding*
  - Robert Ash, *Information Theory*
  - John Pierce, *An Introduction to Information Theory*
  - David MacKay, *Information Theory, Inference, and Learning Algorithms*, available online for free
    http://www.inference.phy.cam.ac.uk/mackay/itprnn/book.html
- **Shannon's seminal paper**:
  A Mathematical Theory of Communication, Bell System Technical Journal, vol. 27, pp. 379-423 and 623-656, July and October, 1948.
  http://www.essrl.wustl.edu/~jao/itrg/shannon.pdf

Exam

- Written exam covering both “theory” and “practice”
- Possible oral examination
- Optional project for 0 to 3 additional points

Now… a preview

What is a *bit*?

What is a *bit*?

What is a *bit*?

Here they are

00000000000000000000000000000000000000001000000010000000
00000000001000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000
00000000000000000100000000000000000000000000000000000100
00000100000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000……
……

What is a *bit*?

Here they are

00000000000000000000000000000000000000000000100000000100000000
00000000001000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000
00000000000000000010000000000000000000000000000000000000000100
00000100000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000……
……

Is this 5GB of *information*?

Bits & Information

ASCII Code… `"Call me as soon as possible"`

```
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
1 1 1 1 0 1 1 0 1 1 0 1 1 1 1 0 1 1 0 1 1 1 1 1 1 1 1
0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
0 0 0 0 0 0 0 0 1 0 1 0 0 0 0 0 1 0 1 0 1 1 0 0 0 0 0
0 0 1 1 0 1 0 0 0 0 0 0 1 1 1 0 0 0 0 0 1 0 0 1 0 1 0
0 0 1 1 0 1 1 0 0 0 0 0 0 1 1 1 0 0 0 0 0 1 0 0 0 0 1 1
1 0 0 0 0 0 0 0 0 1 0 1 1 1 1 0 0 1 0 0 1 1 1 0 1 0 0
1 1 0 0 0 1 1 0 1 1 0 1 1 1 0 0 1 1 0 0 1 1 1 1 0 0 1
```

27x8=216 bits?
that is
8 *bits*/character?

Compression

??

```
Cal asap
0 0 0 0 0 0 0 0
1 1 1 0 1 1 1 1
0 1 1 1 1 1 1 1
0 0 0 0 1 0 1
0 0 1 0 0 0 0 0
0 0 1 0 0 0 0 0
1 0 0 0 0 1 0 0
1 1 0 0 1 1 1 0
```

8x8=64 bit?
that is
≈1.728 *bits*/character?

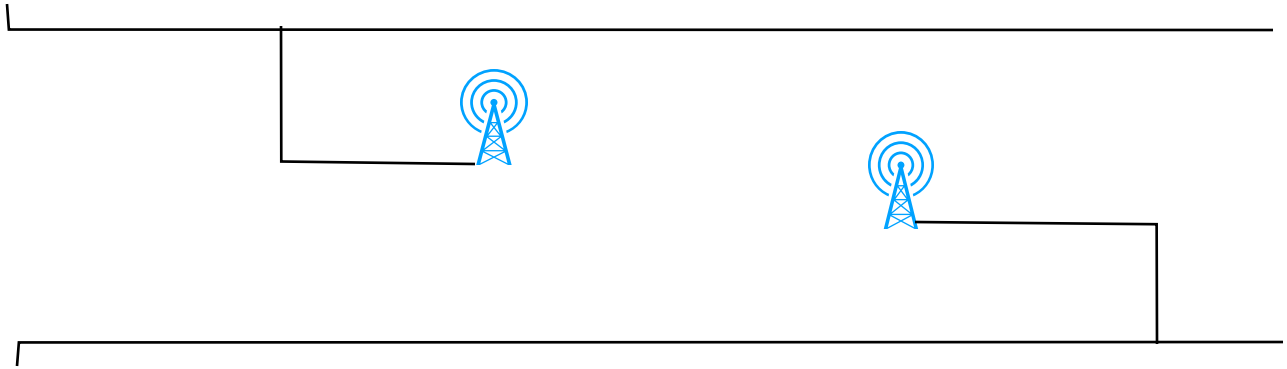
Language is *redundant*… how many *bits* are actually needed?

Bits & Information

00100101001010110101111000101000011001110101110100 **50bits**

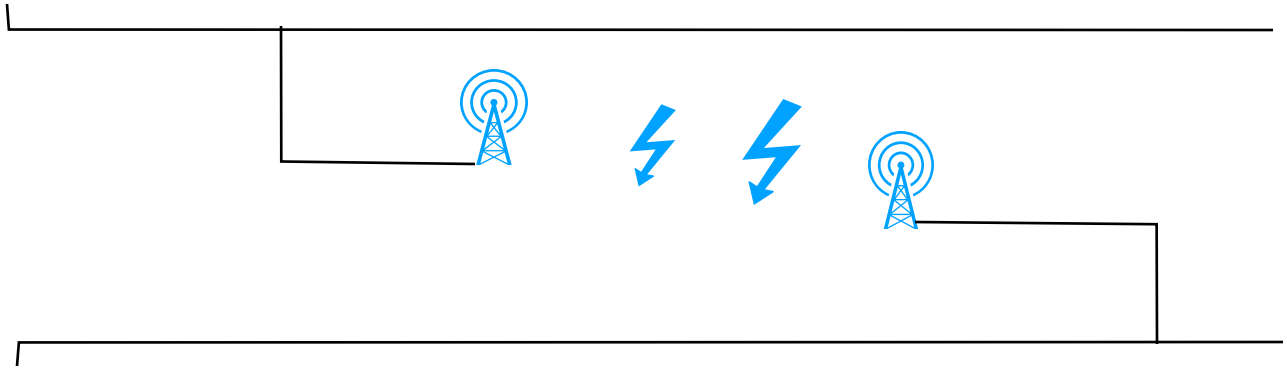
Bits & Information

00100101001010110101111000101000011001110101110100 **50bits**



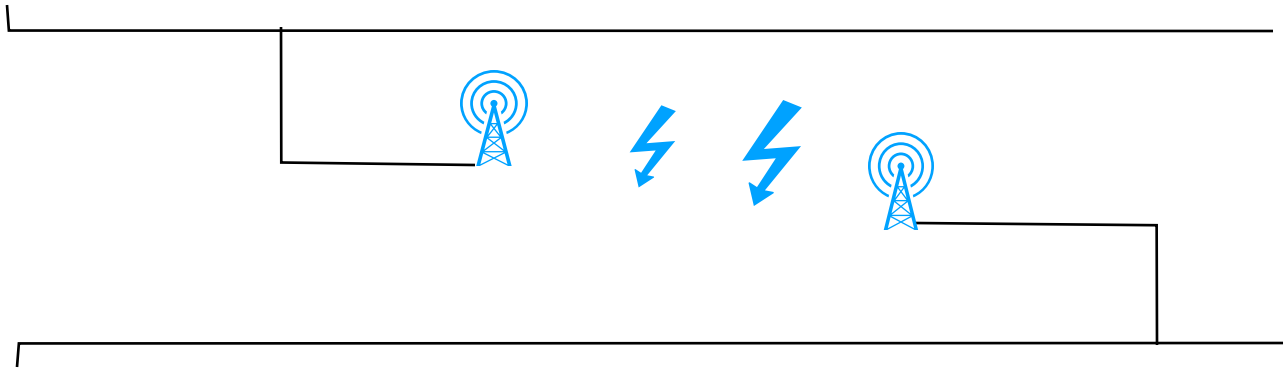
Bits & Information

00100101001010110101111000101000011001110101110100 **50bits**



Bits & Information

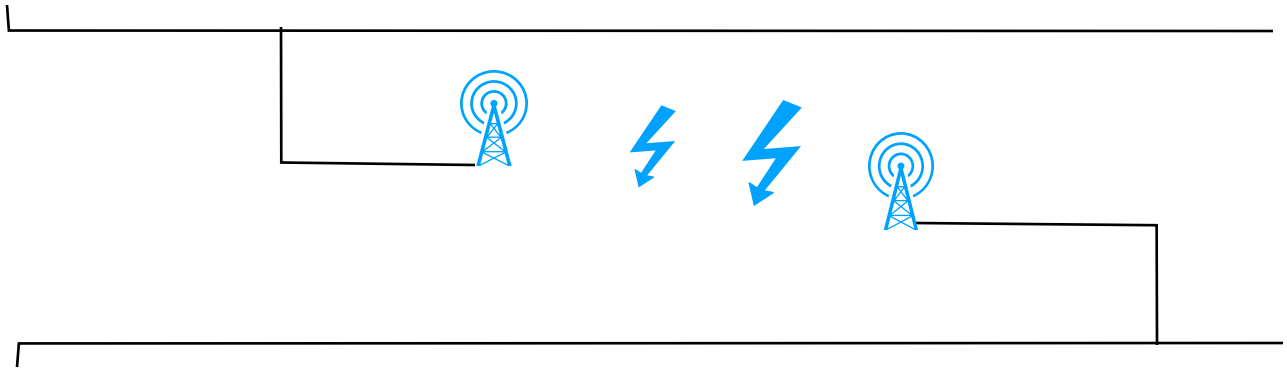
00100101001010110101111000101000011001110101110100 50bits



00110101100010110001111000110000100011101111110100 10% errors

Bits & Information

00100101001010110101111000101000011001110101110100 50bits

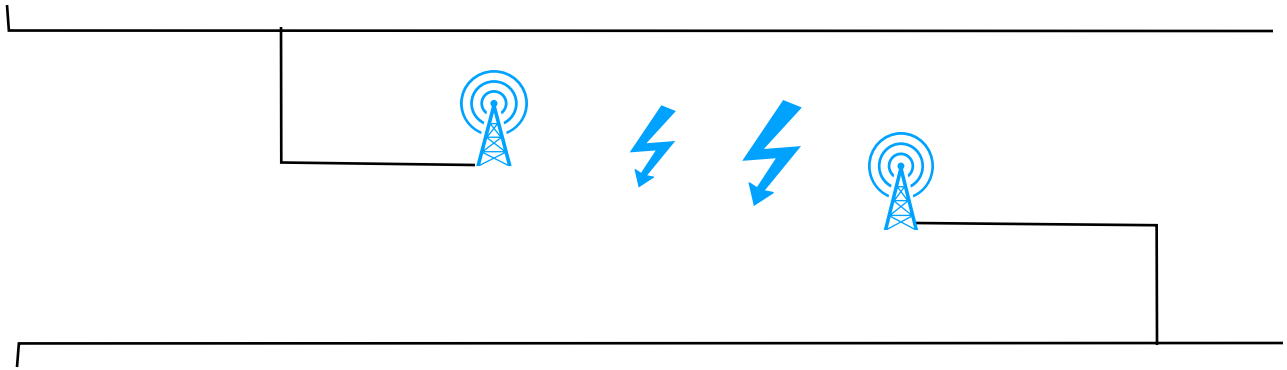


00110101100010110001111000110000100011101111101 00 10% errors

Did we really send 50 *bits*?

Bits & Information

00100101001010110101111000101000011001110101110100 50bits

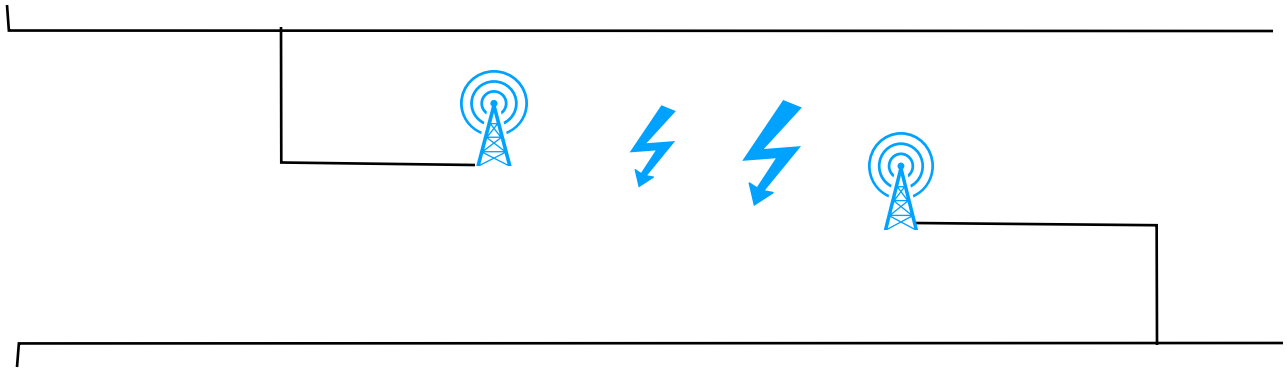


00110101100010110001111000110000100011101111101 10% errors

Did we really send 50 *bits*?
Maybe 45?

Bits & Information

00100101001010110101111000101000011001110101110100 50bits



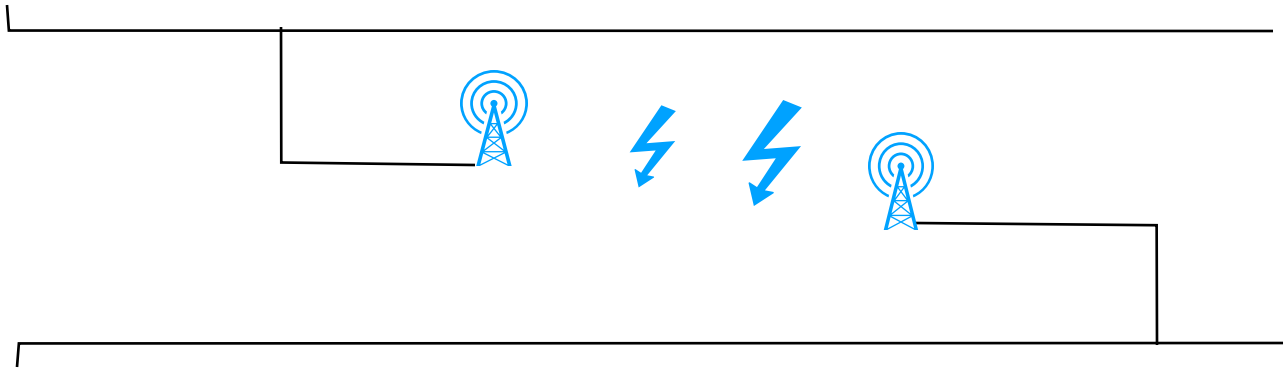
00110101100010110001111000110000100011101111101 00 10% errors

Did we really send 50 *bits*?
Maybe 45?

Loss of *bits?*

Bits & Information

00100101001010110101111000101000011001110101110100 50bits



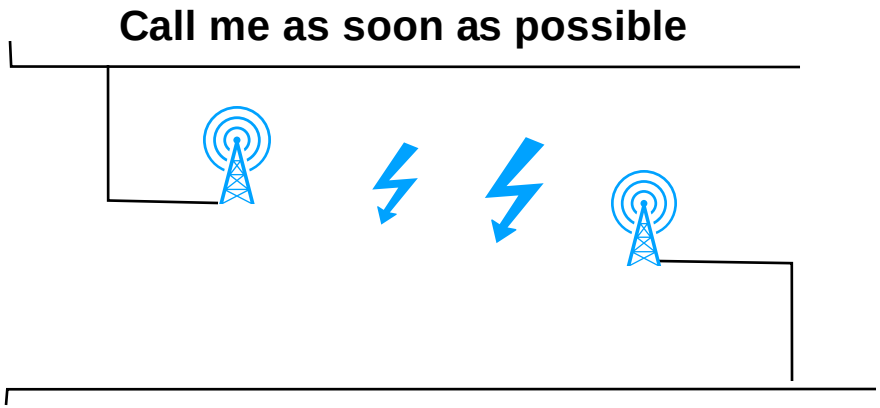
00110101100010110011111000110000100011101111110100 10% errors

Did we really send 50 *bits*?
Maybe 45?

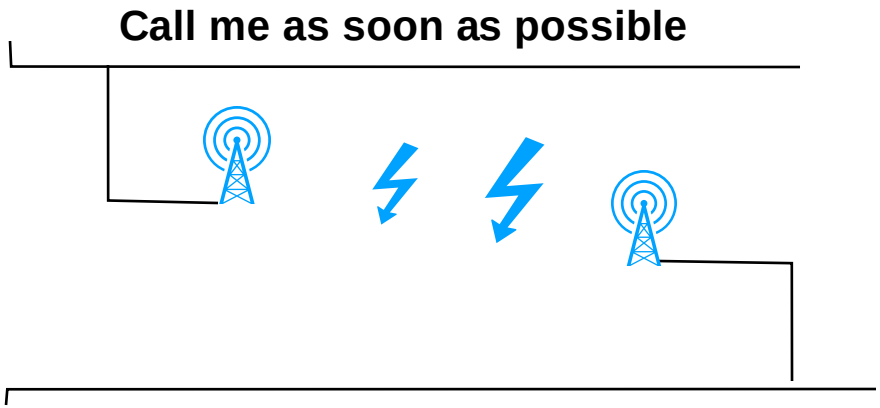
Loss of *bits?*

Loss of *information*?

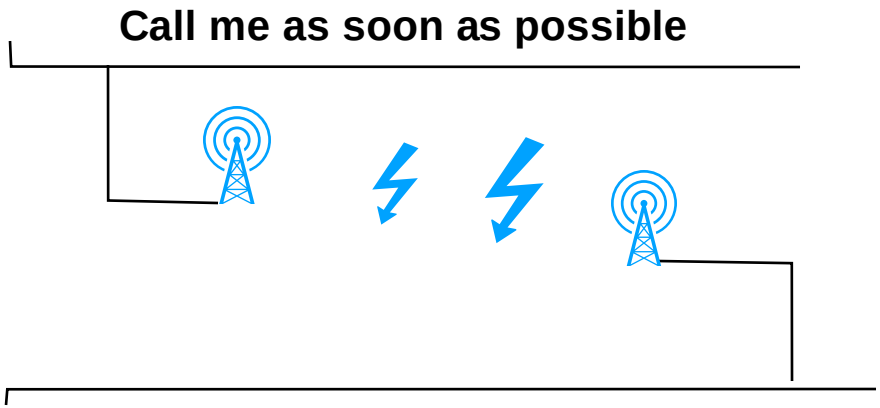
Information loss?

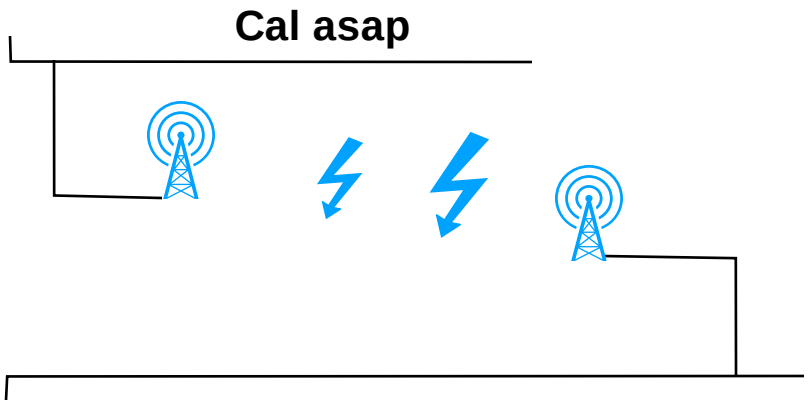


Information loss?

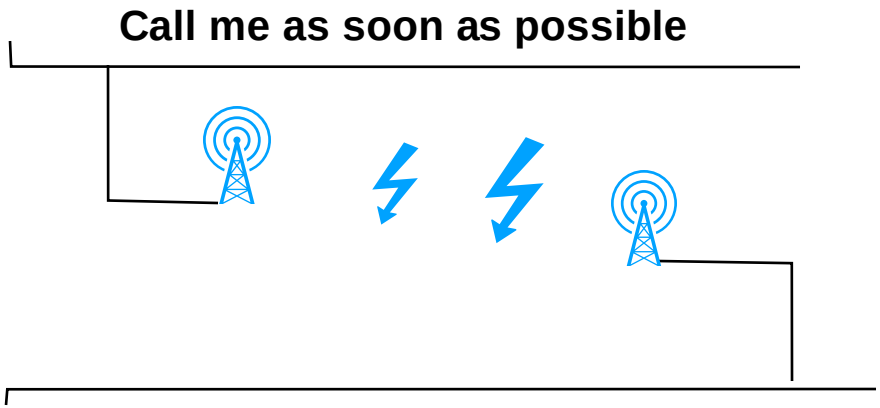


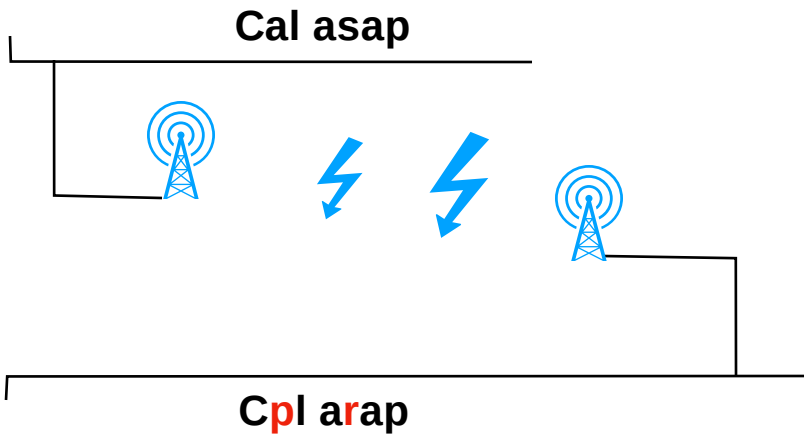
Information loss?





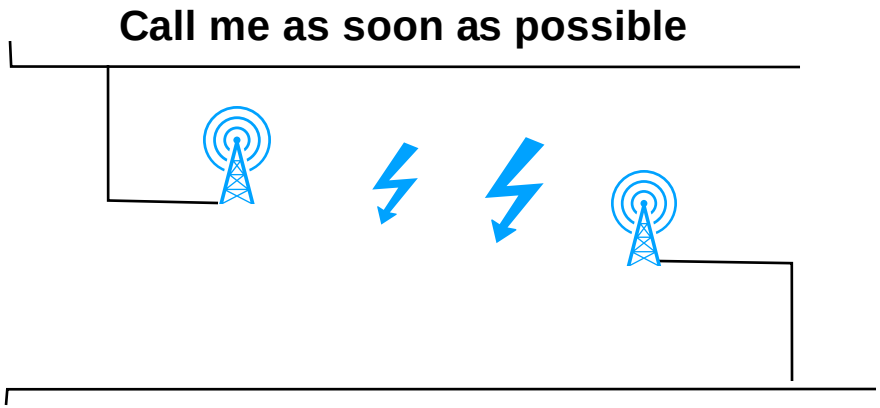
Information loss?

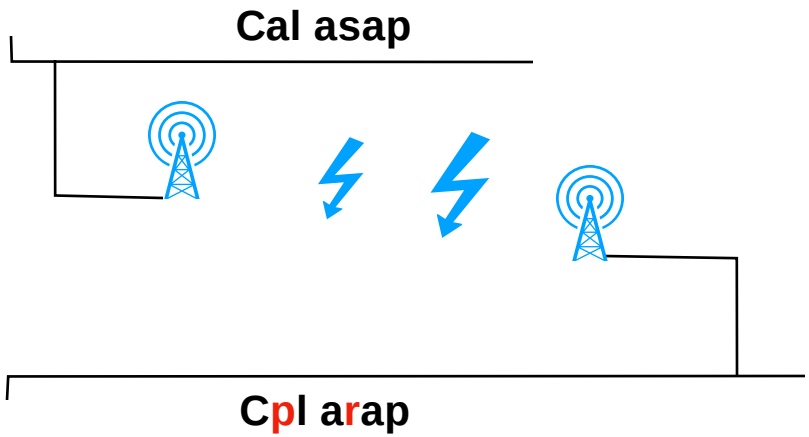




Information loss?

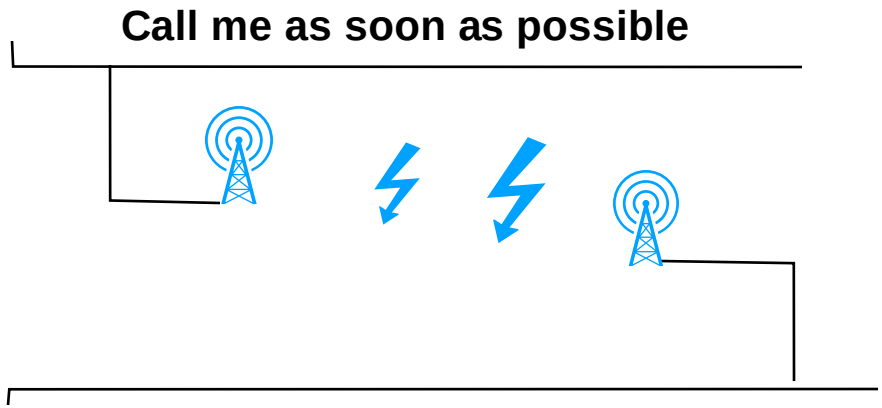
Redundant
=
robust



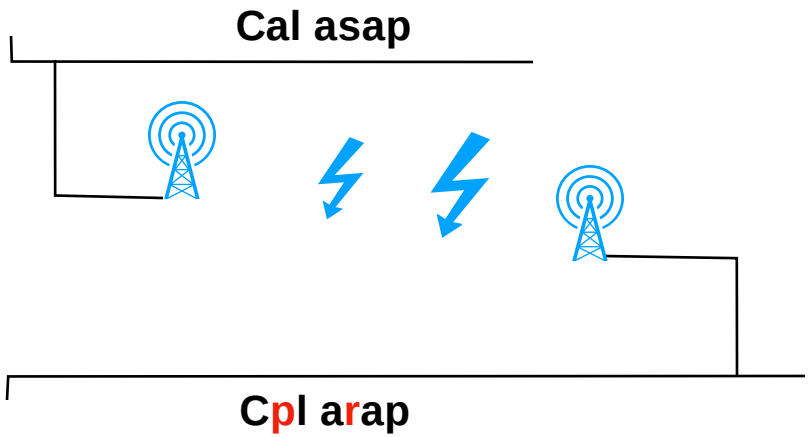


Information loss?

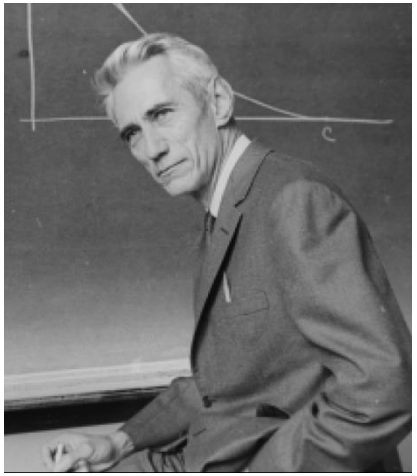
Redundant
=
robust



Compressed
=
fragile

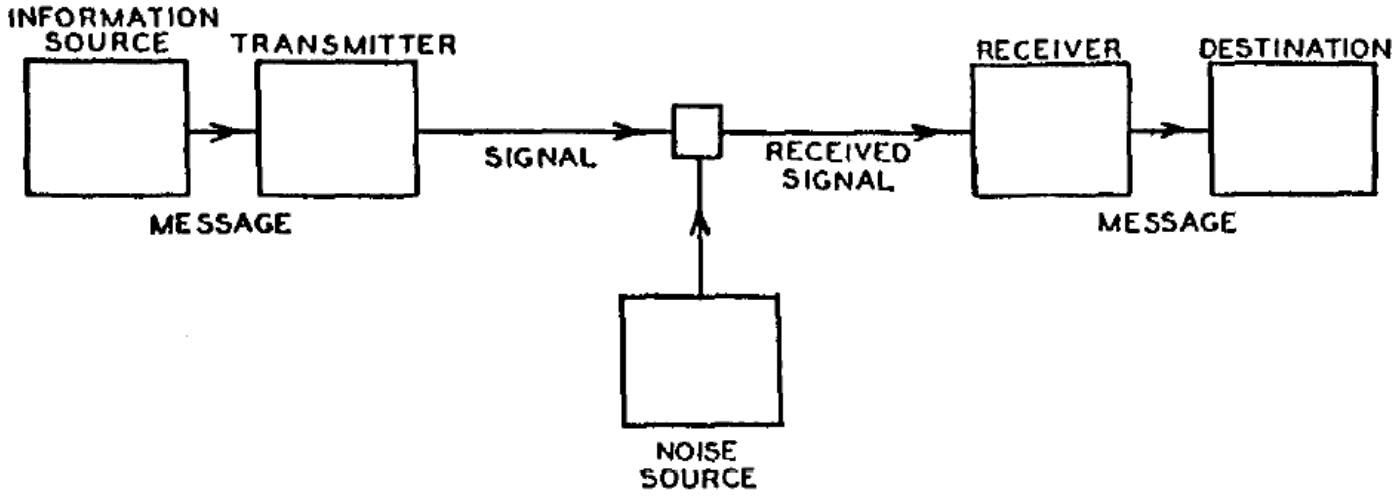


Communication Systems



C. E. Shannon, 1948

"The fundamental problem of communication is that of reproducing at one point either exactly or approximately a message selected at another point. […]
The significant aspect is that the actual message is one selected from a set of possible messages."



Communication Systems

(….before 1948)

Communication systems were already used extensively:

- Telegraph (Morse, 1830)
- Telephone (Meucci, 1871 – Bell, 1876)
- Wireless Telegraph (Marconi, 1887)
- AM radio (early ’900)
- Television (1925-1927)
- FM radio (Armstrong, 1936)
- Pulse-Coded Modulation (Reeves, 1937-1939)
- Spread Spectrum (’40s)

Communication Systems

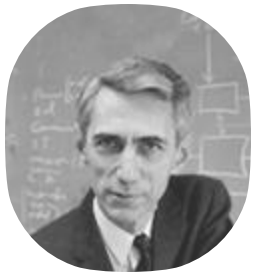
- …but there was no solid theory
  - Many heuristic results were in use
  - Few theoretical limits were known
  - It was not easy to understand the efficiency of used systems

- In 1948 Shannon gave birth to a whole new science

  "The Mathematical Theory of Communication … came as a bomb"

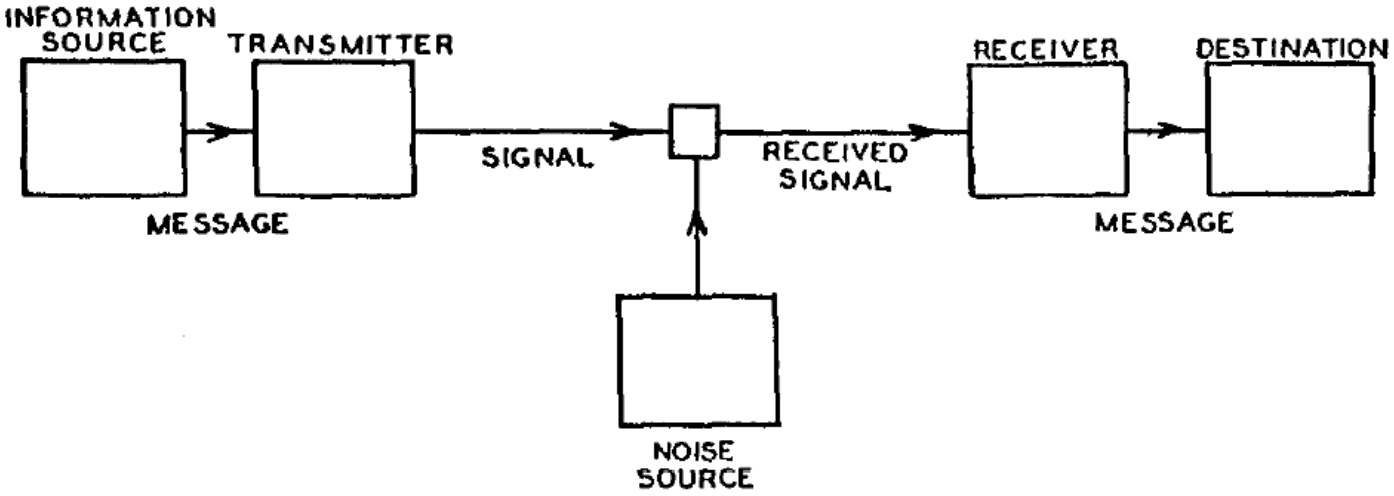
  J. R. Pierce

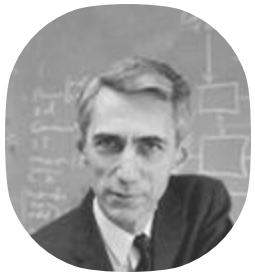
  - He defined a *measure of information*
  - He found theoretical limits for the performance of communication and coding systems



Shannon 1948

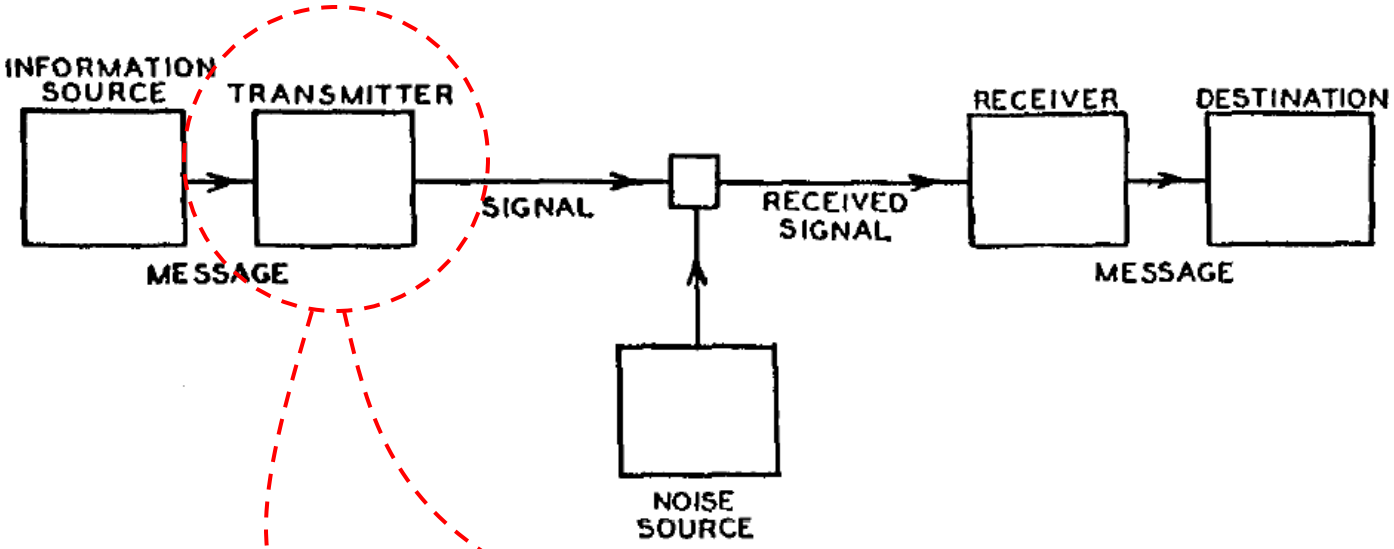
Redundancy and Coding

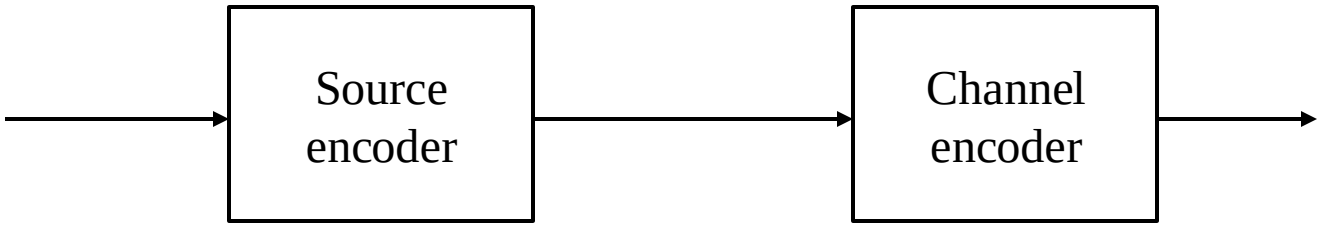


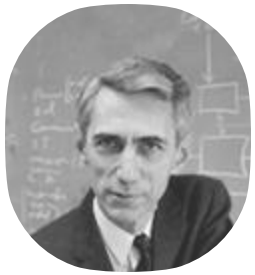


Shannon 1948

Redundancy and Coding

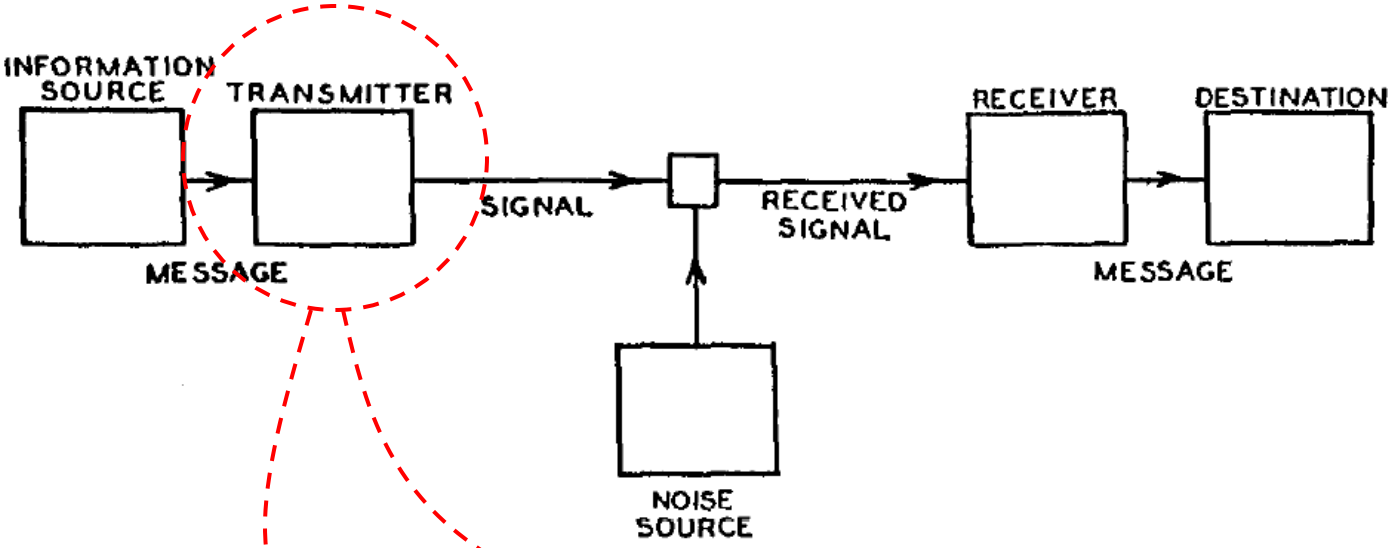


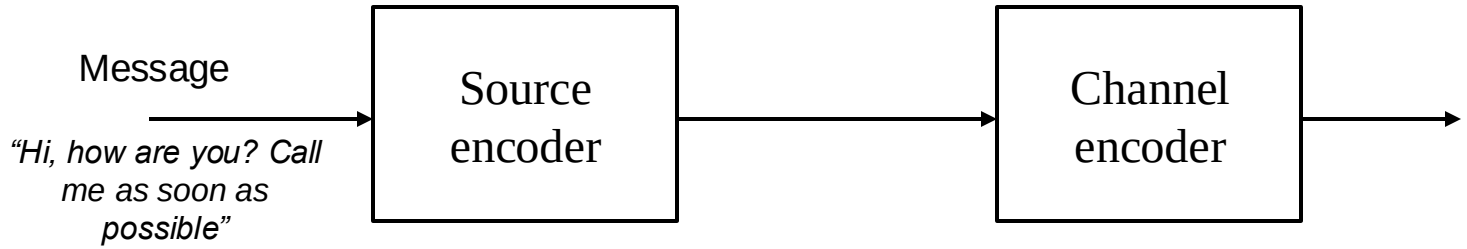


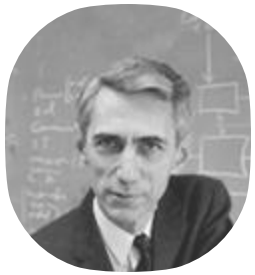


Shannon 1948

Redundancy and Coding

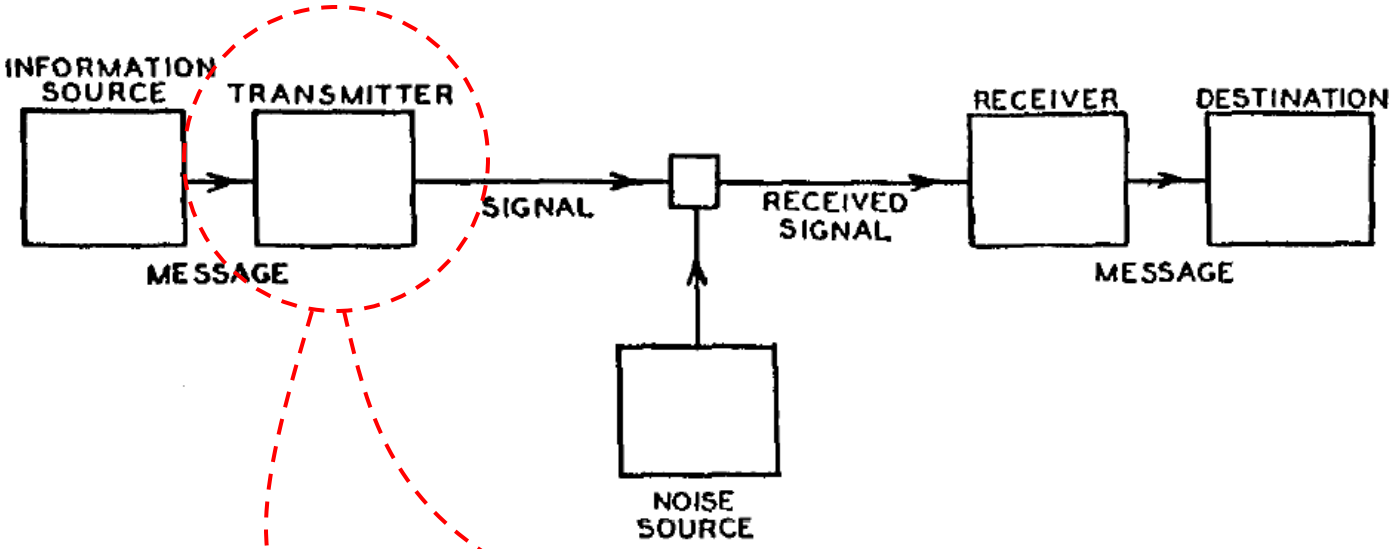






Shannon 1948

Redundancy and Coding



Message

"Hi, how are you? Call me as soon as possible"

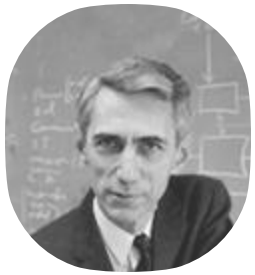
Source encoder

It removes the source redundancy

Source code

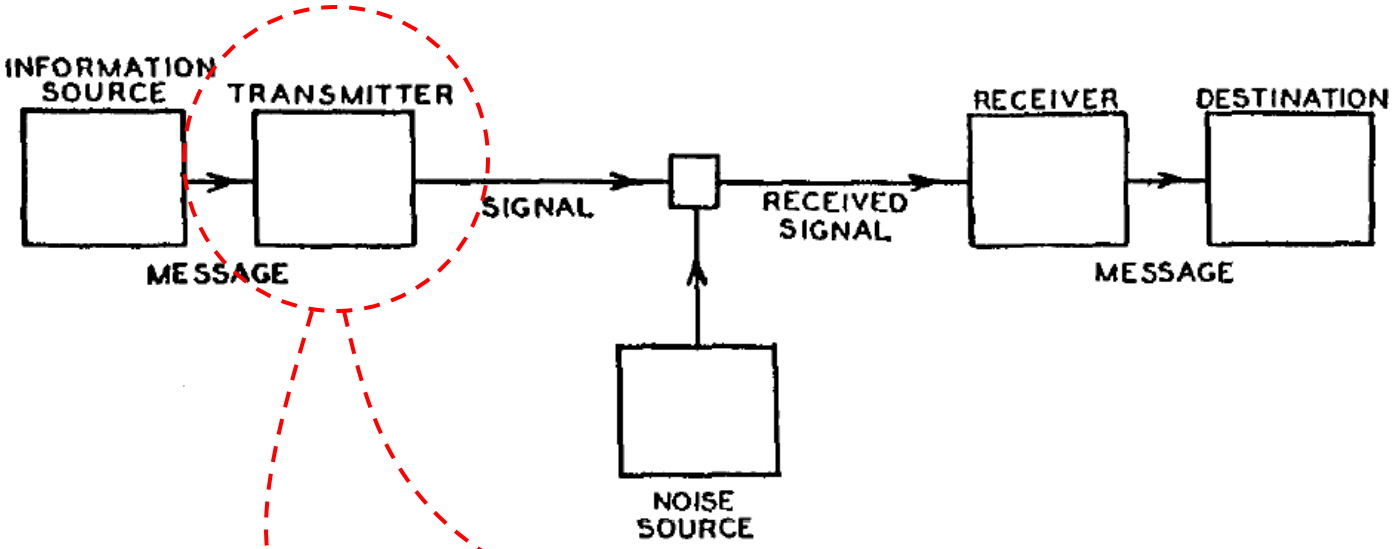
H, hw r u? Cal asap"

Channel encoder



Shannon 1948

Redundancy and Coding



Message

"Hi, how are you? Call me as soon as possible"

Source encoder

It removes the source redundancy

Source code

H, hw r u? Cal asap"

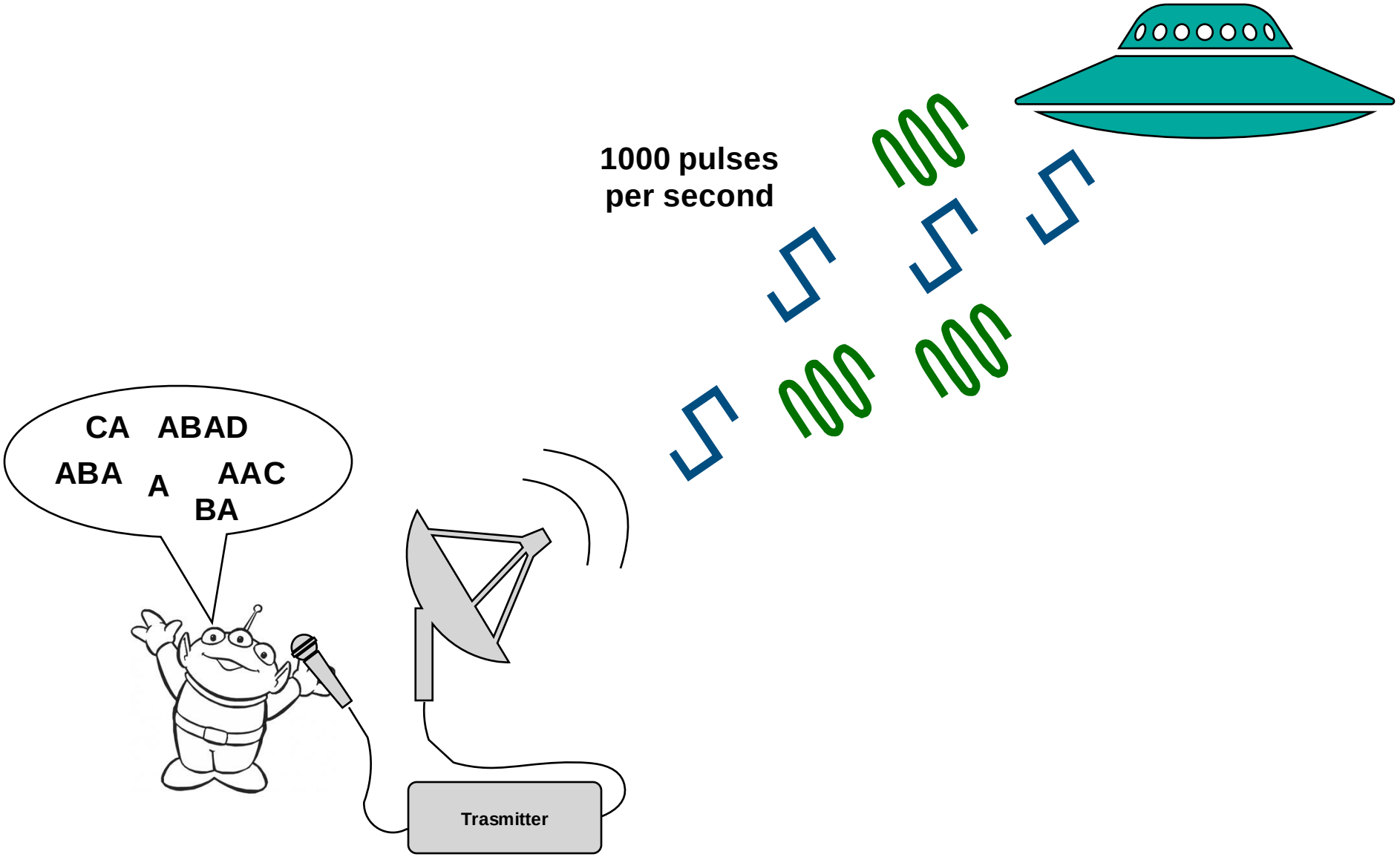
Channel encoder

It adds redundancy to reduce the effects of the channel errors

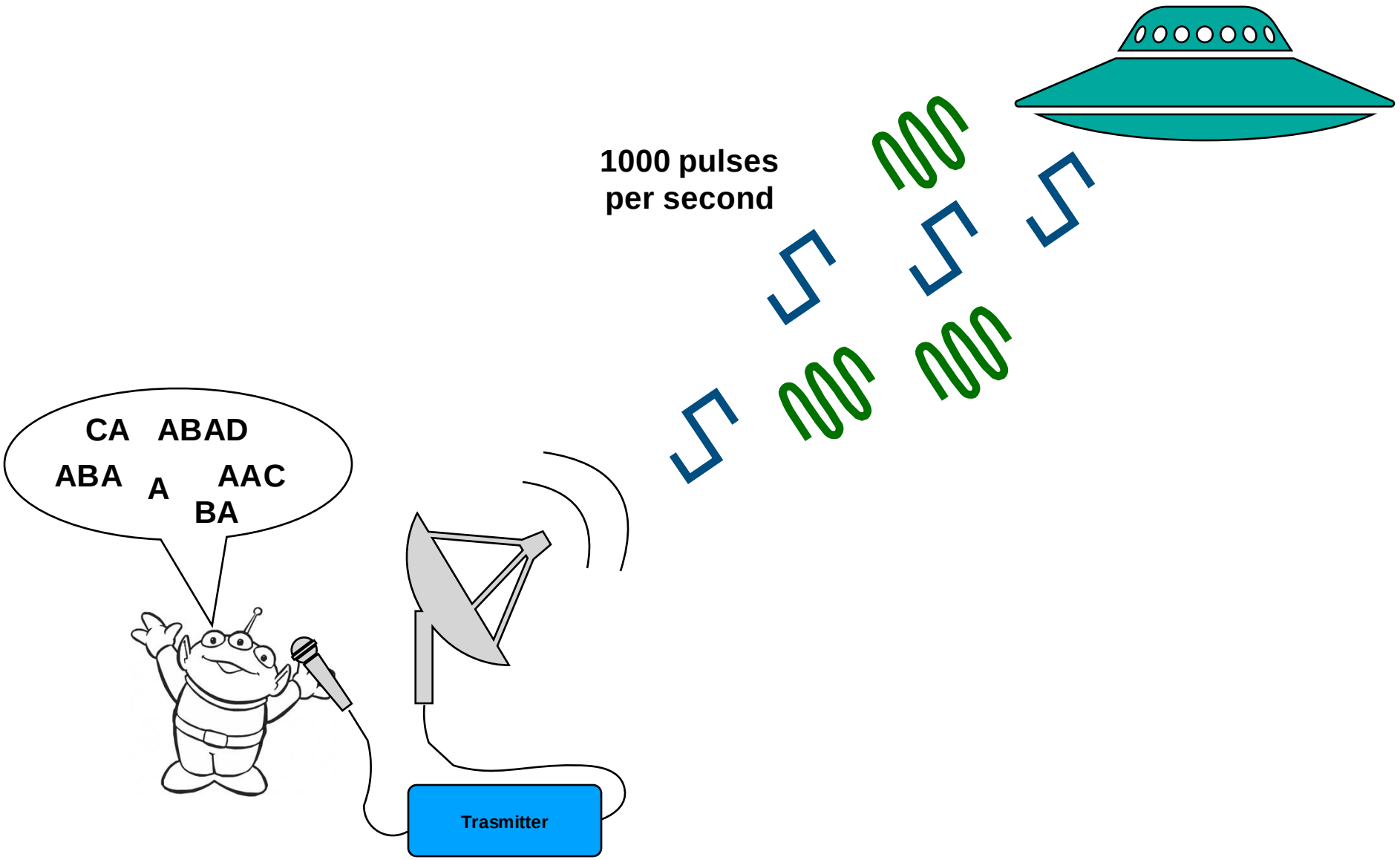
Cannel code

Hhh, hhhww rrr uu? Cccalll aaasssaaappp

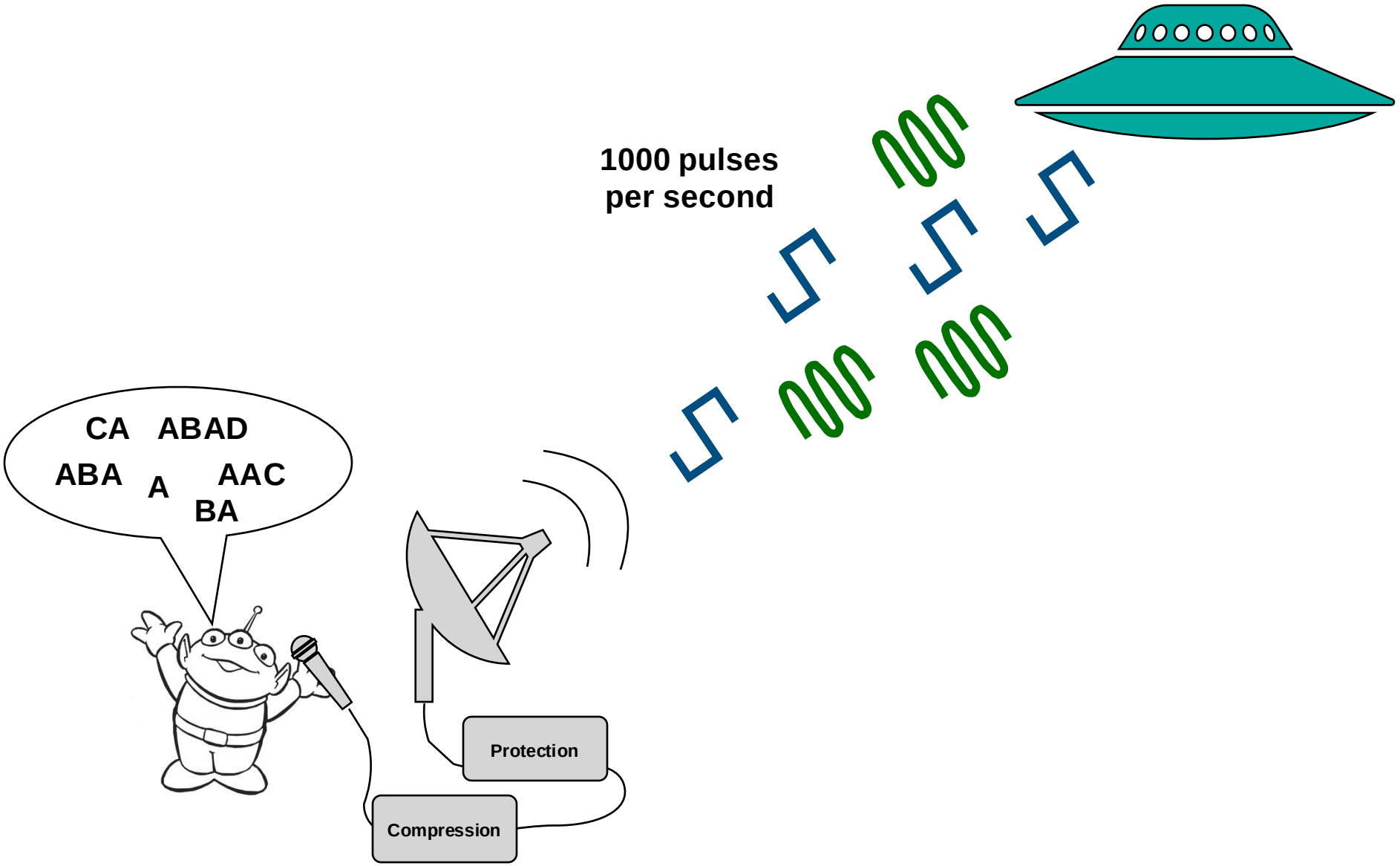
Transmission of Information



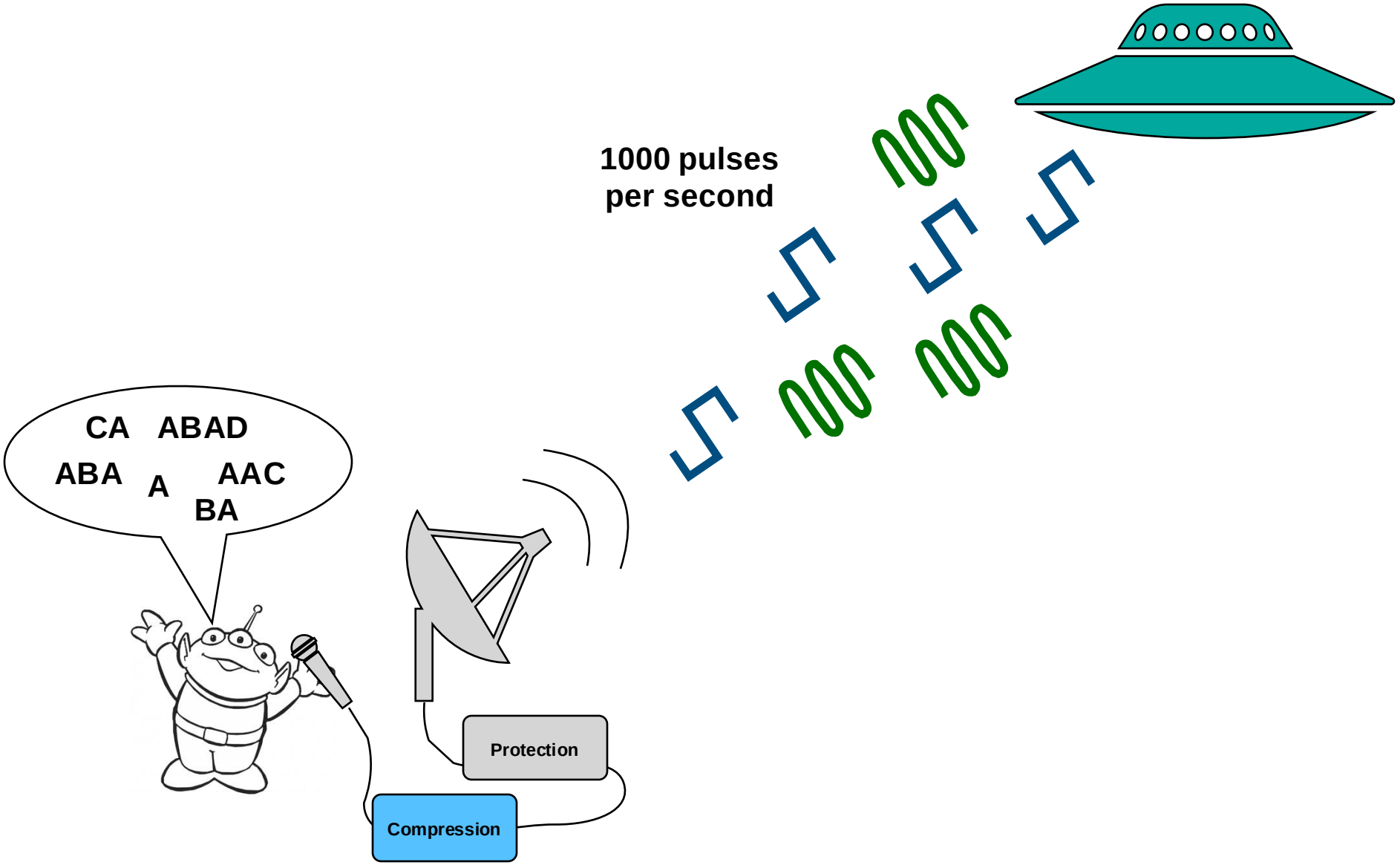
Transmission of Information



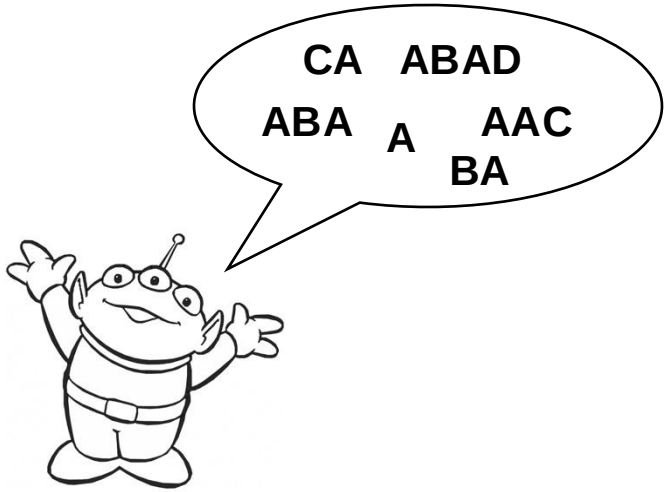
Transmission of Information



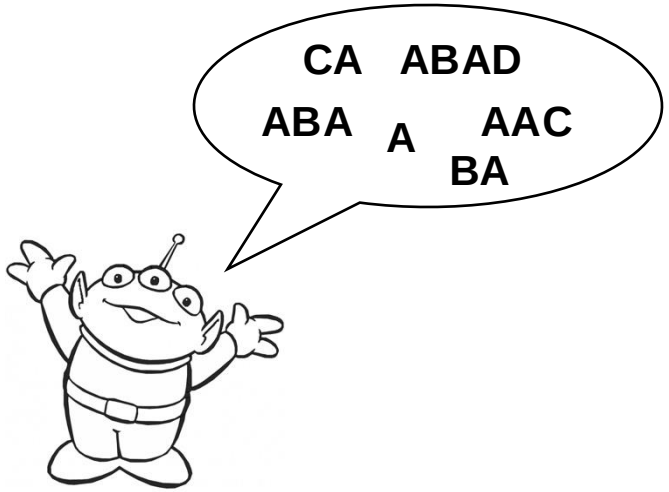
Transmission of Information



Compression



Compression

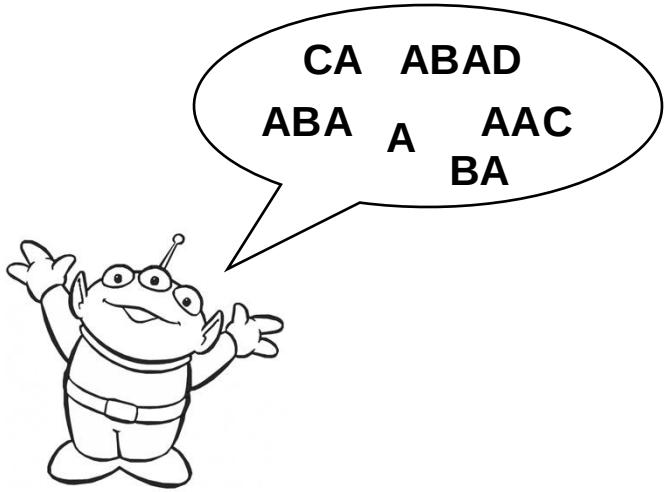


Hartley 1928

4 symbols… 2 bits

Alphabet:		Code:
A	→	00
B	→	01
C	→	10
D	→	11

Compression



Hartley 1928

4 symbols… 2 bits

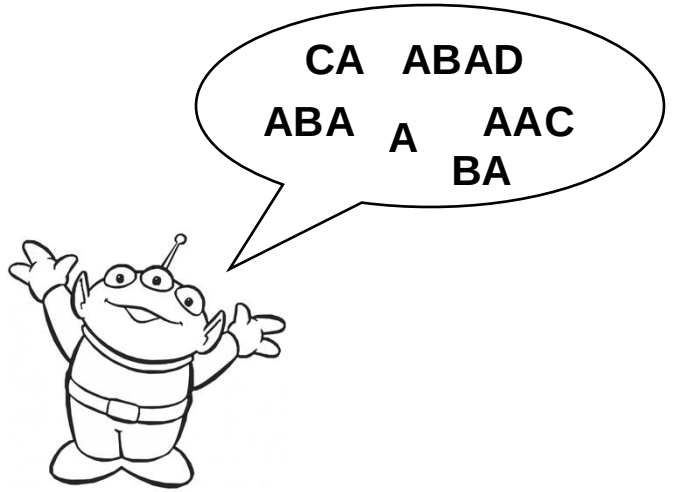
Alphabet:		Code:
A	→	00
B	→	01
C	→	10
D	→	11

A A B A D A A A B B C A C D B A

00 00 01 00 10 00 00 00 01 01 10 00 10 11 01 00

… 2 bits/symbol
Clear!

Compression



4 symbols… 2 bits

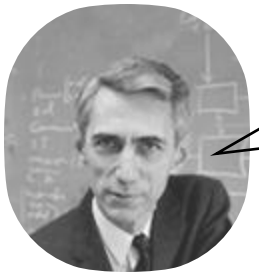
Hartley 1928

Alphabet:		Code:
A	→	00
B	→	01
C	→	10
D	→	11

A A B A D A A A B B C A C D B A

00 00 01 00 10 00 00 00 01 01 10 00 10 11 01 00

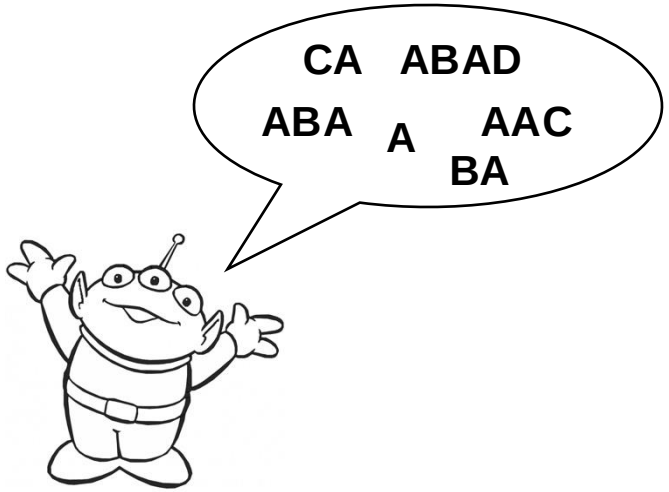
… 2 bits/symbol
Clear!



Let us check how often we use them!

Shannon 1948

Compression



4 symbols… 2 bits

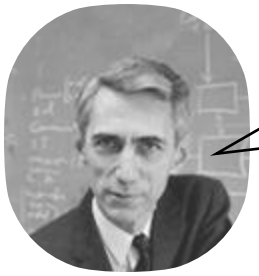
Hartley 1928

Alphabet:		Code:
A	→	00
B	→	01
C	→	10
D	→	11

A A B A D A A A B B C A C D B A

00 00 01 00 10 00 00 00 01 01 10 00 10 11 01 00

… 2 bits/symbol
Clear!

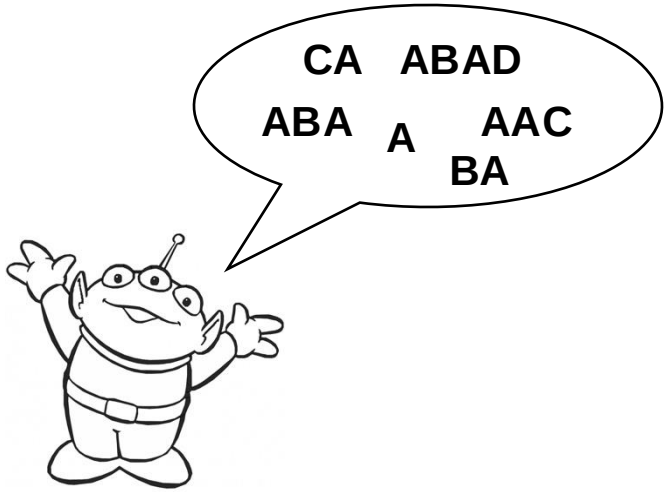


Let us check how often we use them!

Shannon 1948

A	50%	p(A)=1/2
B	25%	p(B)=1/4
C	12.5%	p(C)=1/8
D	12.5%	p(D)=1/8

Compression



Fano 1948

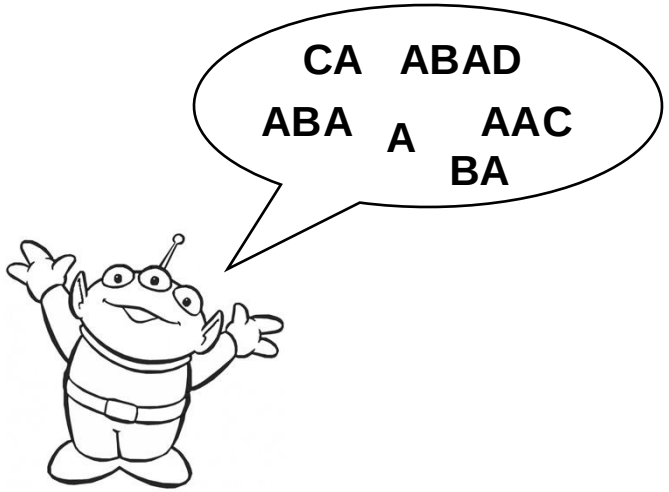
	Code:
p(A)=1/2 →	0
p(B)=1/4 →	10
p(C)=1/8 →	110
p(D)=1/8 →	111

A A B A D A A A B B C A C D B A

0 0 10 0 111 0 0 0 10 10 110 0 110 111 10 0

… 7/4 bits /symbol
Better!

Compression



Fano 1948

		Code:
p(A)=1/2	→	0
p(B)=1/4	→	10
p(C)=1/8	→	110
p(D)=1/8	→	111

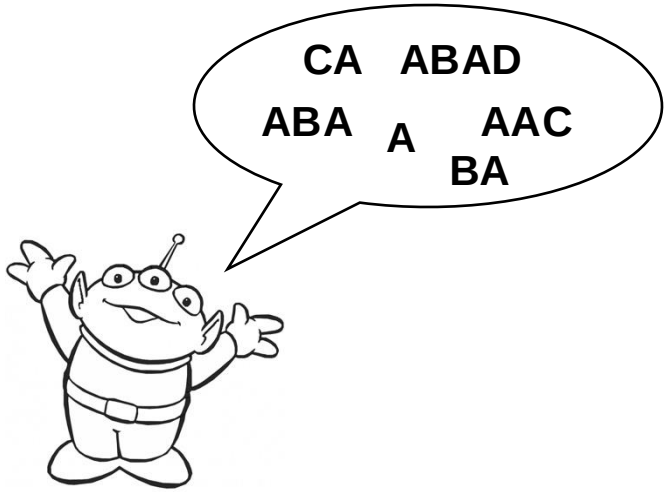
A A B A D A A A B B C A C D B A

0 0 10 0 111 0 0 0 10 10 110 0 110 111 10 0

… 7/4 bits /symbol
Better!

Can we do even better?

Compression



Fano 1948

Code:

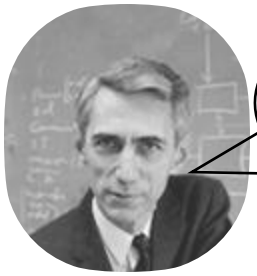
p(A)=1/2	→	0
p(B)=1/4	→	10
p(C)=1/8	→	110
p(D)=1/8	→	111

A A B A D A A A B B C A C D B A

0 0 10 0 111 0 0 0 10 10110 0 11011110 0

… 7/4 bits /symbol
Better!

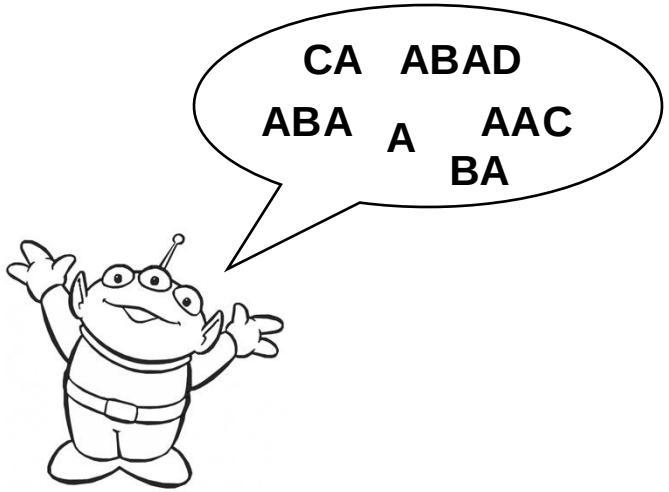
Can we do even better?



If symbols are independent… No!

Shannon 1948

Compression



Fano 1948

Code:

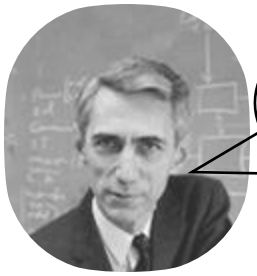
p(A)=1/2 → 0
p(B)=1/4 → 10
p(C)=1/8 → 110
p(D)=1/8 → 111

A A B A D A A A B B C A C D B A

0 0 10 0 111 0 0 0 10 10110 0 11011110 0

… 7/4 bits /symbol
Better!

Can we do even better?



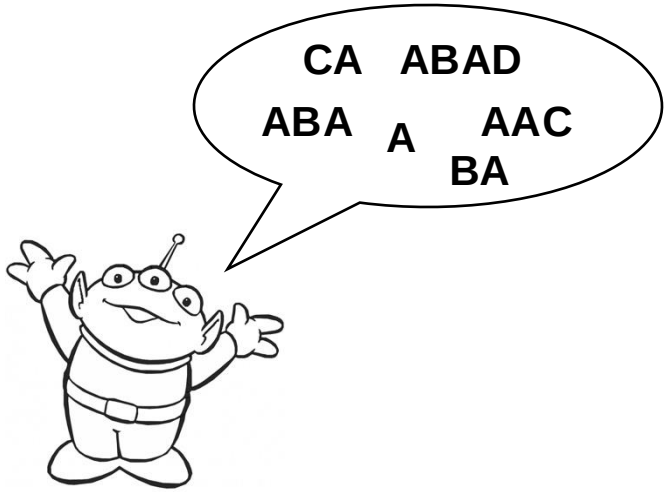
Shannon 1948

If symbols are independent… No!

Entropy of the Source

$$\sum p \log_2 \frac{1}{p} = \frac{1}{2}\log_2 2 + \frac{1}{4}\log_2 4 + \frac{1}{4}\log_2 4 + \frac{1}{8}\log_2 8 = \mathbf{7/4}$$

Compression



Fano 1948

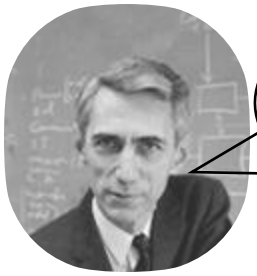
		Code:
p(A)=1/2	→	0
p(B)=1/4	→	10
p(C)=1/8	→	110
p(D)=1/8	→	111

A A B A D A A A B B C A C D B A

0 0 10 0 111 0 0 0 10 10110 0 11011110 0

… 7/4 bits /symbol
Better!

Can we do even better?



Shannon 1948

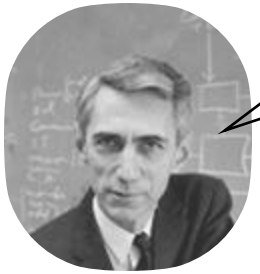
If symbols are independent… No!

Entropy of the Source

$$\sum p \log_2 \frac{1}{p} = \frac{1}{2}\log_2 2 + \frac{1}{4}\log_2 4 + \frac{1}{4}\log_2 4 + \frac{1}{8}\log_2 8 = \mathbf{7/4}$$

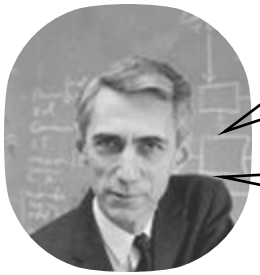
Information

A symbol of probability p
gives
$$\log_2 \frac{1}{p}$$
bits of information



Shannon 1948

Information

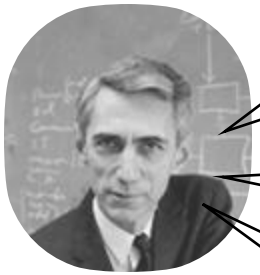


Shannon 1948

A symbol of probability p
gives
$$\log_2 \frac{1}{p}$$
bits of information

But it only appears a fraction p
of the times!

Information



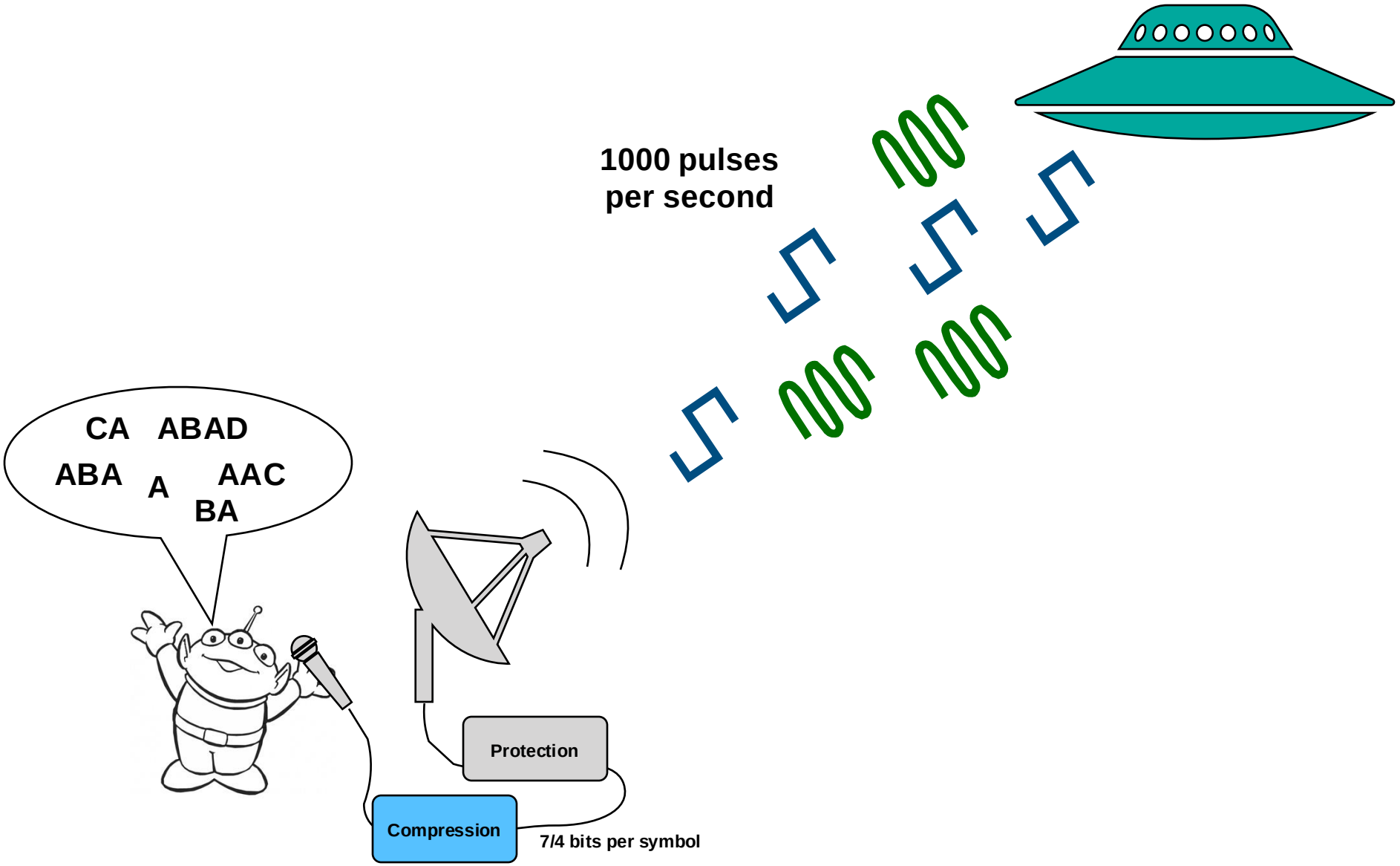
Shannon 1948

A symbol of probability p
gives
$$\log_2 \frac{1}{p}$$
bits of information

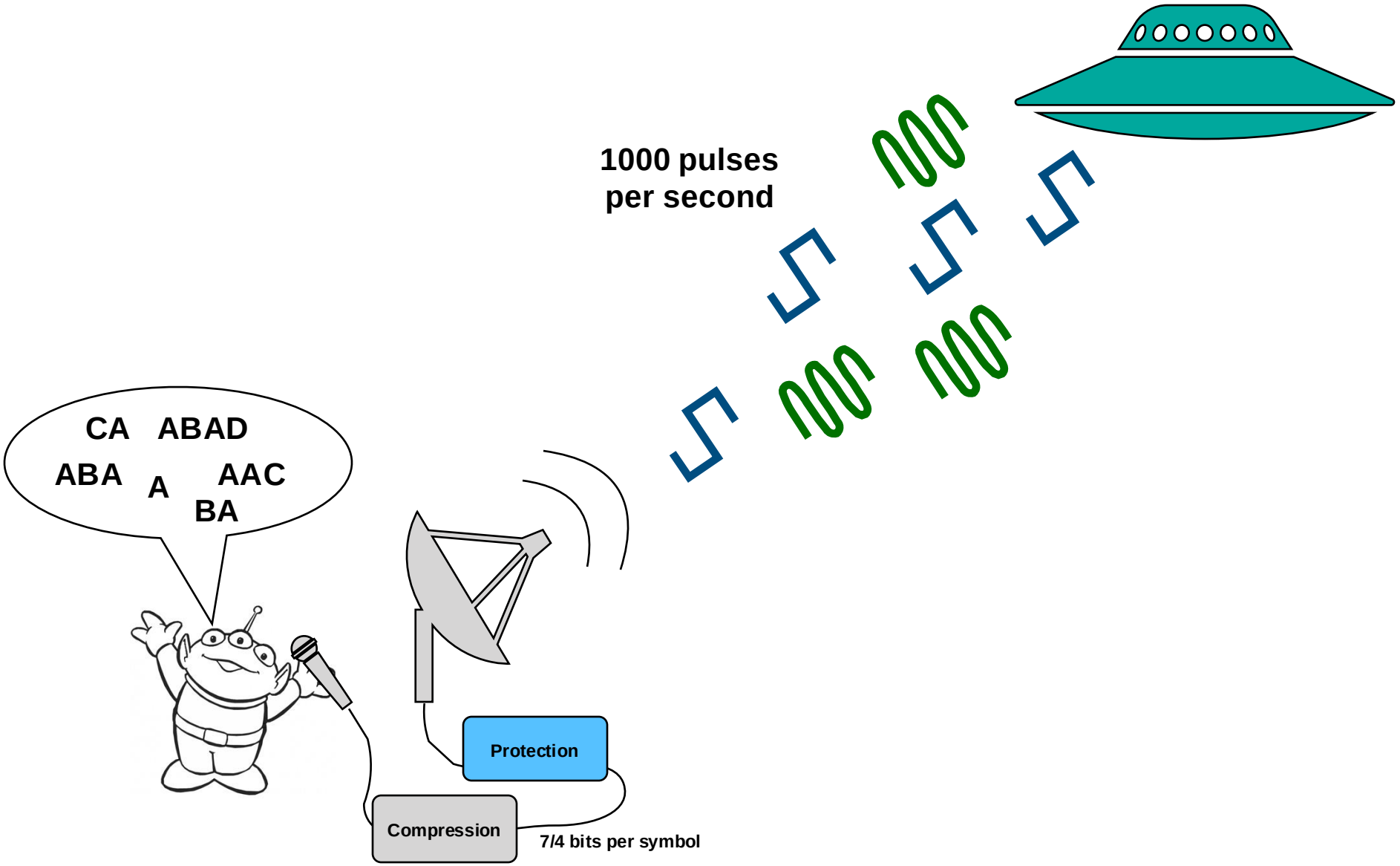
But it only appears a fraction p
of the times!

So, overall it brings an average of
$$p\log_2 \frac{1}{p}$$
bits!

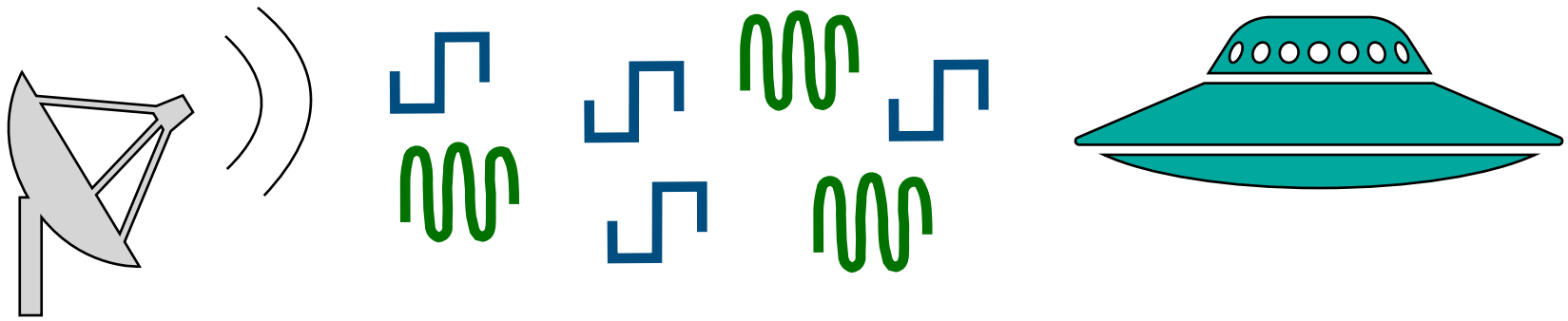
Transmission of Information



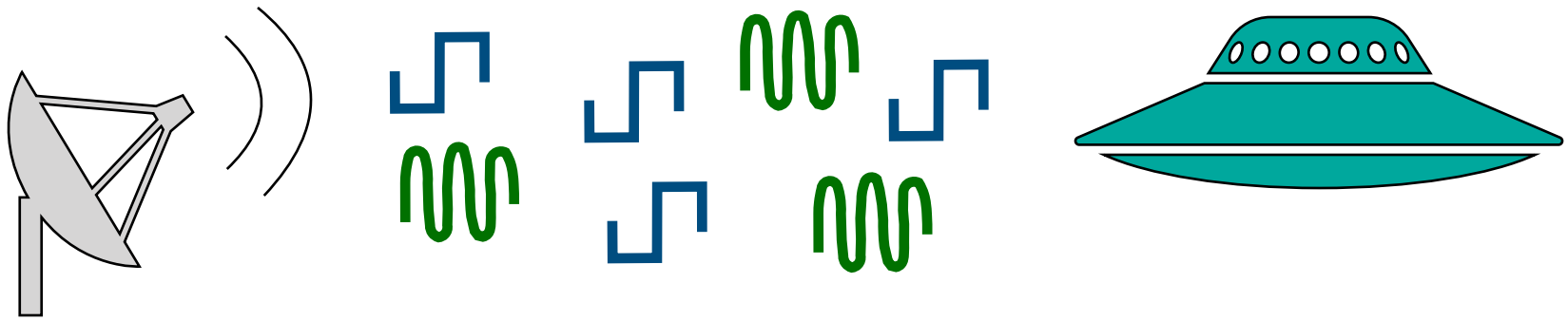
Transmission of Information

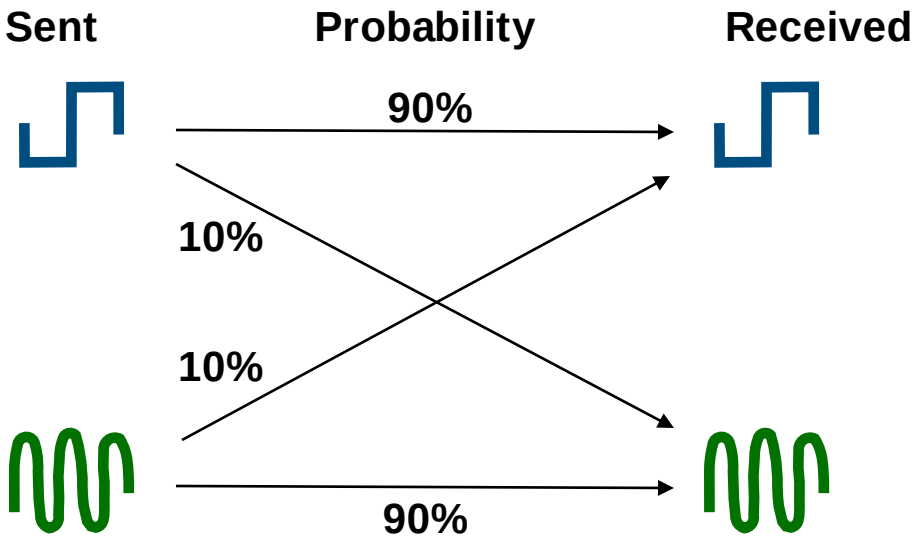


Channel

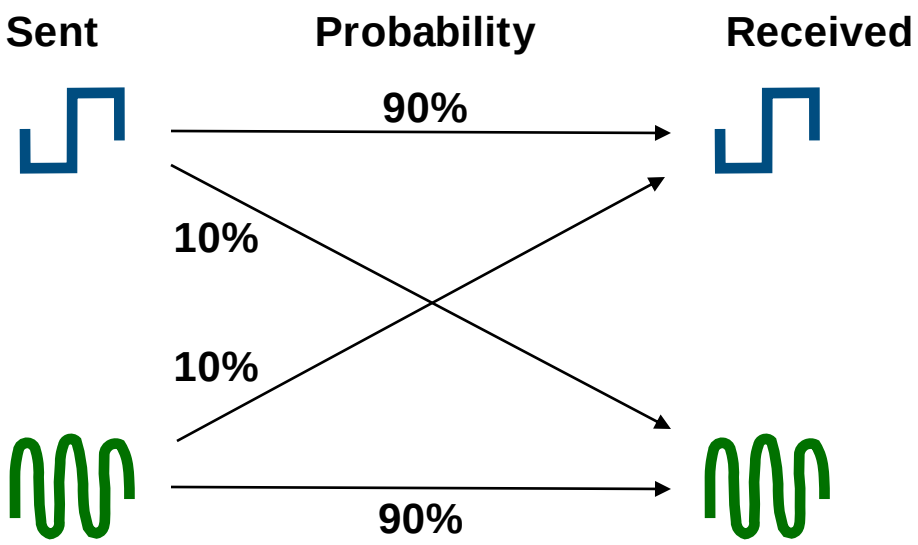


Channel

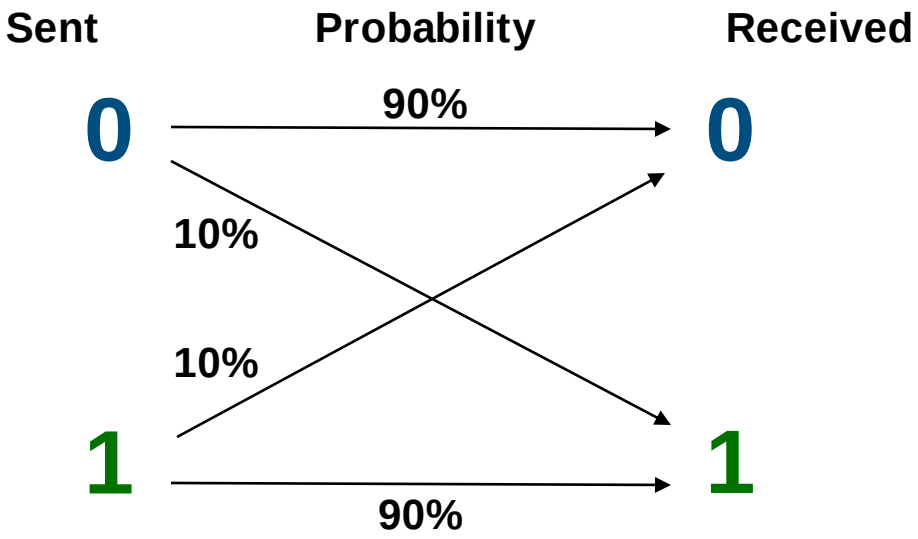




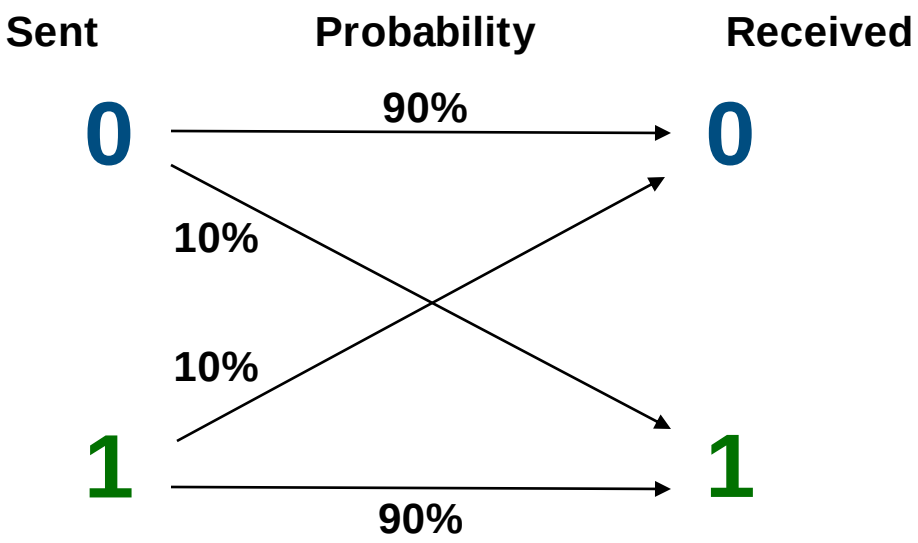
Channel



Channel

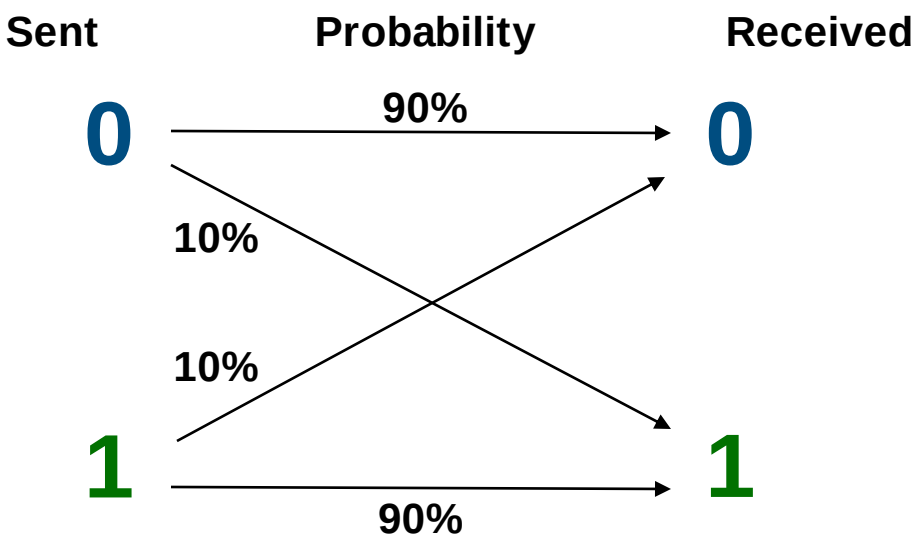


Channel

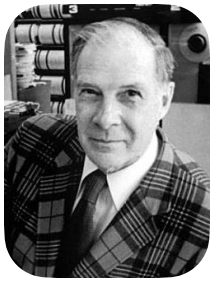


Can I protect my message?

Channel

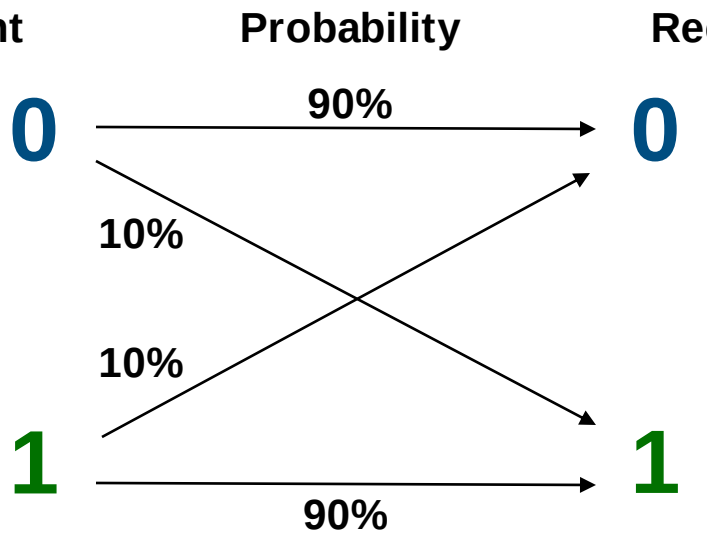


Can I protect my message?

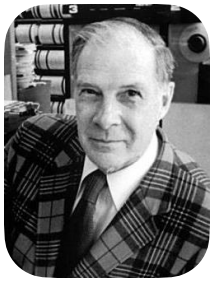


Hamming 1950

Channel



Can I protect my message?



Hamming 1950

4 bits *of information* to be sent

$i_1 i_2 i_3 i_4$

0 1 1 0

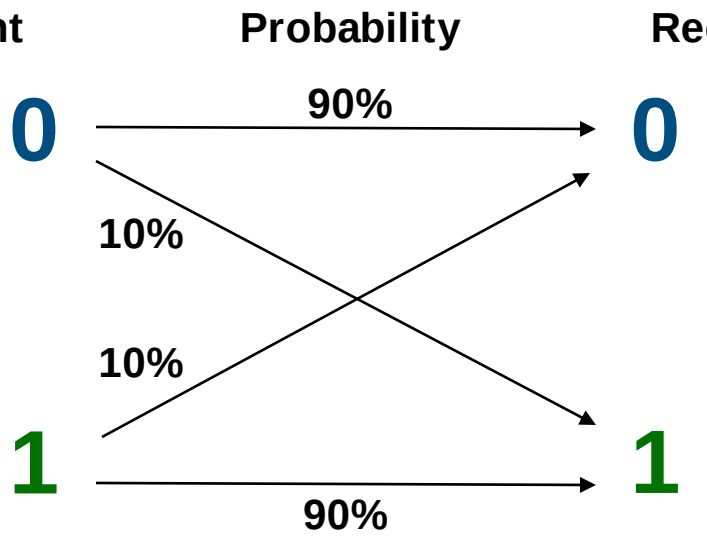
encoding

7 *code* bits sent over the channel

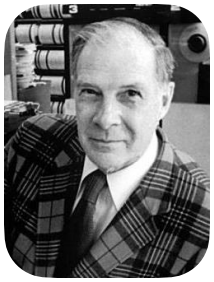
$i_1 i_2 i_3 i_4 p_1 p_2 p_3$

0 1 1 0 1 1 0

Channel



Can I protect my message?



Hamming 1950

4 bits *of information* to be sent

$i_1 i_2 i_3 i_4$

0 1 1 0

encoding

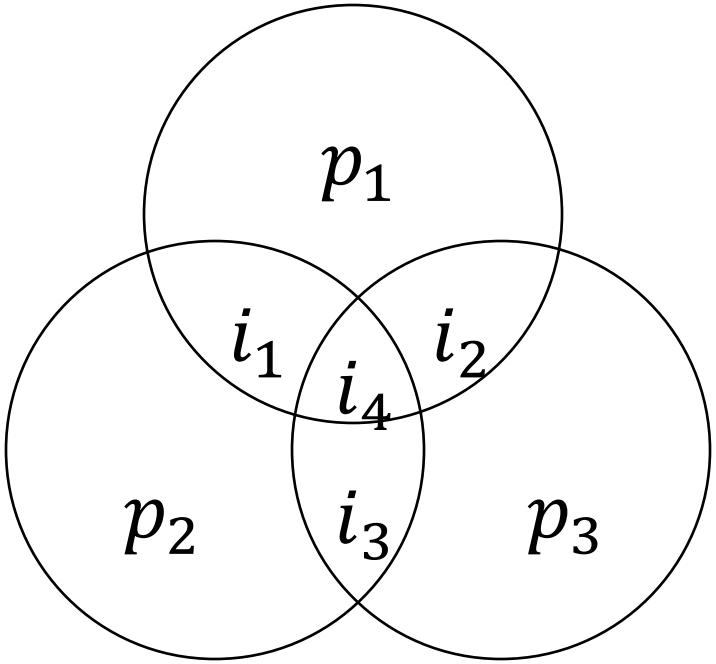
7 *code* bits sent over the channel

$i_1 i_2 i_3 i_4 p_1 p_2 p_3$

0 1 1 0 1 1 0

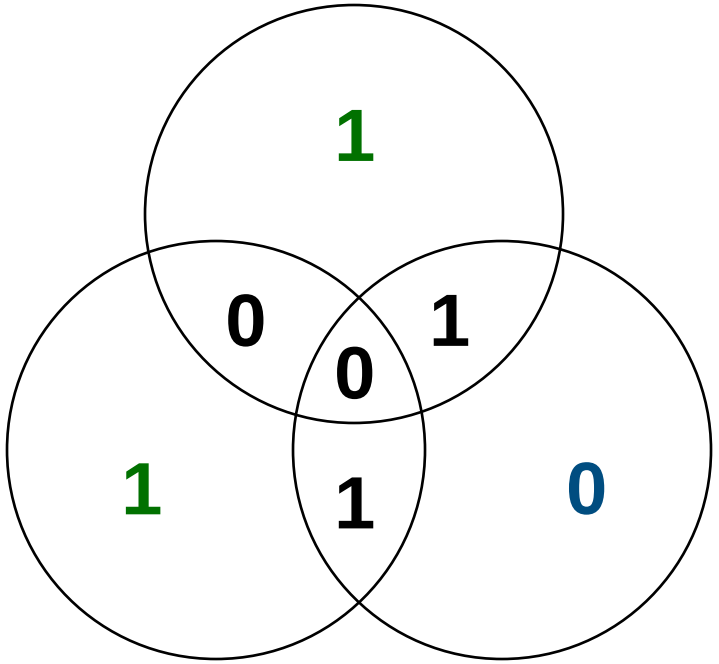
parity bits … redundancy

Error protection



Information bits

0 1 1 0

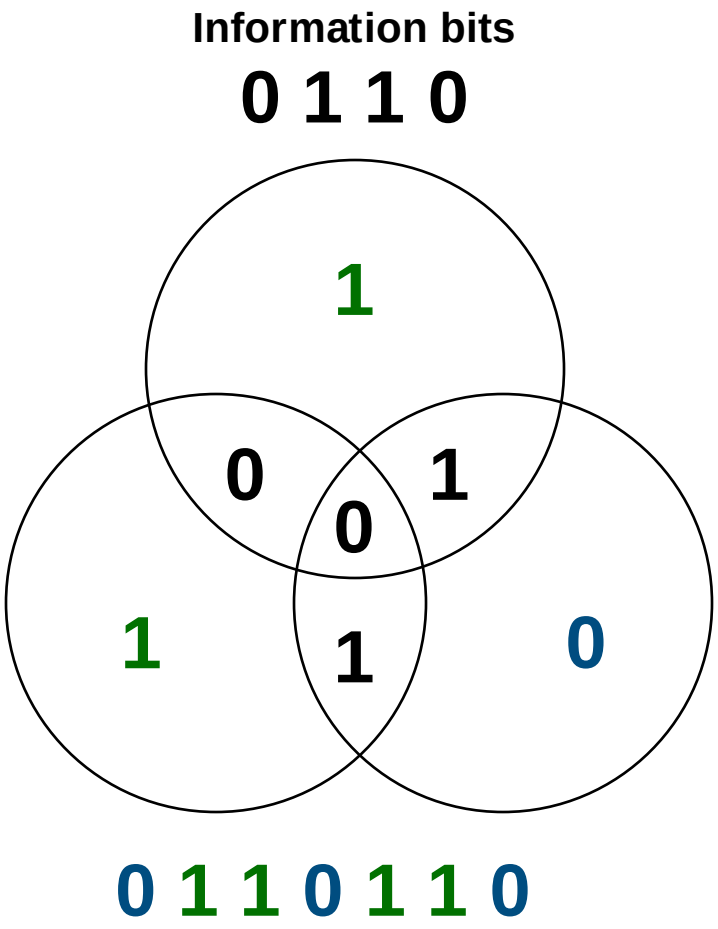


0 1 1 0 1 1 0

Code bits

Rate=4/7

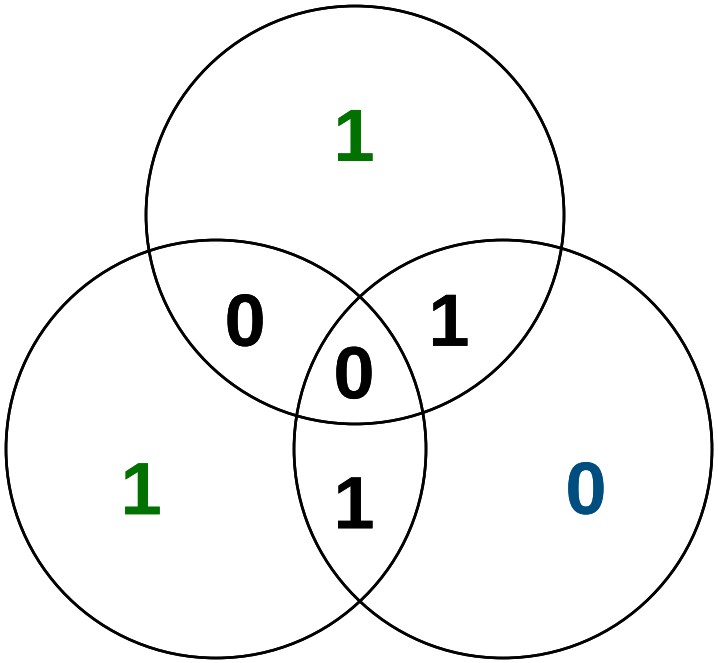
It can correct one error!



It can correct one error!

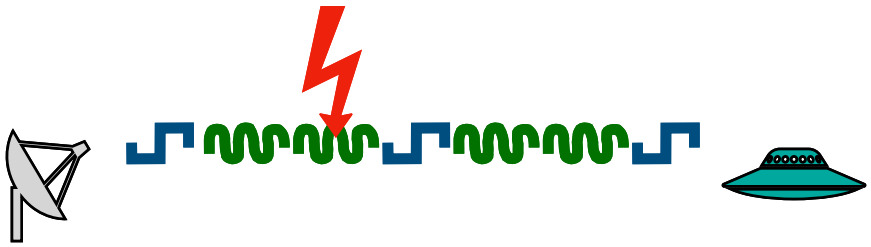
Information bits

0 1 1 0



0 1 1 0 1 1 0

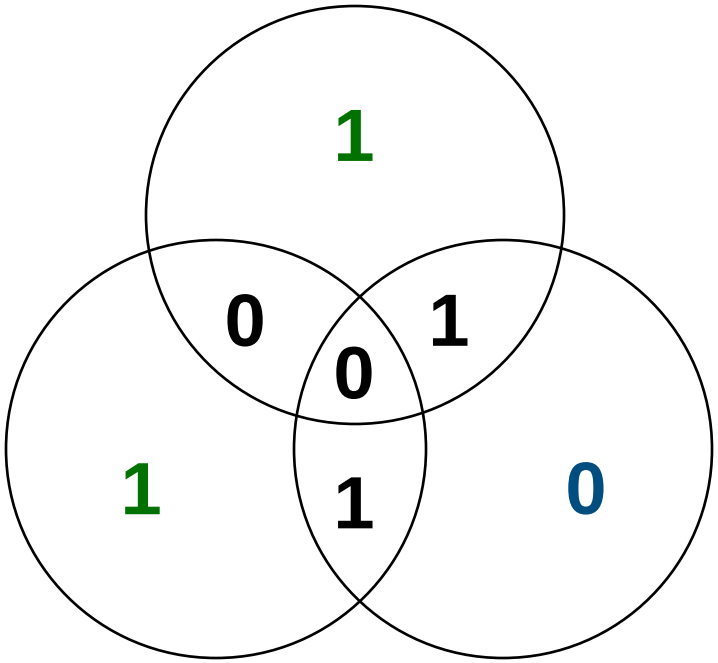
Code bits sent



It can correct one error!

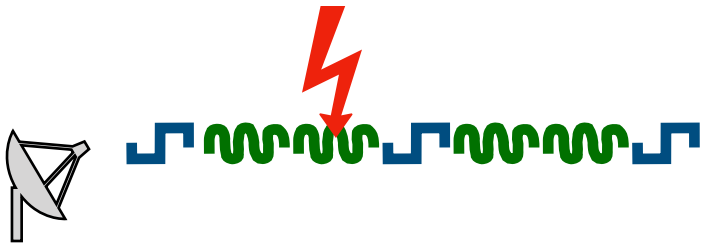
Information bits

0 1 1 0



0 1 1 0 1 1 0

Code bits sent



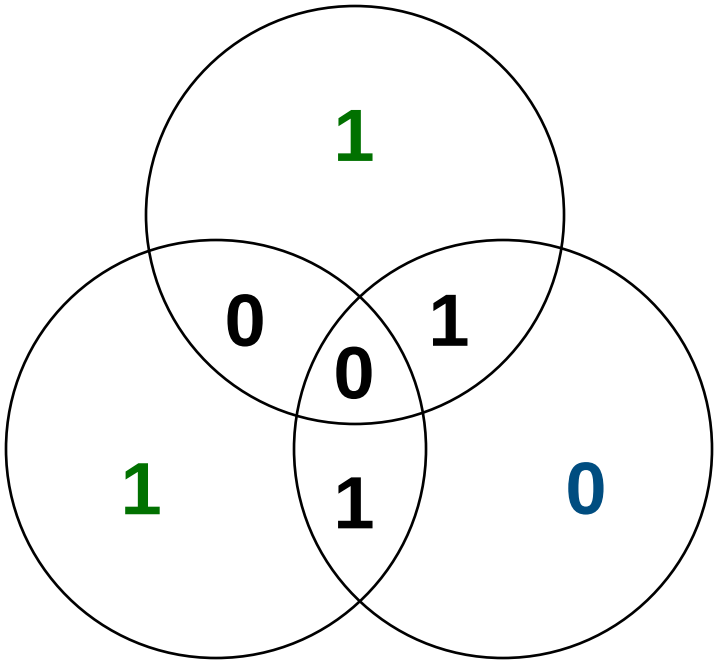
0 1 0 0 1 1 0

Code bits received

It can correct one error!

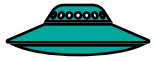
Information bits

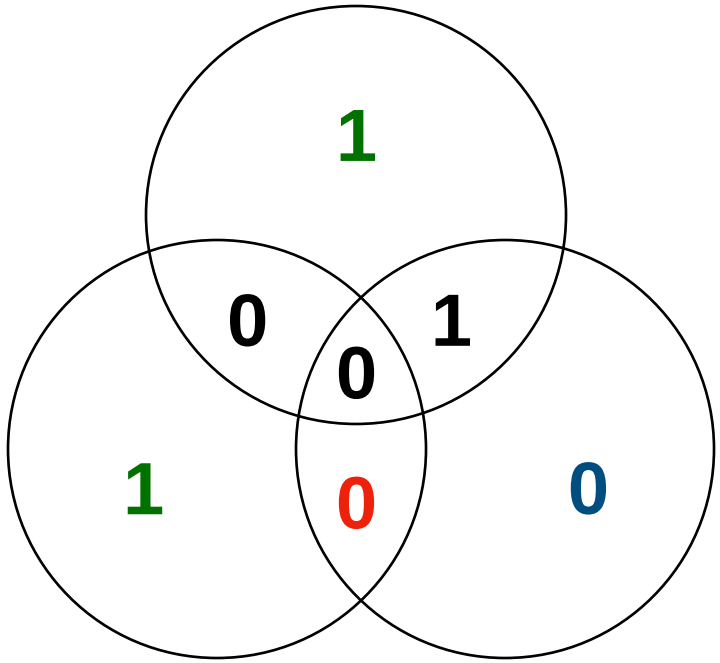
0 1 1 0



0 1 1 0 1 1 0

Code bits sent





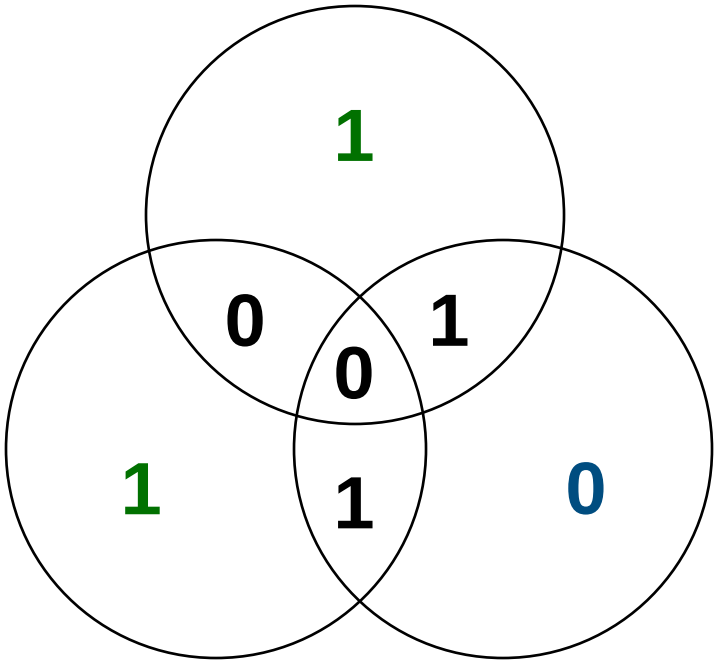
0 1 0 0 1 1 0

Code bits received

It can correct one error!

Information bits

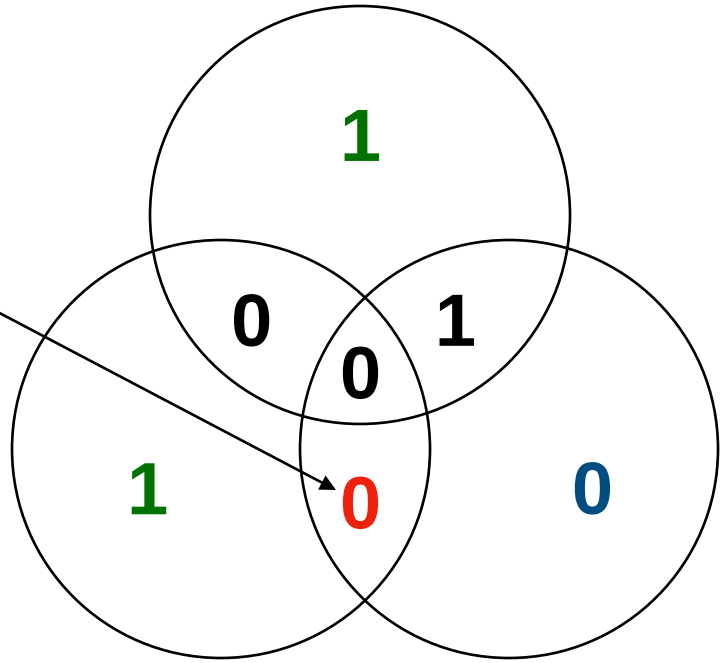
0 1 1 0



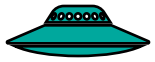
0 1 1 0 1 1 0

Code bits sent

Error!



0 1 0 0 1 1 0

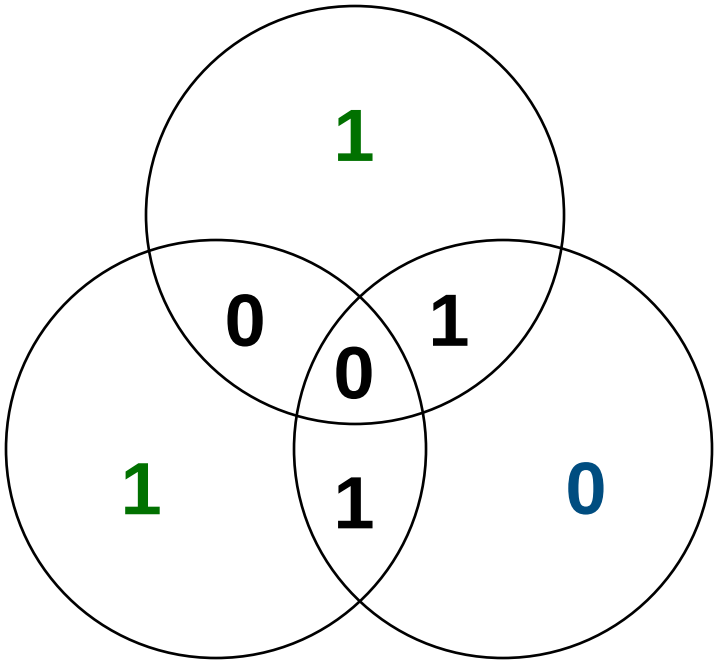


Code bits received

It can correct one error!

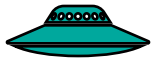
Information bits

0 1 1 0



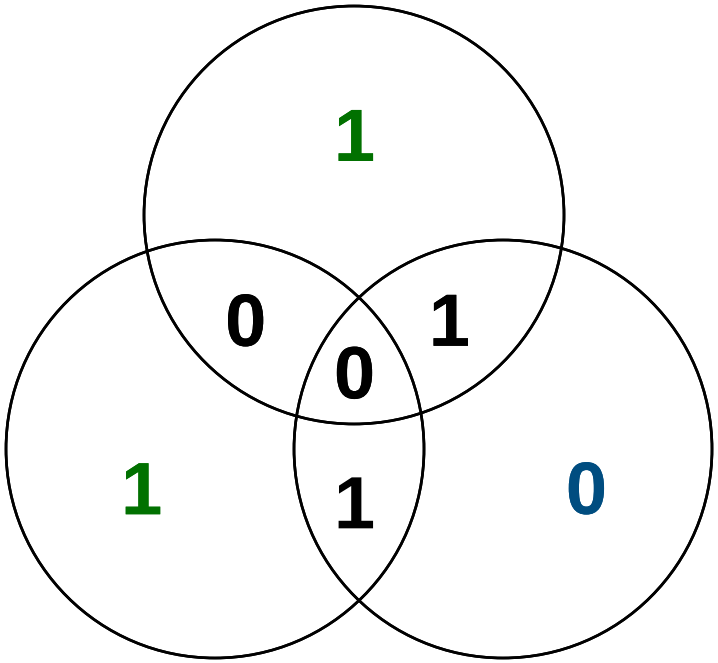
0 1 1 0 1 1 0

Code bits sent



Information bits

0 1 1 0



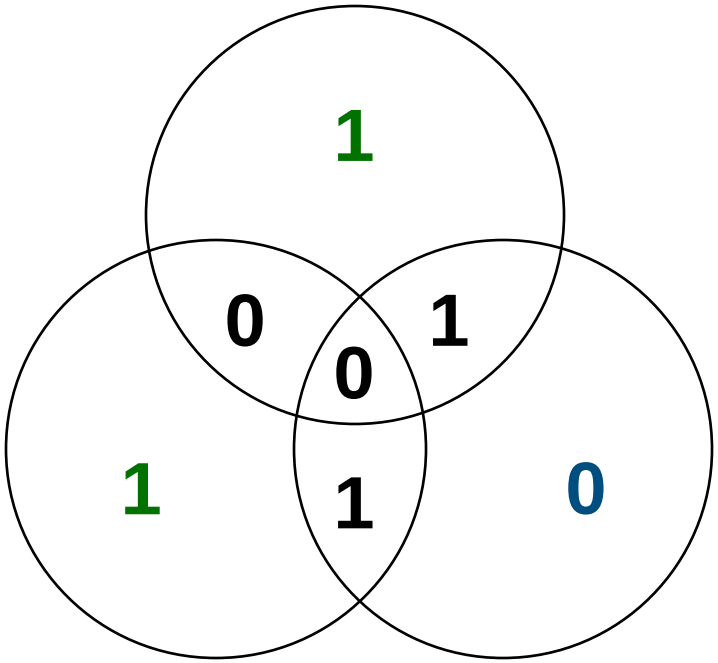
0 1 0 0 1 1 0

Code bits received

It cannot correct two errors!

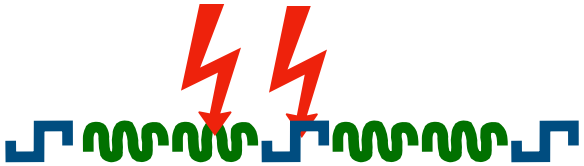
Information bits

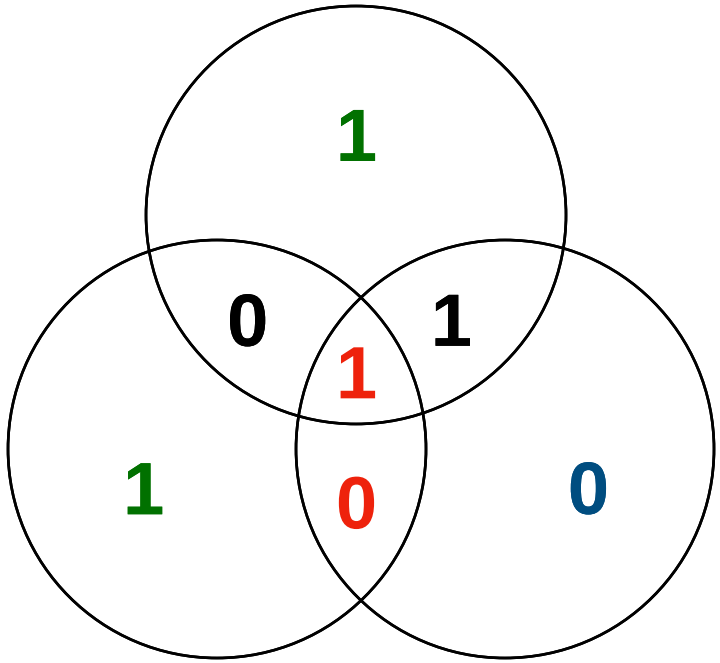
0 1 1 0



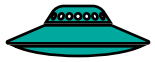
0 1 1 0 1 1 0

Code bits sent





0 1 0 1 1 1 0

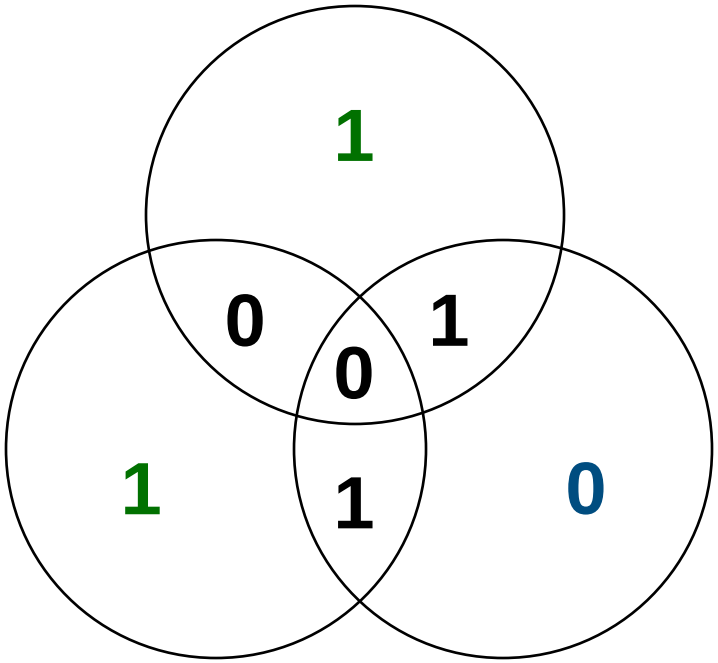


Code bits received

It cannot correct two errors!

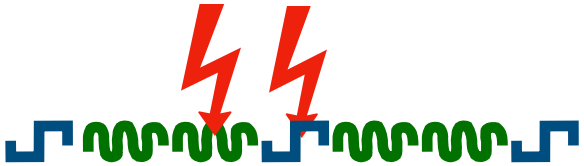
Information bits

0 1 1 0

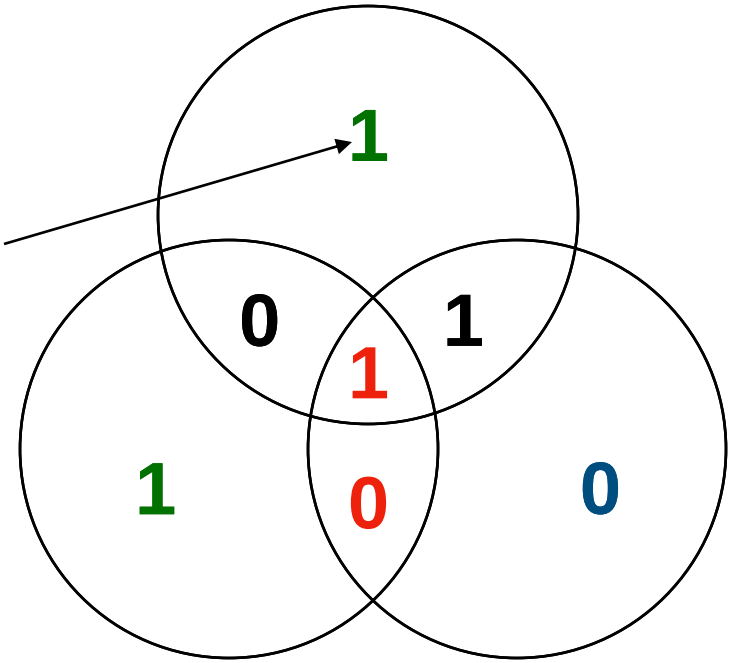


0 1 1 0 1 1 0

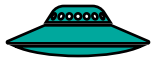
Code bits sent



Error!



0 1 0 1 1 1 0

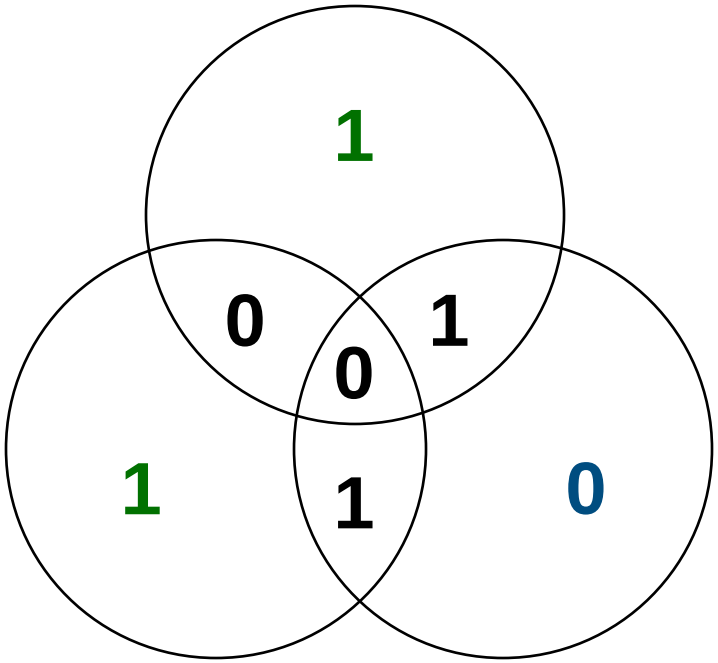


Code bits received

It cannot correct two errors!

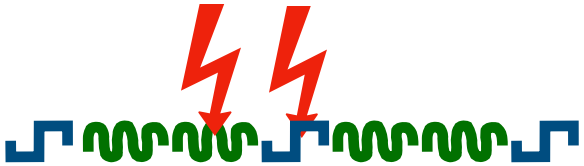
Information bits

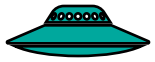
0 1 1 0



0 1 1 0 1 1 0

Code bits sent

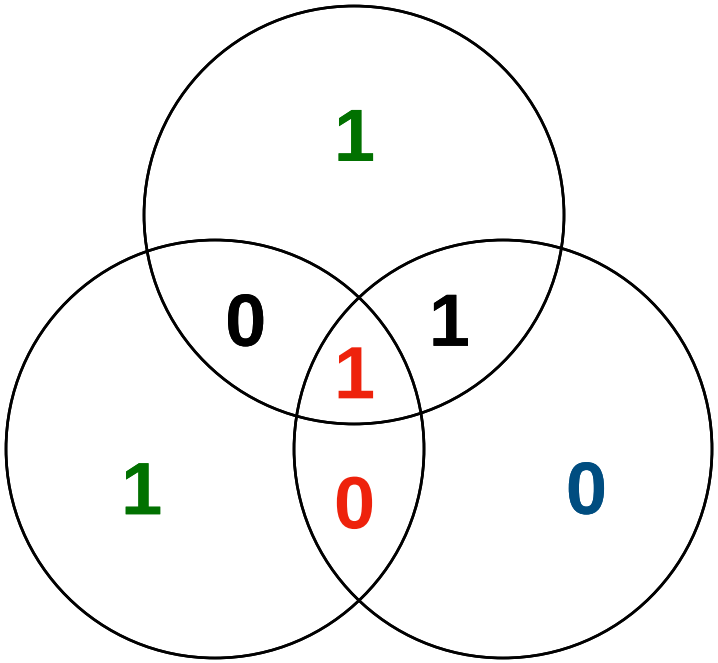




Code bits received

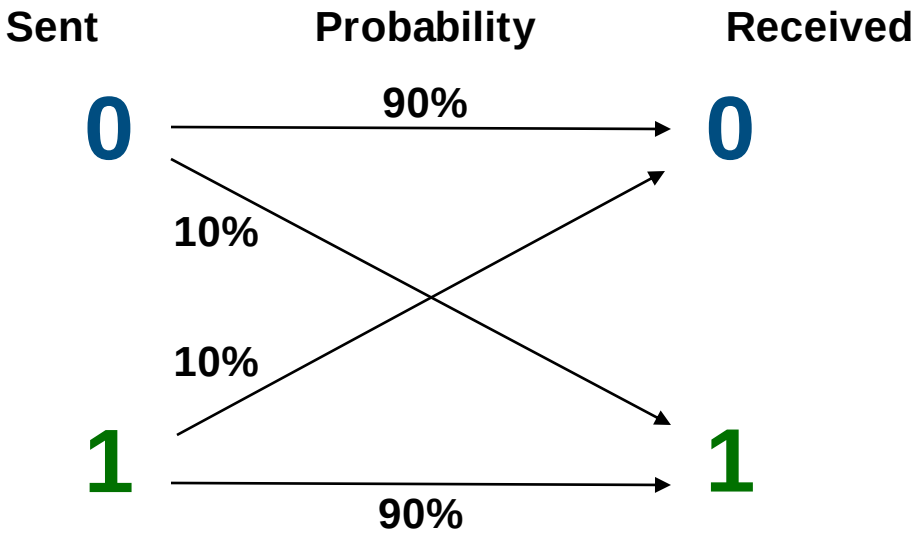
Information bits

0 1 0 1



0 1 0 1 1 1 0

Channel

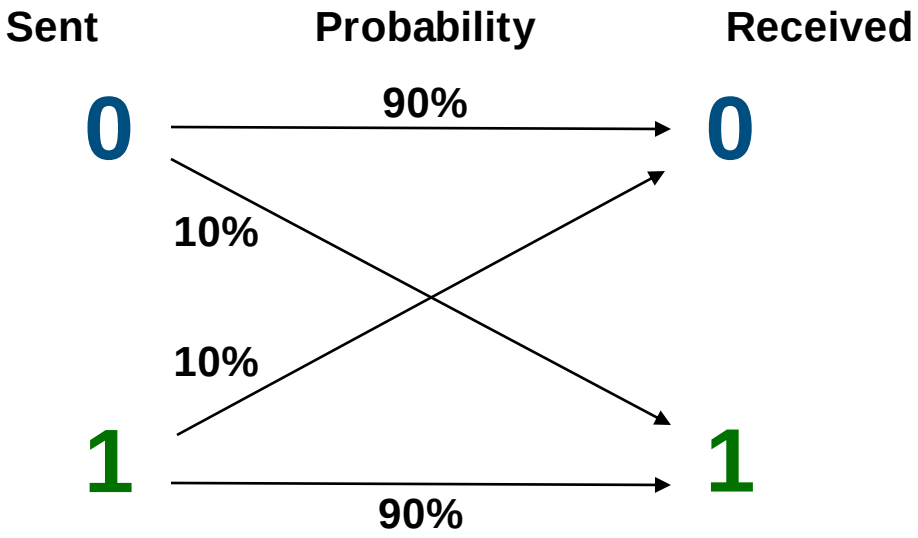


Rate=4/7

≈0.5714 bits/pulse

I am left with nearly 6% of errors

Channel



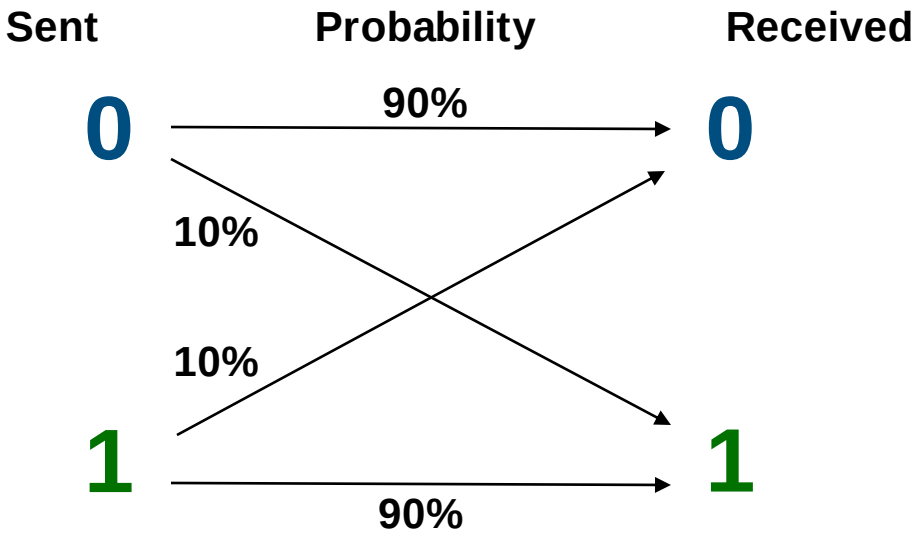
Rate=4/7

≈0.5714 bits/pulse

I am left with nearly 6% of errors

Is it possible to do better?

Channel

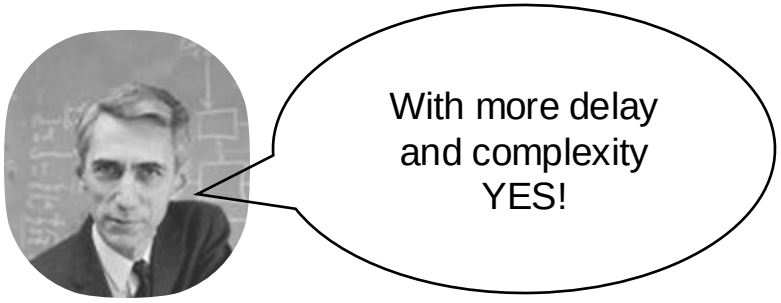


Rate=4/7

≈0.5714 bits/pulse

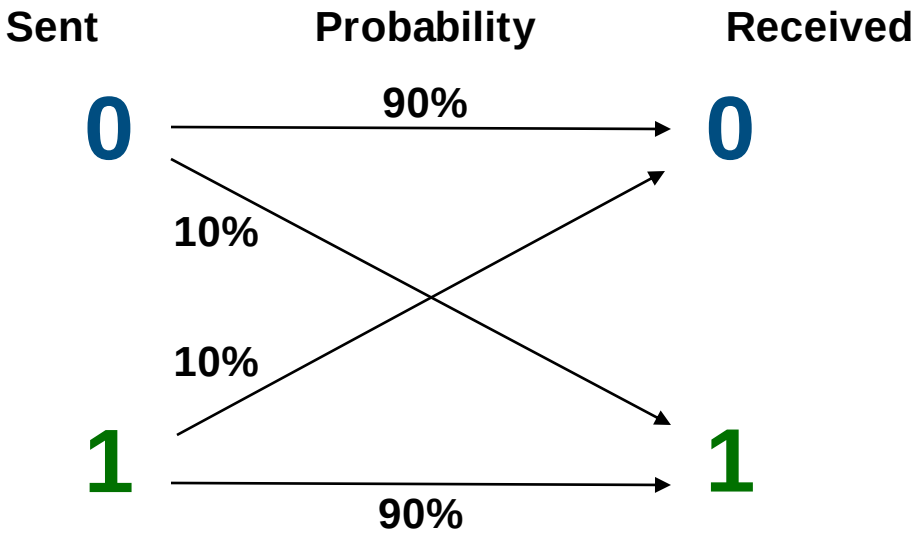
I am left with nearly 6% of errors

Is it possible to do better?



Shannon 1948

Channel

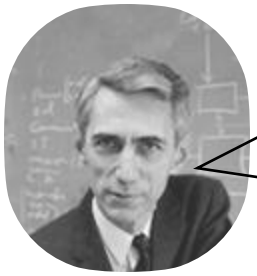


Rate=4/7

≈0.5714 bits/pulse

I am left with nearly 6% of errors

Is it possible to do better?



With more delay and complexity YES!

Shannon 1948

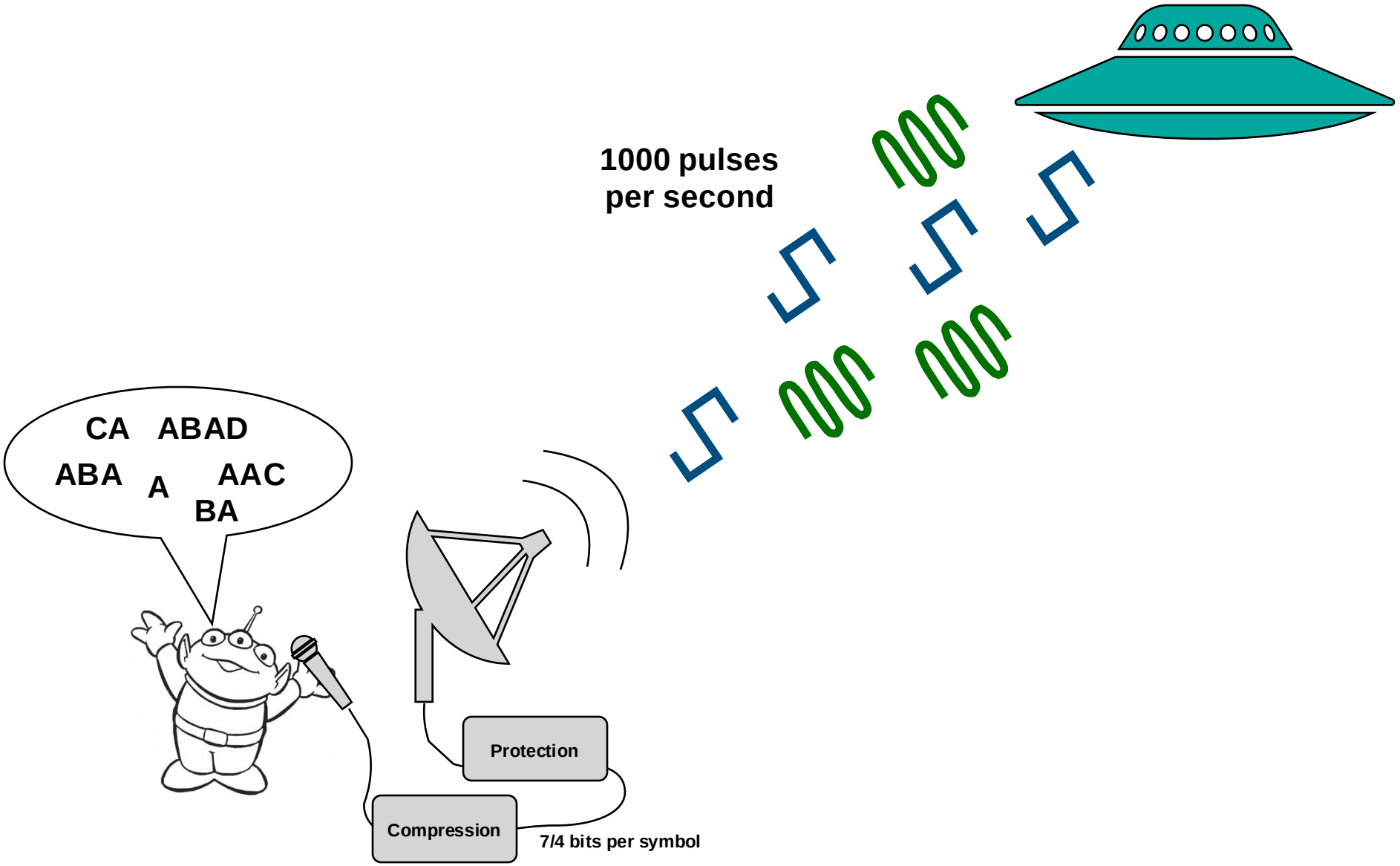
Channel Capacity

$$C = 1 + \frac{1}{10}\log_2 10 + \frac{9}{10}\log_2 \frac{10}{9}$$

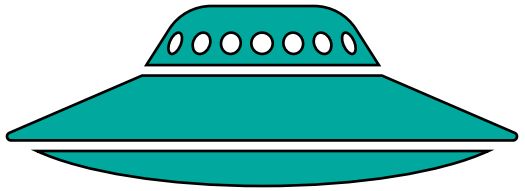
$$\approx 0.531 \, bits/pulse$$

≈ error free!!

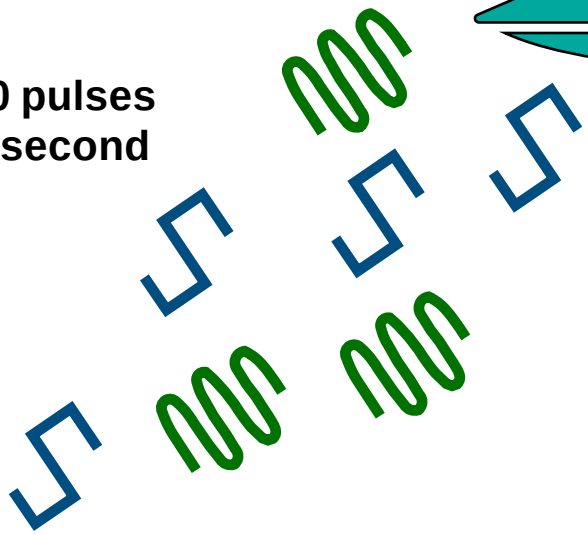
Transmission of Information

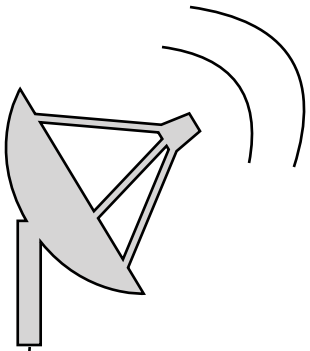


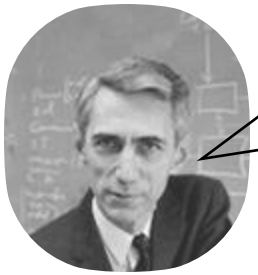
Transmission of Information



1000 pulses per second







You can send **reliably**

$$1000 \cdot 0.531 \cdot \frac{4}{7}$$

$$\approx 300$$

symbols per second

Protection

Compression

7/4 bits per symbol

Overview

- **Information Measures**

  Entropy, Mutual Information, Entropy Rate, their properties…

- **Source Coding**

  How much can we compress a source? What is the optimal code? Some practical examples…

- **Channel Coding**

  How much information can we send through a channel? … Theory, no practical examples here