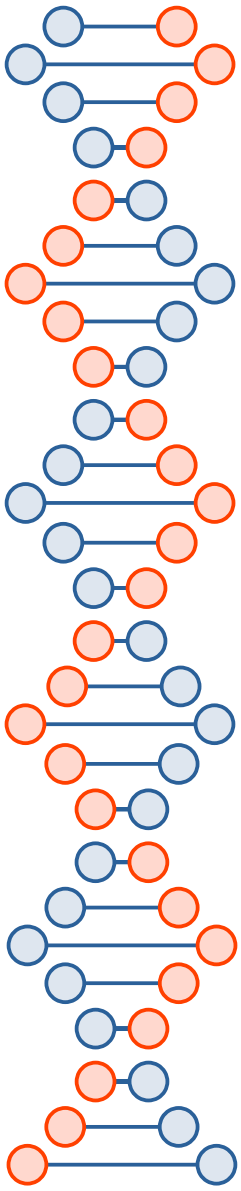
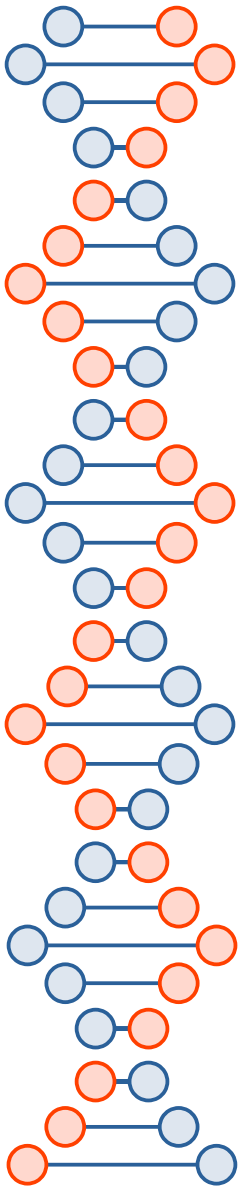


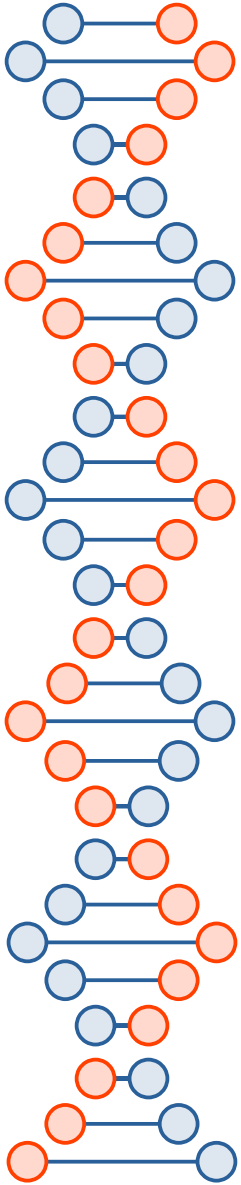
Symmetric Cryptography (Through the 1980s or so...)

CSE 468 Fall 2025
jedimaestro@asu.edu



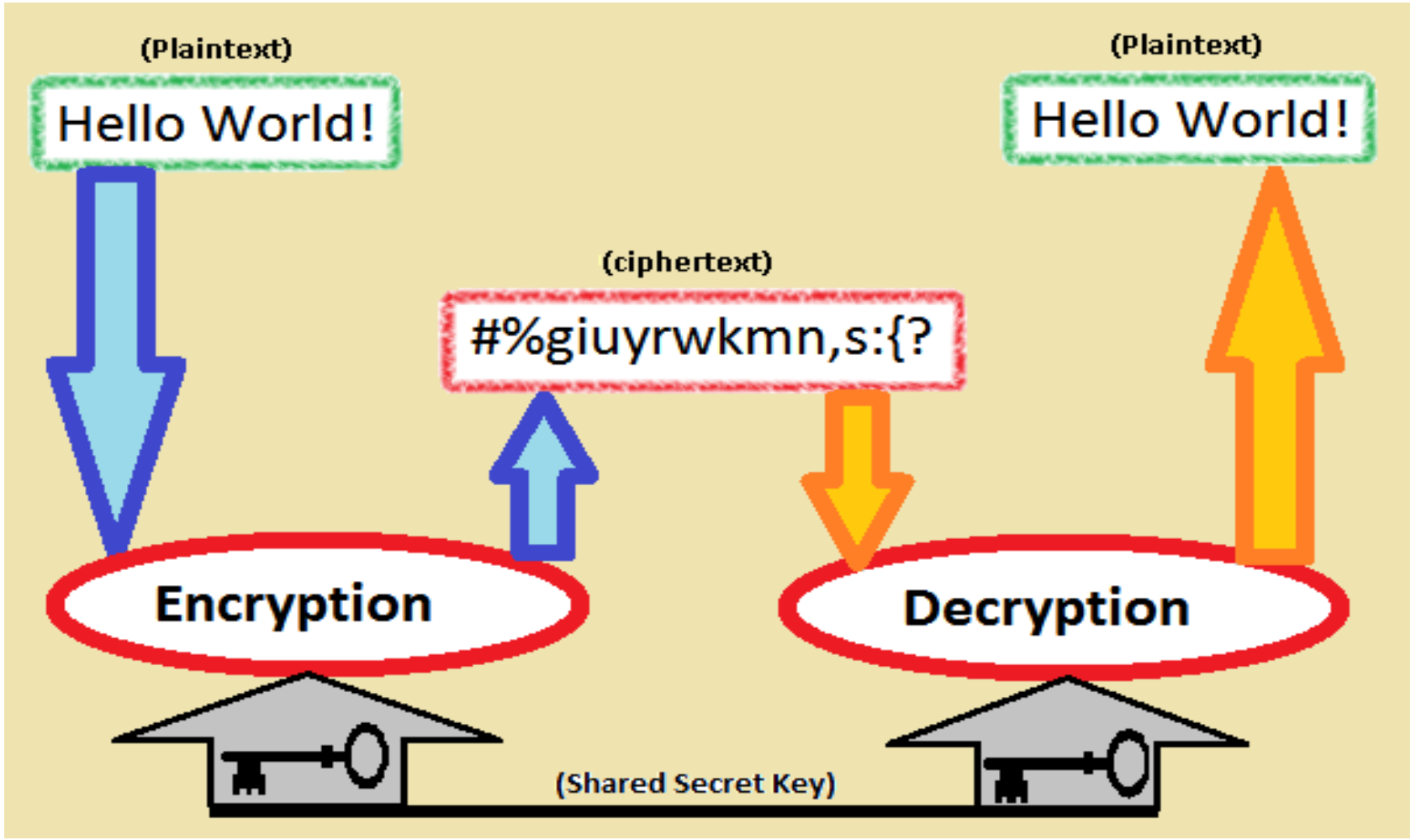
Basics of crypto (review)...

- Symmetric encryption
  - Assumes two parties wishing to communicate already have a shared secret
- Asymmetric encryption
  - Makes different assumptions (*e.g.*, that everybody knows the public key or that the eavesdropper is passive)
  - Quantum computers break current algorithms that are used in practice
- Secure hash functions and message authentication

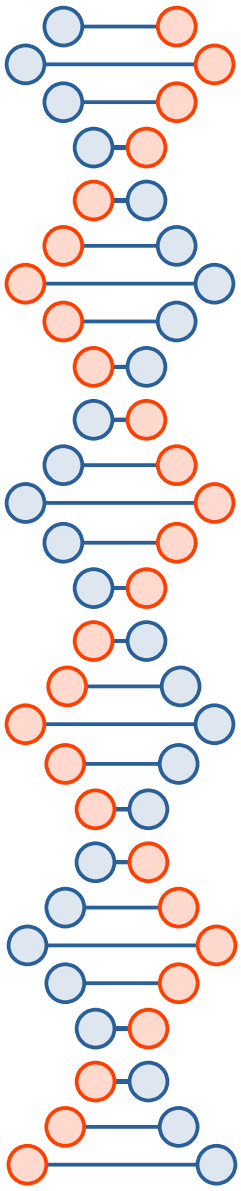


Symmetric Crypto

- Confidentiality
- Integrity
- ~~Availability~~
- Authentication
- ~~Non-repudiation~~
- ~~A way to distribute the shared secret keys~~

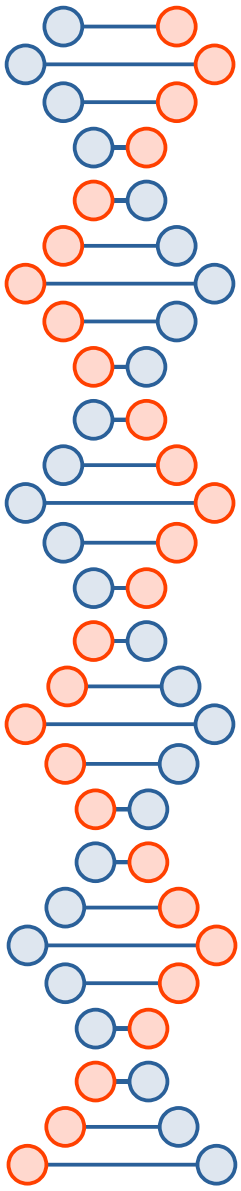


Source: Wikipedia



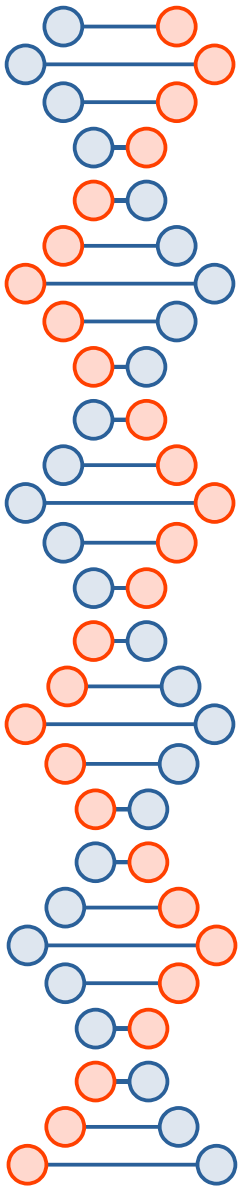
Terminology

- Plaintext – before encryption, easy to read
- Ciphertext – after encryption, hopefully indecipherable without the key
- Key – the shared secret, typically just bits that were generated with a high entropy process



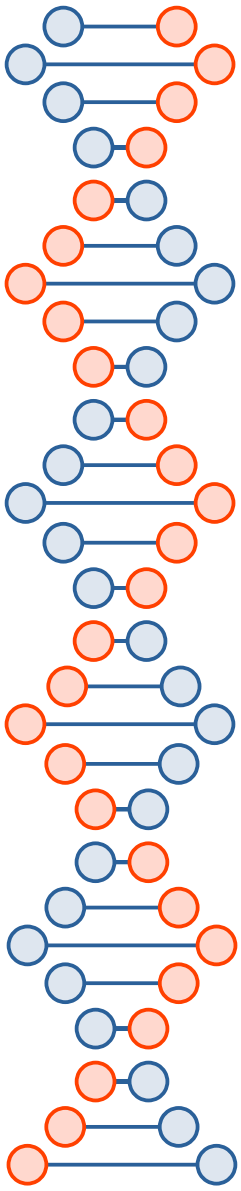
Review on your own...

- Caesar Cipher
- Vigenere Cipher and related attacks



Modern symmetric crypto

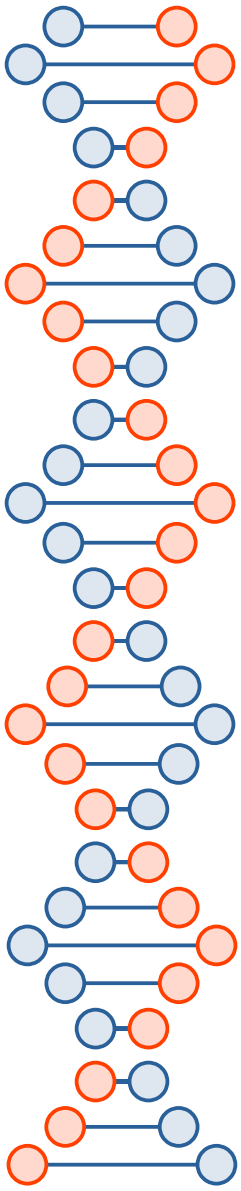
- Mostly:
  - Substitution
  - Permutation (or transposition)
  - XOR



Substitution

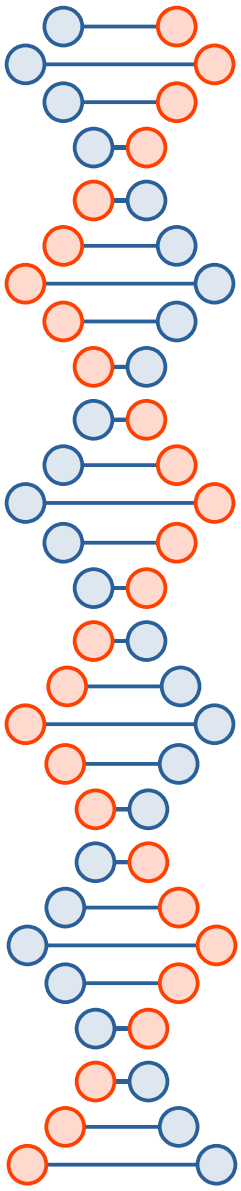
HELLO WORLD

TNWWX DXPWE



Permutation

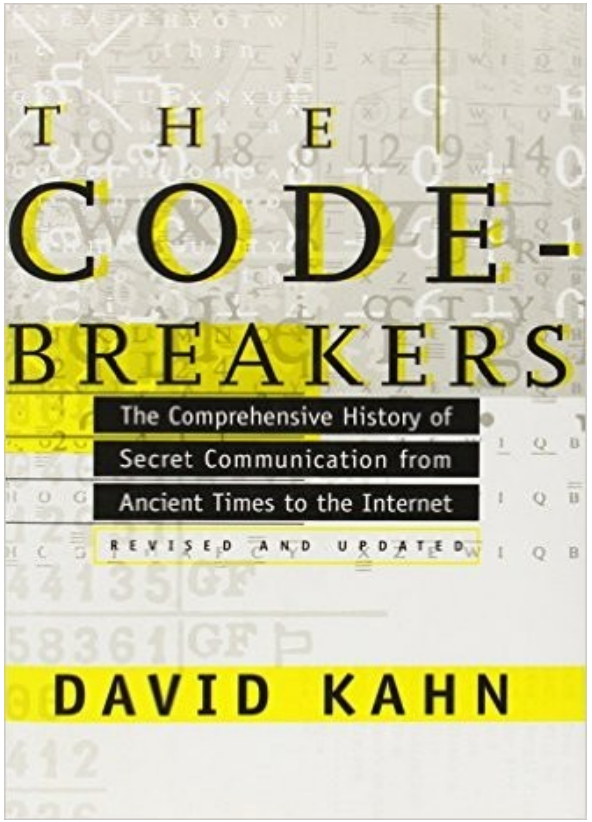
ABCD	ABDC	ACBD	ACDB	ADBC	ADCB
BACD	BADC	BCAD	BCDA	BDAC	BDCA
CABD	CADB	CBAD	CBDA	CDAB	CDBA
DABC	DACB	DBAC	DBCA	DCAB	DCBA

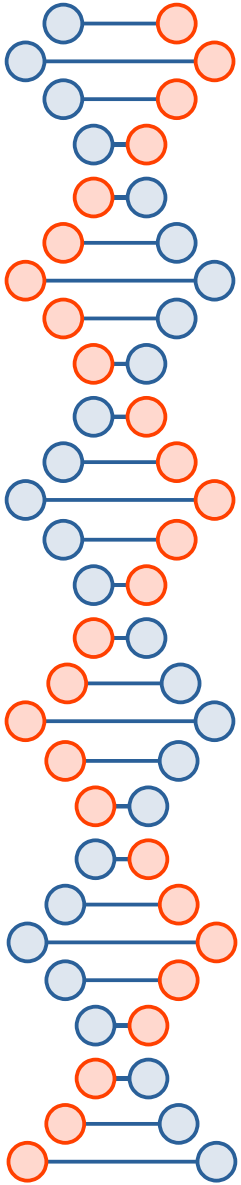


Bitwise XOR

$$
\begin{aligned}
&00101010_{\mathrm{b}} \\
\oplus\,&10000110_{\mathrm{b}} \\
=\,&10101100_{\mathrm{b}}
\end{aligned}
$$

2000+ years of history...



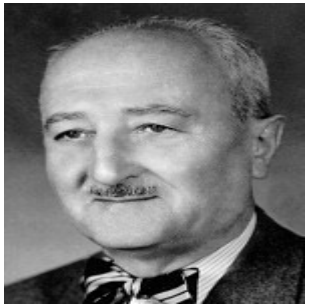


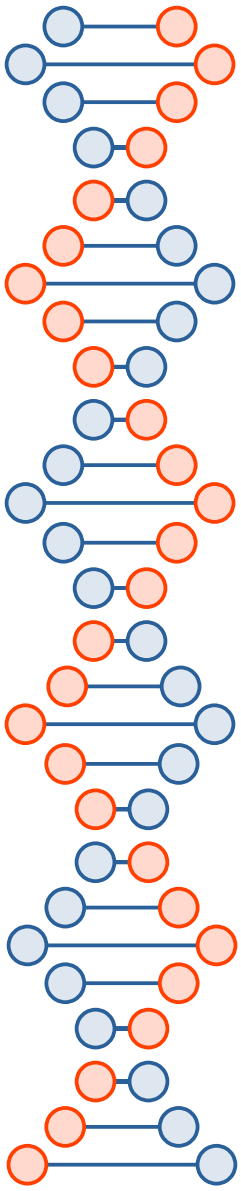
Symmetric encryption over time

- Handwritten notes, *etc.* for centuries
  - Typically the algorithm was secret
- 1883 … Kerckhoff's rules
  - Now we know the key should be the only secret
- 1975 … DES
  - Efficient in hardware, not in software
- 2001 … AES
  - Efficient in software, and lots of different kinds of hardware

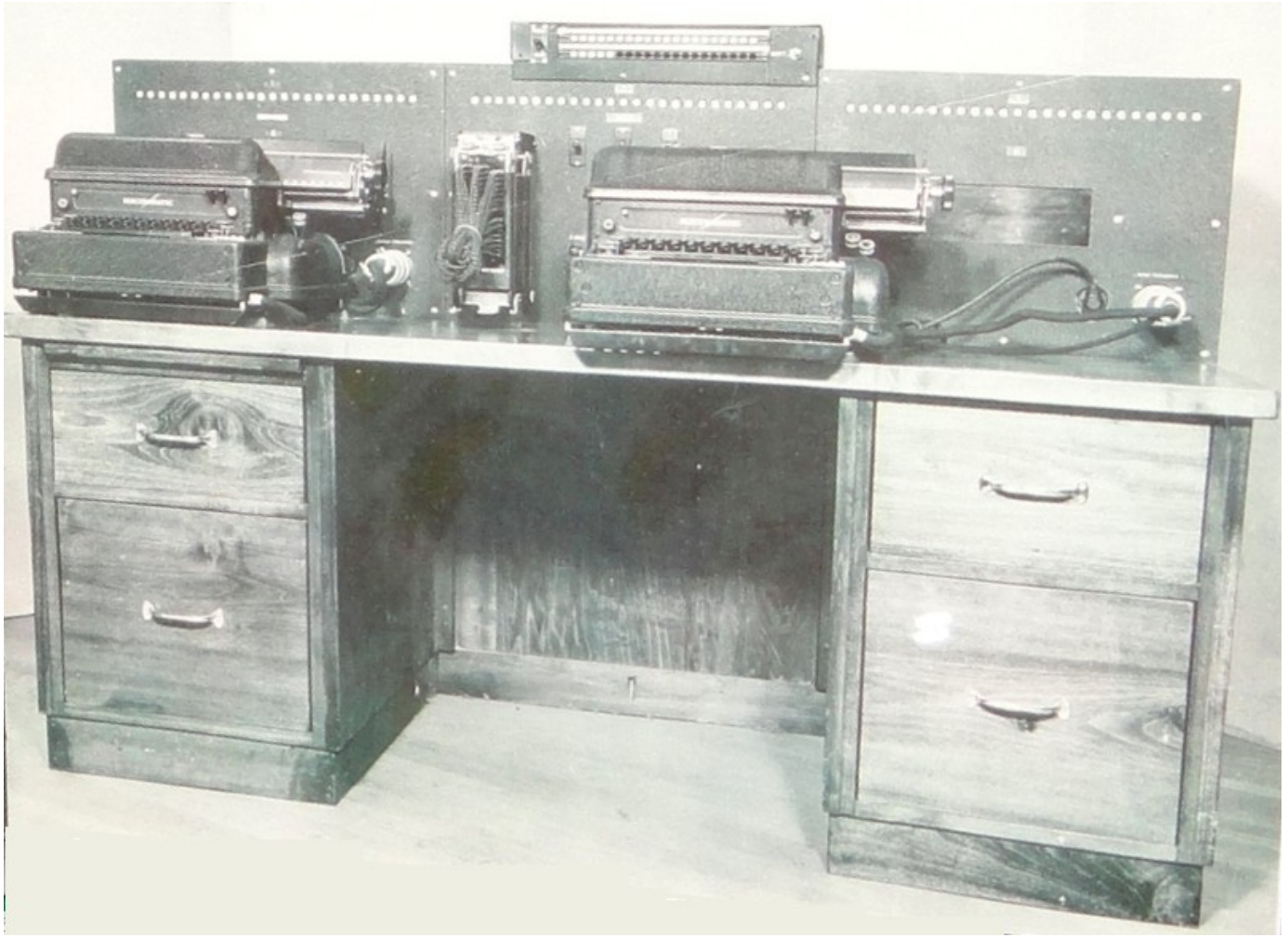
William and Elizabeth Friedman

- Met while analyzing Shakespeare ciphers at Riverbank Laboratories (“William Friedman wrote Shakespeare's plays”)
- Elizabeth solved ciphers of alcohol and drug smugglers, then German ambassadors in South America (three enigma machines)
- William led a team that solved PURPLE, conceived CryptoAG scheme





Substitution and/or permutation...

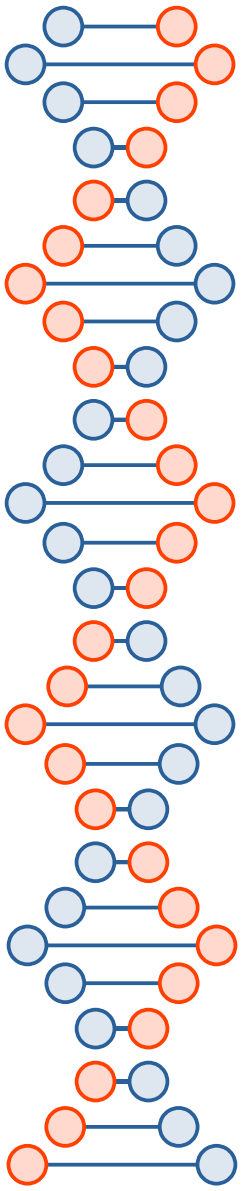


https://en.wikipedia.org/wiki/Type_B_Cipher_Machine#/media/File:Purple_cipher_machine_analog_bw_photo_NCM.jpg

https://en.wikipedia.org/wiki/Enigma_machine#/media/File:Enigma_(crittografia)_-_Museo_scienza_e_tecnologia_Milano.jpg

Zodiac cipher

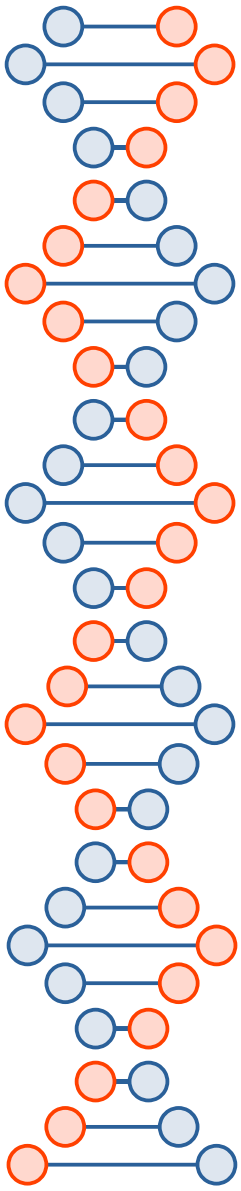
Image from wikia



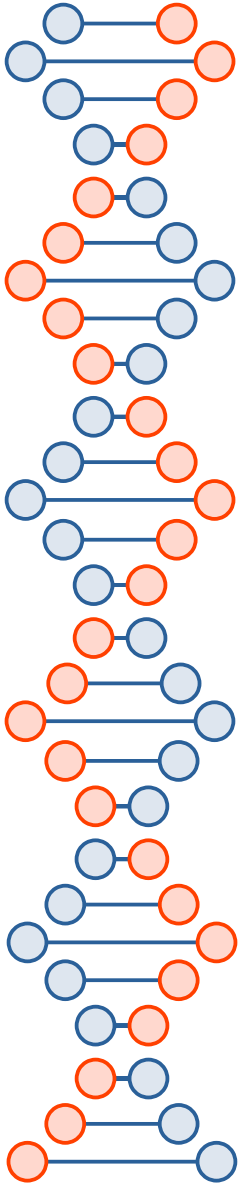
How to crack?

- Frequency analysis
- “The most common letter in the english language is e”
- “Gsv nlhg xlnnlm ovggvi rm gsv vmtorhs ozmtfztv rh v”
  - 7 v’s, 4 g’s, 4 m’s, 3 s’s, 3 n’s, 3 l’s …

- Guess what quantum computers are good at?

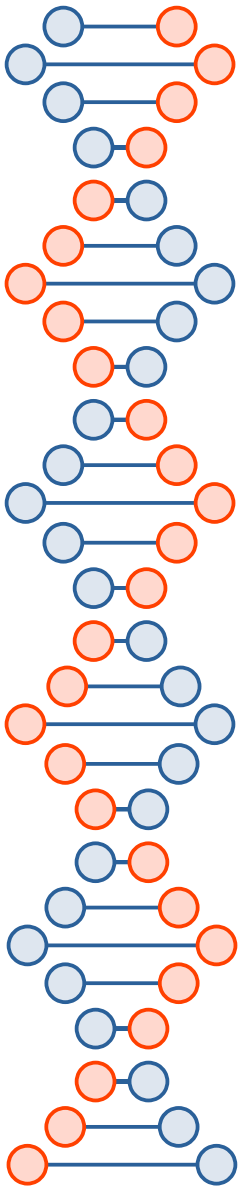


XOR...



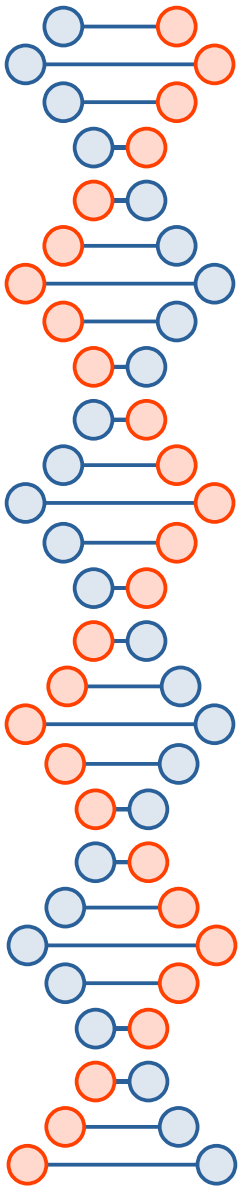
Bitwise XOR as a cipher itself

- Typically used by malware, 8 or 32 bits
  - WEP attack uses these properties
- (B xor K) xor K = B
- (A xor K) xor (B xor K) = A xor B
- (0 xor K) = K
- (K xor K) = 0
- Frequency analysis or brute force



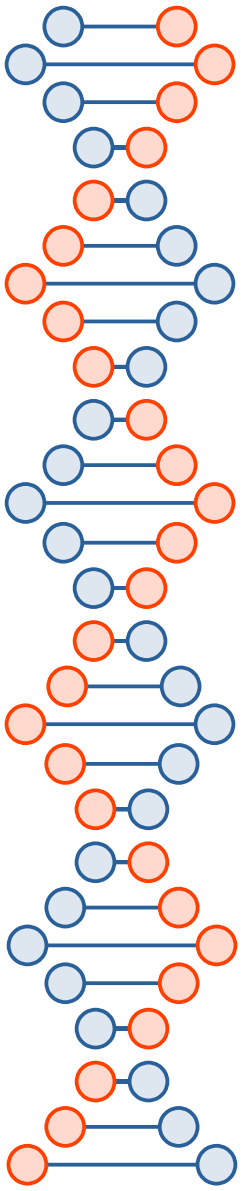
One-time pad

- *E.g.*, an XOR cipher or Caesar cipher where the key has good randomness and is as long as the plaintext
  - And never gets reused
- Most codes made by the NSA through the 1980s were one-time pads
  - What if it's not practical to share enough key material beforehand, *e.g.*, on the Internet?



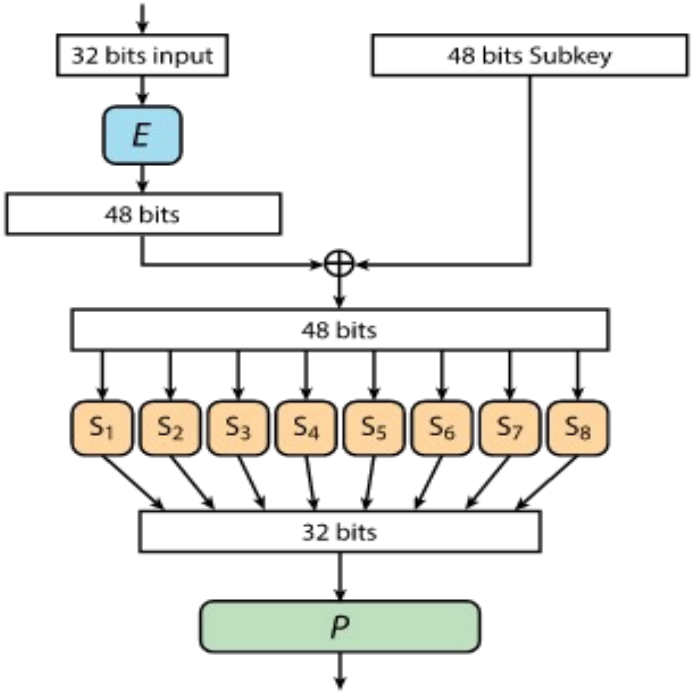
Preview:

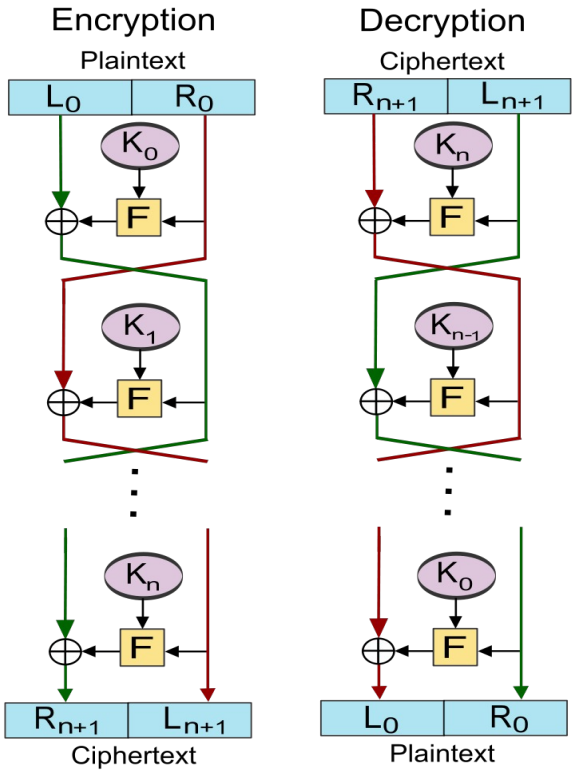
CNOT, the quantum version of XOR, will defy your concept of time and causality and we'll see that the outputs sometimes affect the inputs.

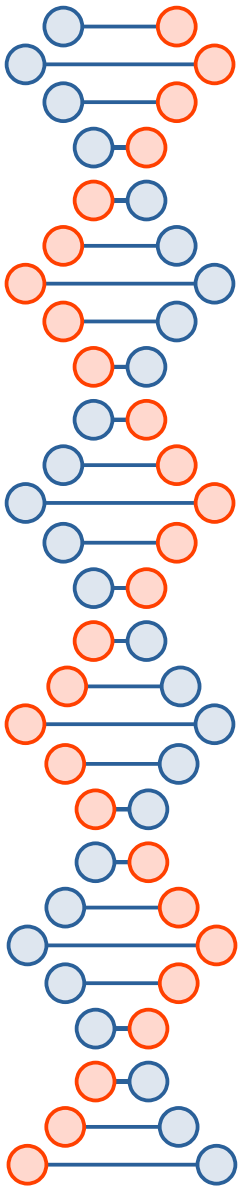


Now, let’s look at the first really good (in Jed’s opinion) symmetric cipher...

1977 - DES (16 rounds, 64-bit blocks, 56-bit key)



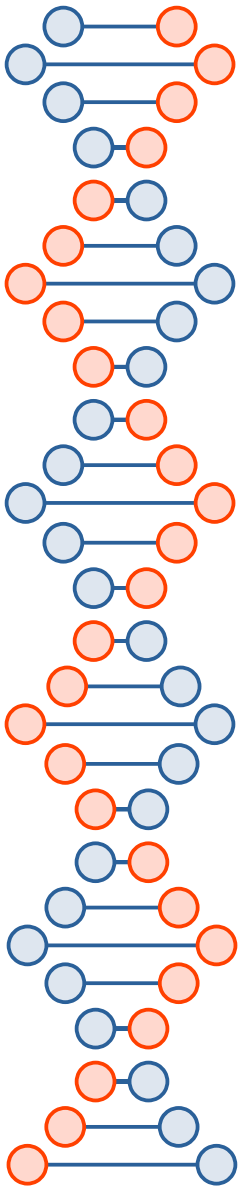




DES S-boxes

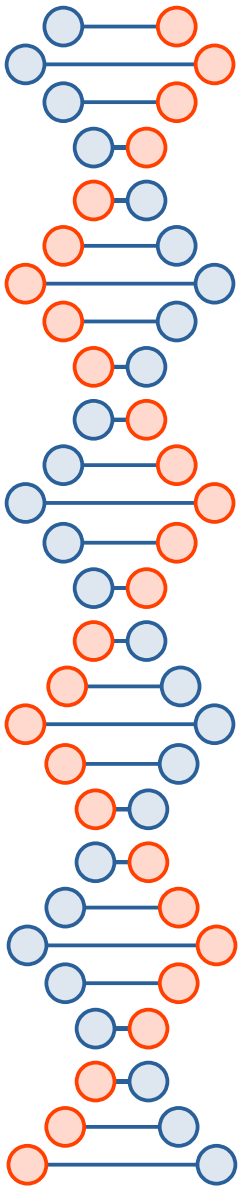
- 6 bits becomes 4 bits
- Values somewhat arbitrary
  - IBM proposed some, NSA replaced with others
    - Linear and differential cryptanalysis (unknown in the open literature at the time) were probably the reasons

שורה	מס' עמודה															
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
	S_1															
0	14	4	13	1	2	15	11	8	3	10	6	12	5	9	0	7
1	0	15	7	3	14	2	13	1	10	6	12	11	9	5	3	8
2	4	1	14	8	13	6	2	11	15	12	9	7	13	10	5	0
3	15	12	8	2	4	9	1	7	5	11	3	14	10	0	6	13
	S_2															
0	15	1	8	14	6	11	3	4	9	7	2	13	12	0	5	10
1	3	13	4	7	15	2	8	14	12	0	1	10	6	9	11	5
2	0	14	7	11	10	4	13	1	5	8	12	6	9	3	2	15
3	13	8	10	1	3	15	4	2	11	6	7	12	0	5	14	9
	S_3															
0	10	0	9	14	6	3	15	5	1	13	12	7	11	4	2	8
1	13	7	0	9	3	4	6	10	2	8	5	14	12	11	15	1
2	13	6	4	9	8	15	3	0	11	1	2	12	5	10	14	7
3	1	10	13	0	6	9	8	7	4	15	14	3	11	5	2	12
	S_4															
0	7	13	14	3	0	6	9	10	1	2	8	5	11	12	4	15
1	13	8	11	5	6	15	0	3	4	7	2	12	1	10	14	9
2	10	6	9	0	12	11	7	13	15	1	3	14	5	2	8	4
3	3	15	0	6	10	1	13	8	9	4	5	11	12	7	2	14
	S_5															
0	2	12	4	1	7	10	11	6	8	5	3	15	13	0	14	9
1	14	11	2	12	4	7	13	1	5	0	15	10	3	9	8	6
2	4	2	1	11	10	13	7	8	15	9	12	5	6	3	0	14
3	11	8	12	7	1	14	2	13	6	15	0	9	10	4	5	3
	S_6															
0	12	1	10	15	9	2	6	8	0	13	3	4	14	7	5	11
1	10	15	4	2	7	12	9	5	6	1	13	14	0	11	3	8
2	9	14	15	5	2	8	12	3	7	0	4	10	1	13	11	6
3	4	3	2	12	9	5	15	10	11	14	1	7	6	0	8	13
	S_7															
0	4	11	2	14	15	0	8	13	3	12	9	7	5	10	6	1
1	13	0	11	7	4	9	1	10	14	3	5	12	2	15	8	6
2	1	4	11	13	12	3	7	14	10	15	6	8	0	5	9	2
3	6	11	13	8	1	4	10	7	9	5	0	15	14	2	3	12
	S_8															
0	13	2	8	4	6	15	11	1	10	9	3	14	5	0	12	7
1	1	15	13	8	10	3	7	4	12	5	6	11	0	14	9	2
2	7	11	4	1	9	12	14	2	0	6	10	13	15	3	5	8
3	2	1	14	7	4	10	8	13	15	12	9	0	3	5	6	11



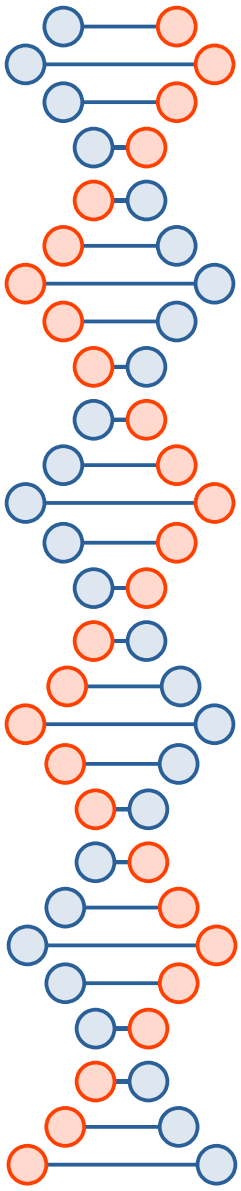
Importance of substitution

- XOR and permutation are linear functions
  - Solve for the key given plaintext and ciphertext?
- Bit differences in inputs are not changed at all by permuting bits
- XOR also preserves differences in bits



Different approaches (preview)

- DES simply tried to thwart these two specific types of attack (linear and differential) by carefully choosing the S boxes and letting them destroy information about the input (okay because of Feistel structure)
- Blowfish used π as the S boxes
- *Preview:* AES is going to do something very clever, that is invertible (no need for the Feistel structure, so fewer rounds) but still thwarts linear and differential cryptanalysis.



But where do the keys come from?

What makes DES, AES, or other “good” ciphers more secure than other block ciphers?

What makes a good symmetric crypto algorithm?

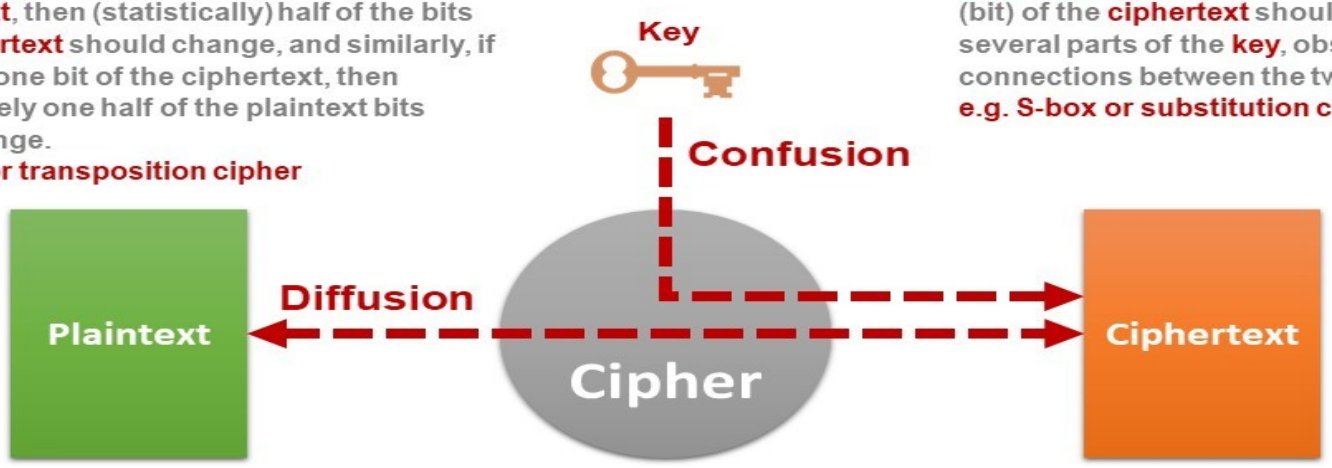
Lots of things, but two you should know are confusion and diffusion (diffusion is also known as the avalanche effect).

Claude Shannon, *A Mathematical Theory of Cryptography (1945 classified report)*

Confusion and Diffusion

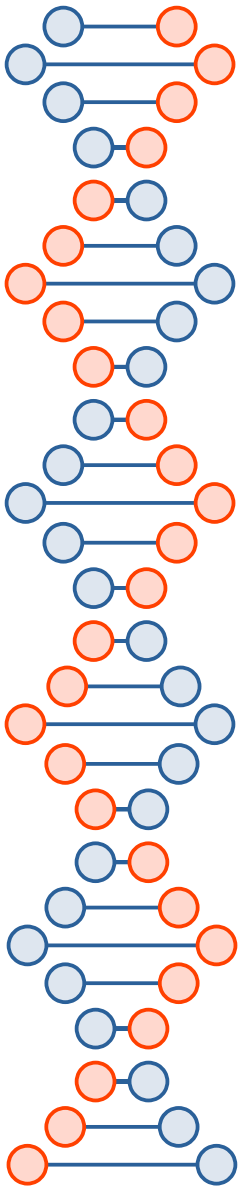
Diffusion means that if we change a single bit of the **plaintext**, then (statistically) half of the bits in the **ciphertext** should change, and similarly, if we change one bit of the ciphertext, then approximately one half of the plaintext bits should change.
e.g. P-box or transposition cipher

Confusion means that each binary digit (bit) of the **ciphertext** should depend on several parts of the **key**, obscuring the connections between the two.
e.g. S-box or substitution cipher



https://en.wikipedia.org/wiki/Confusion_and_diffusion

https://wentzwu.com/2019/09/02/confusion-and-diffusion/



Symmetric attack types

- Ciphertext only
  - With ciphertext only, attacker can recover the plaintext
  - Think Caesar cipher, or Viginere cipher
- Known plaintext
  - Give the attacker enough plaintext /ciphertext pairs, they'll recover the key
  - Linear cryptanalysis
- Chosen plaintext
  - Same as known plaintext, but the attacker gets to choose what plaintexts you encrypt
  - Differential cryptanalysis

Attacks on block ciphers

- Linear and differential cryptanalysis
  - NSA must have known about these when giving input about DES, rest of the world found out in the 1990s
- Many others
  - E.g., rotational cryptanalysis
- CBC padding oracle attacks and others that are typically performed on live systems

For more details and the image source for the following two slides, see:
A Tutorial on Linear and Differential Cryptanalysis, by Howard M. Heys
https://jedcrandall.github.io/courses/cse539spring2023/ldc_tutorial.pdf

Linear cryptanalysis

- Solve for the key using plaintext/ciphertext pairs and linear approximations
- XOR is linear arithmetic modulo 2, permutations are also linear, only S-boxes save you

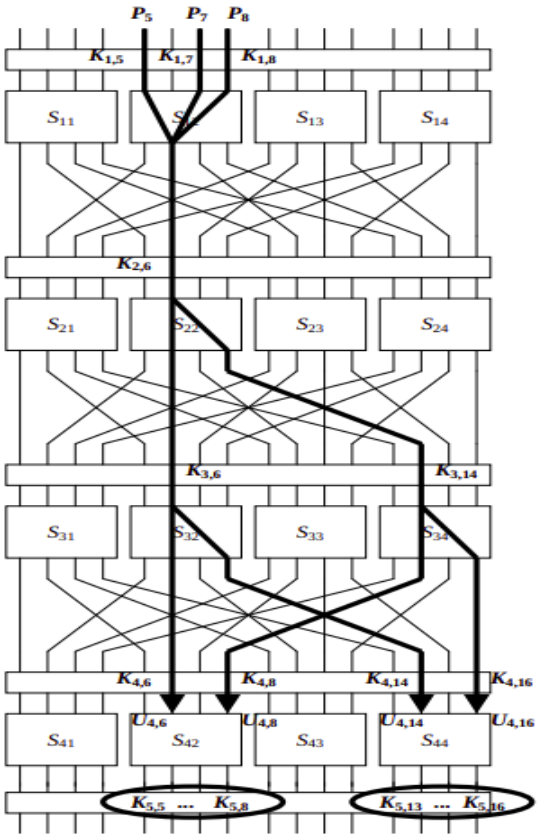


Figure 3. Sample Linear Approximation

Differential cryptanalysis

- Solve for key using plaintext/ciphertext pairs and propagated bit differences
- XOR and permutations don’t hide bit differences, only the S-boxes save you

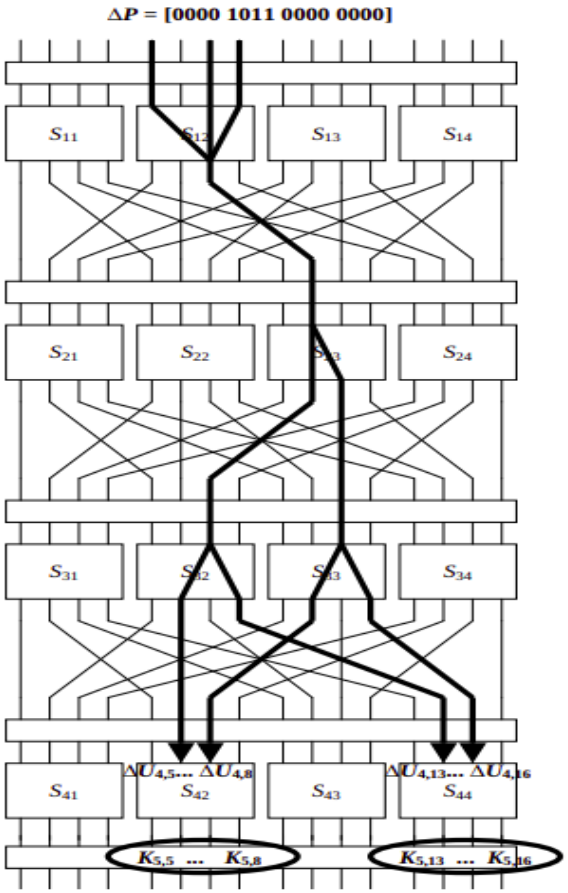
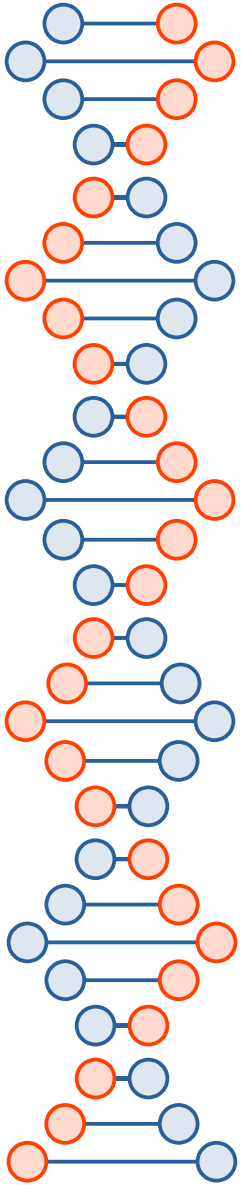


Figure 5. Sample Differential Characteristic

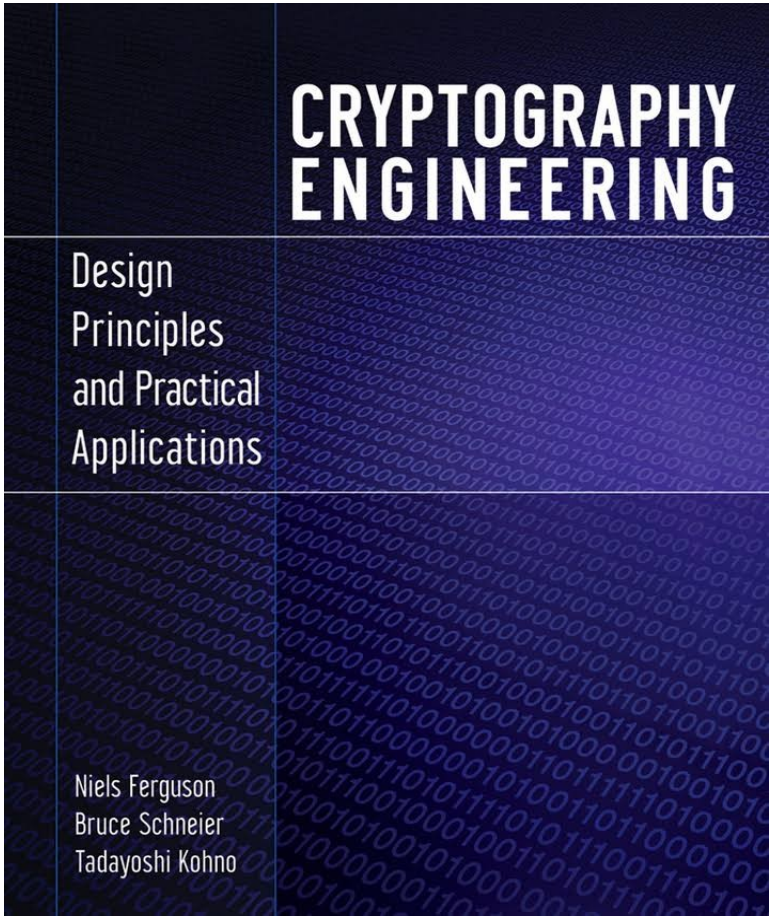
AES S-box requirements (review)

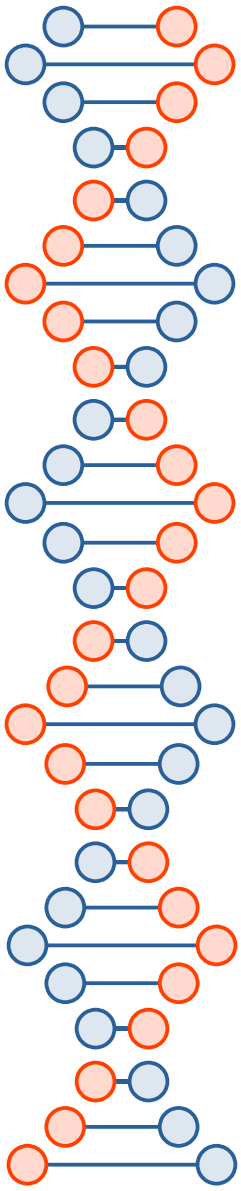
- Can’t pull it out of our &%# like the NSA did for DES
- Should have good nonlinear properties
- Should be reversible
  - Don’t want to use a Feistel structure for performance reasons

Galois fields...
https://www.youtube.com/watch?v=Ct2fyigNgPY



Cryptography Engineering by Ferguson *et al.*





Acknowledgments and resources

- Many of the above images are from Wikipedia
- https://www.youtube.com/watch?v=5mB_FUyfuZE&list=PLmh4YIWteoGgh0E2EuS4ZpzIi7ZhIW9Xp