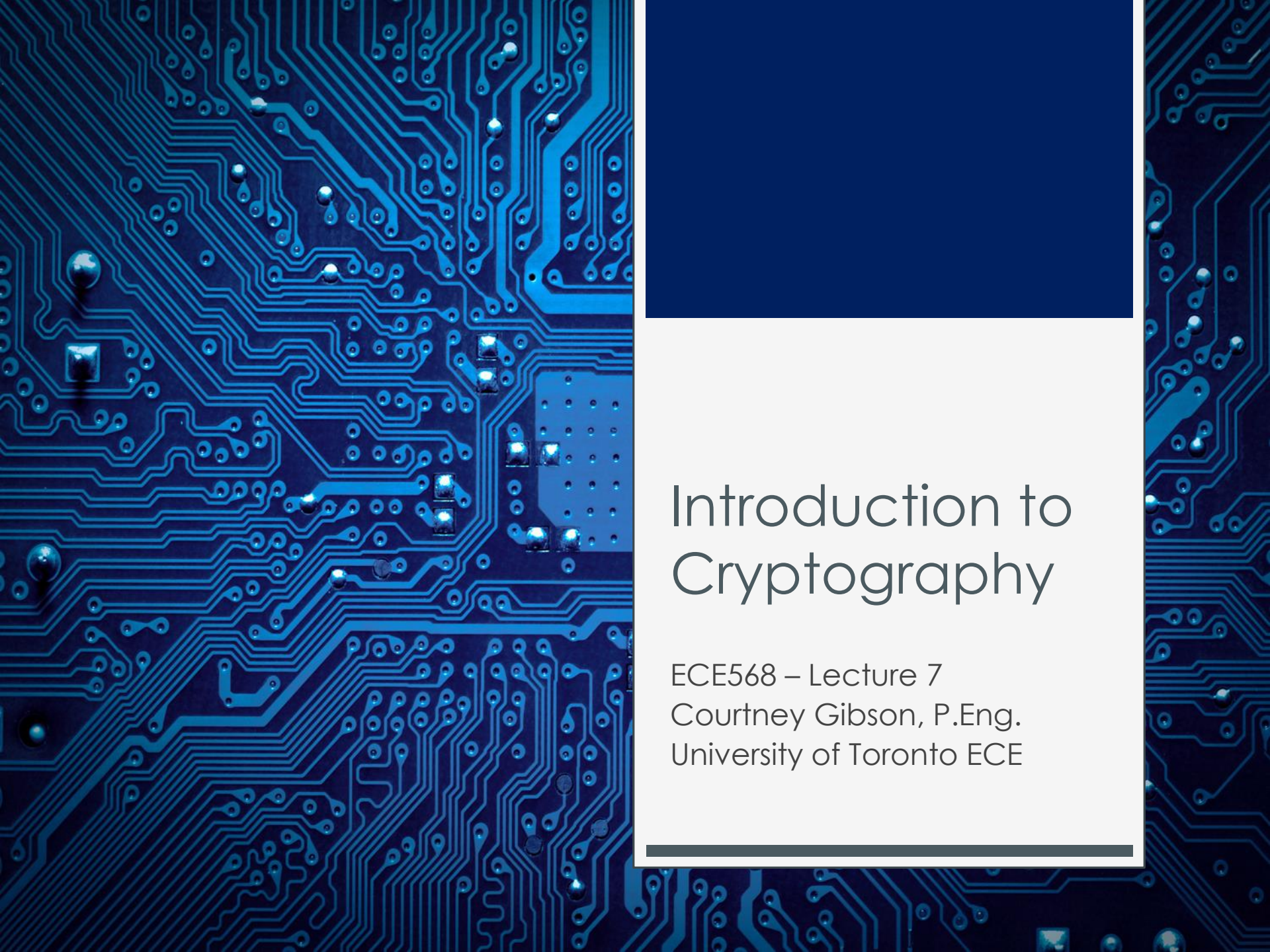


Introduction to Cryptography

ECE568 – Lecture 7

Courtney Gibson, P.Eng.
University of Toronto ECE

Outline

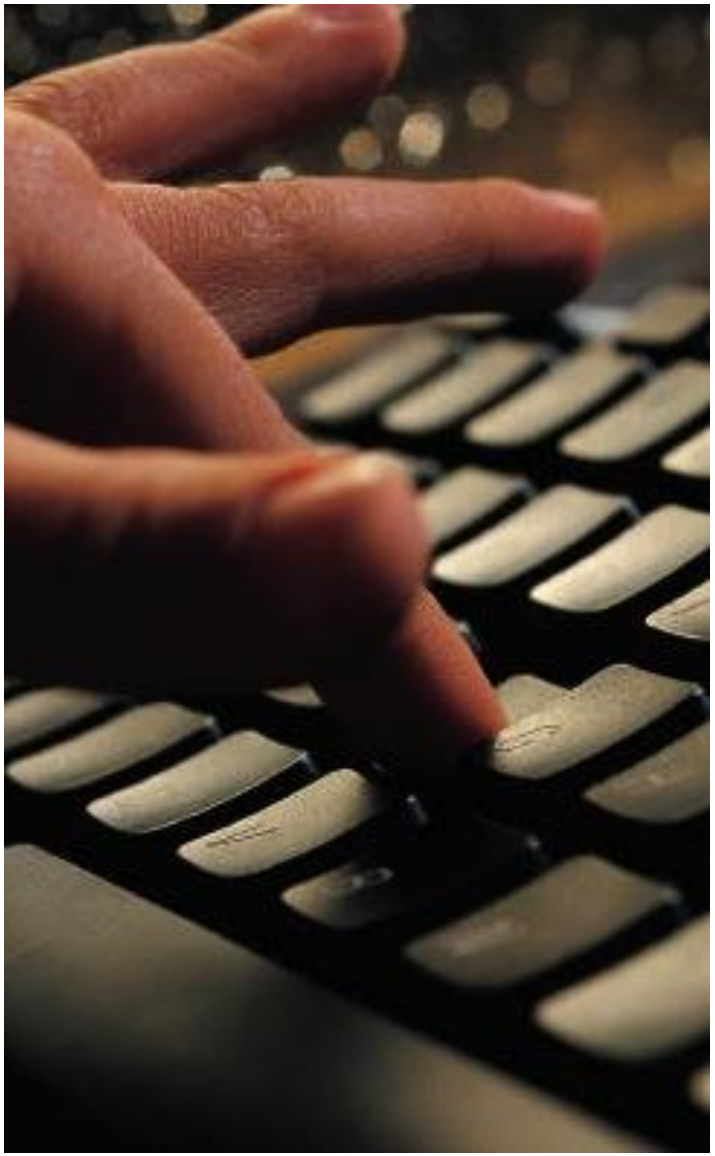
Introduction to Cryptography

- Why is it useful?
- Four important properties

Basic Ciphers

- Kerckhoffs' principle
- Substitution ciphers
- Poly-alphabetic and periodic ciphers
- One-Time Pad and Vernam ciphers

Stream Ciphers versus Block Ciphers

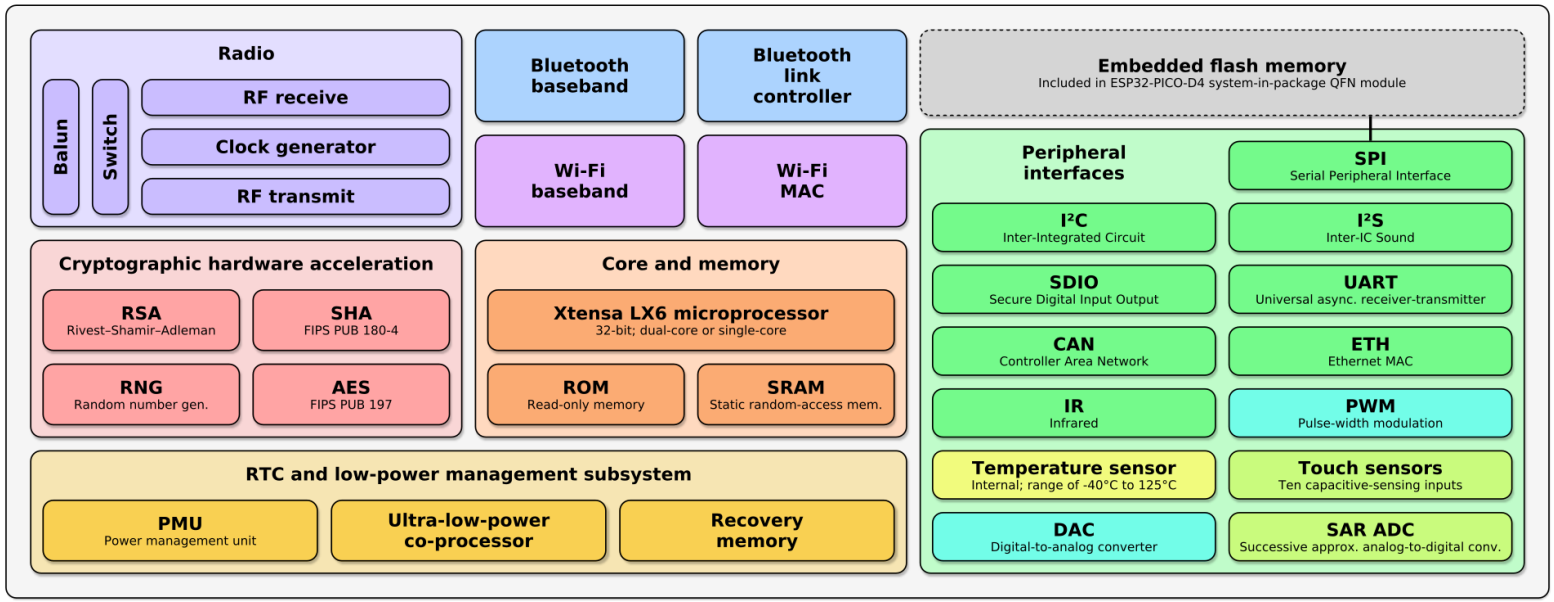


Introduction to Cryptography

Building Secure Products

Case Study: IoT Security

Espressif ESP32
ESP32-WROOM-32



Case Study: IoT Security

IoT Device
Microcontroller

Cloud
Service

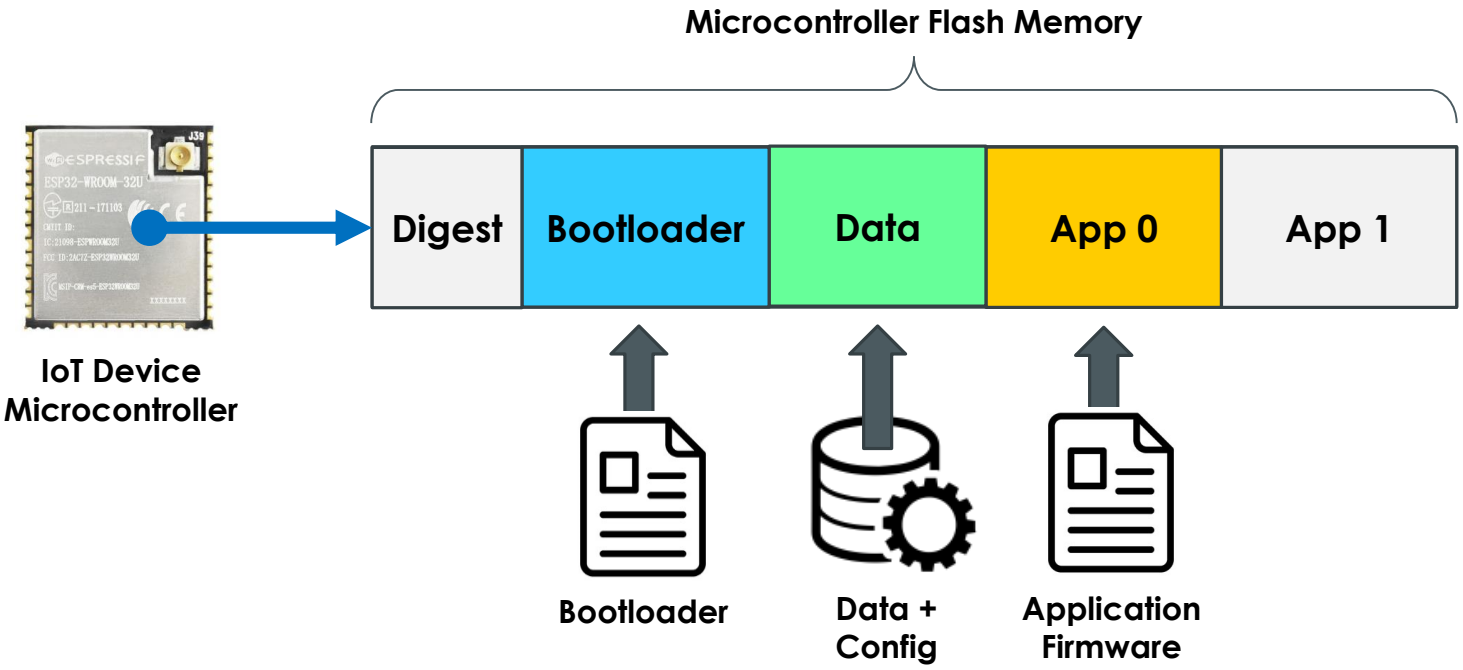
- Real service?
- Real device?
- Secure communications?

Application
Firmware

- Real application?
- Secure data?
- Malicious user?
- Tampering?

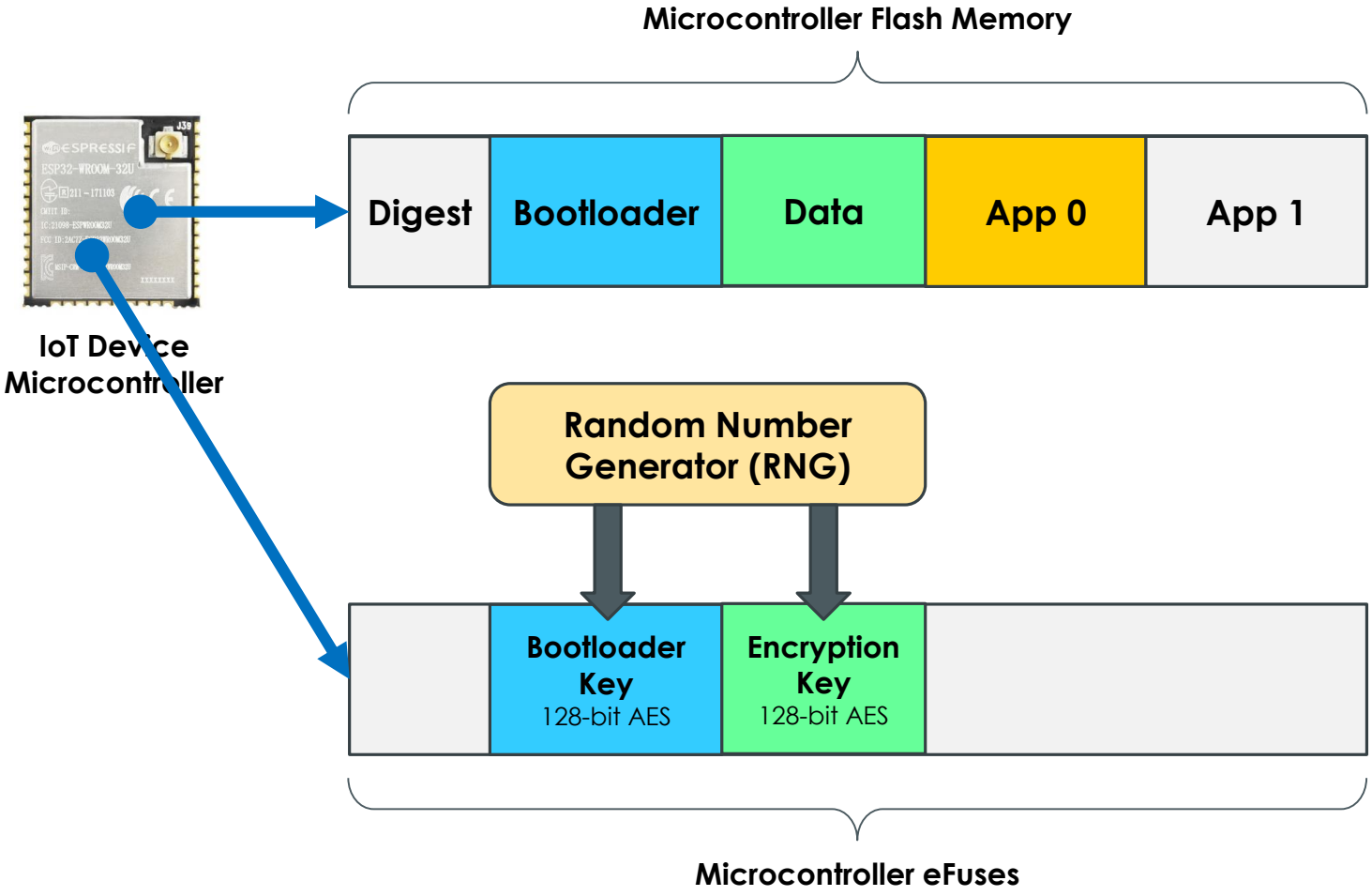
Case Study: IoT Security

Manufacturing Phase: Programming



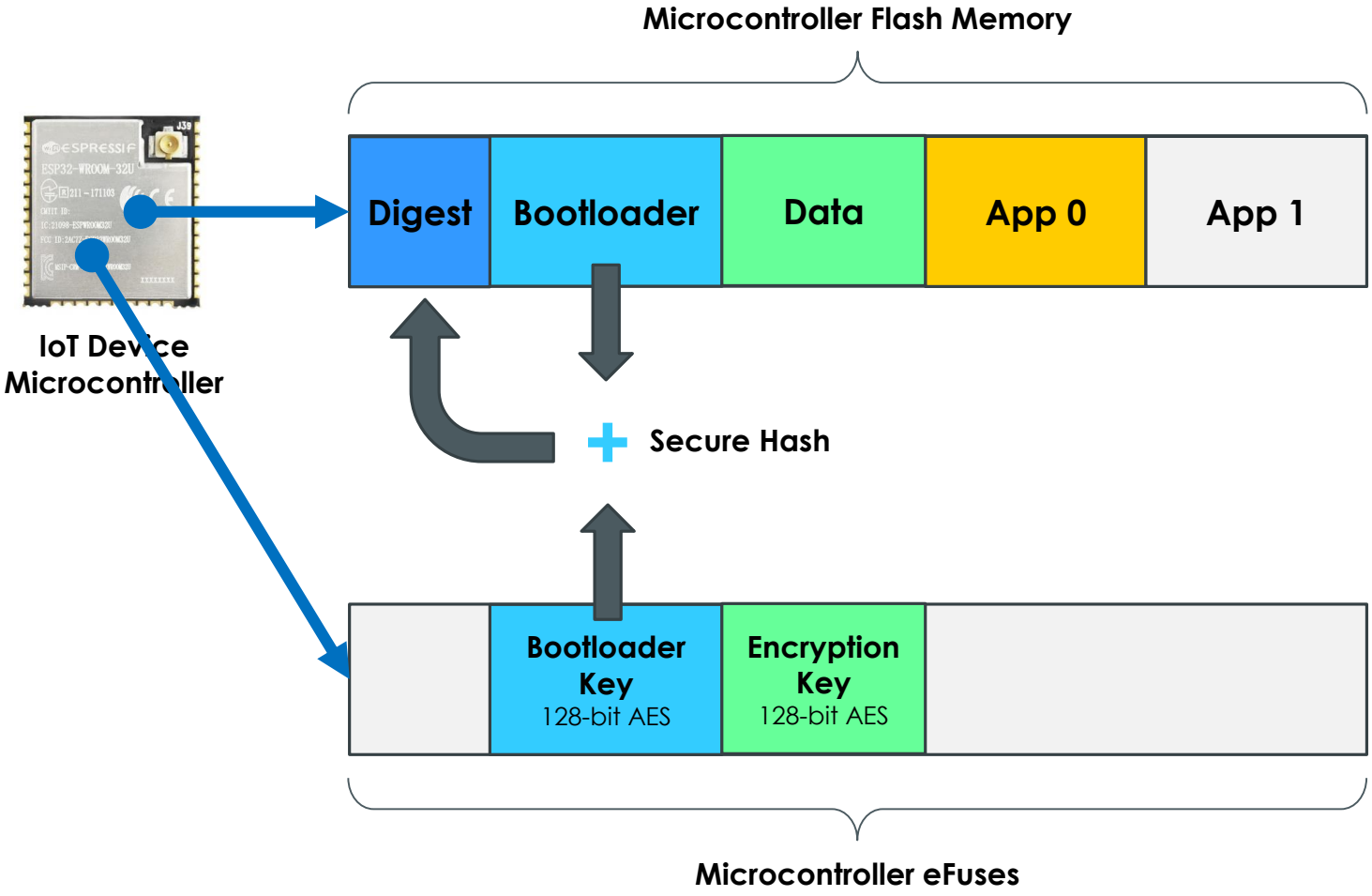
Case Study: IoT Security

Manufacturing Phase: Power-On (Step 1 of 7)



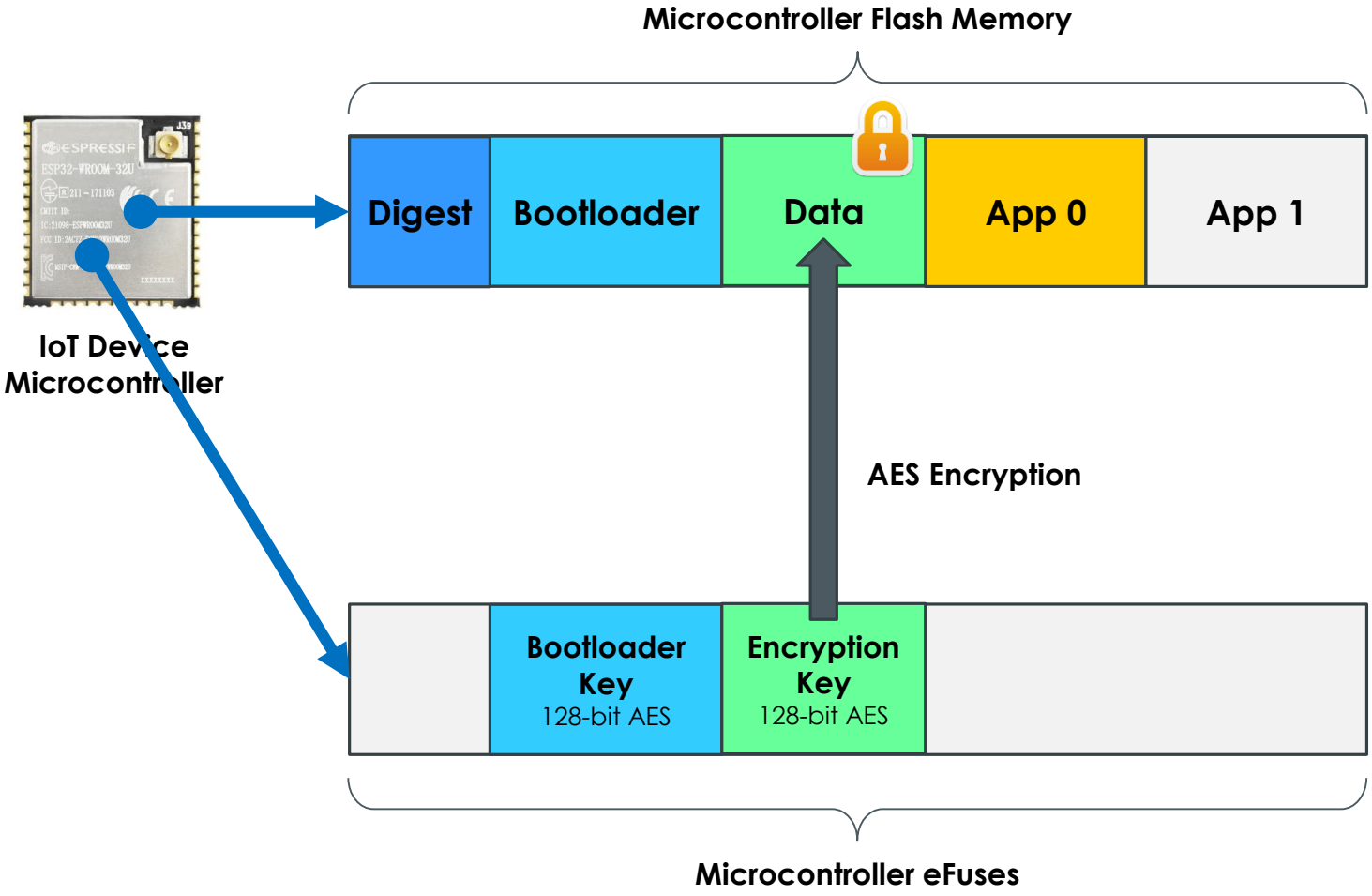
Case Study: IoT Security

Manufacturing Phase: Power-On (Step 2 of 7)



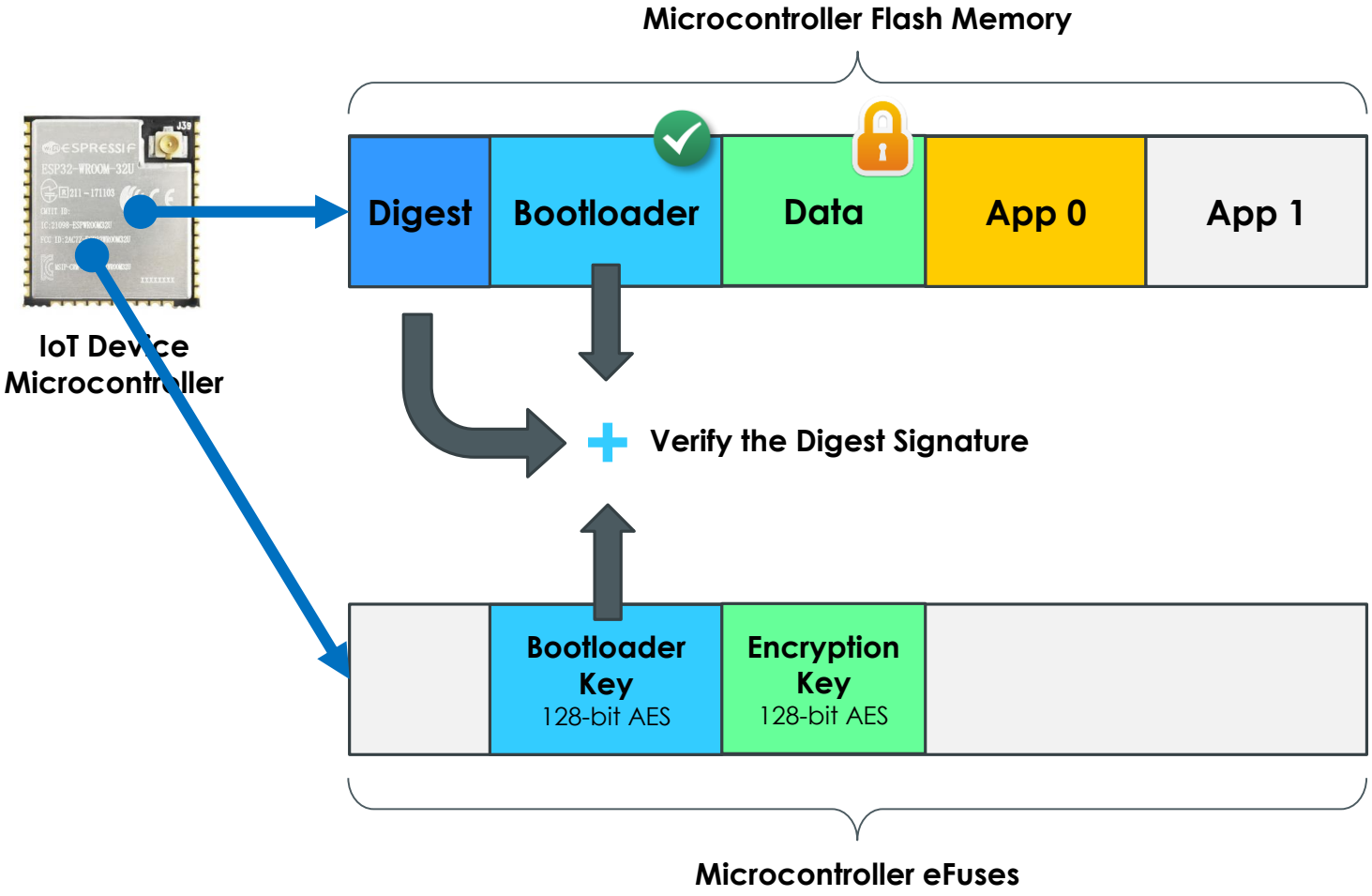
Case Study: IoT Security

Manufacturing Phase: Power-On (Step 3 of 7)



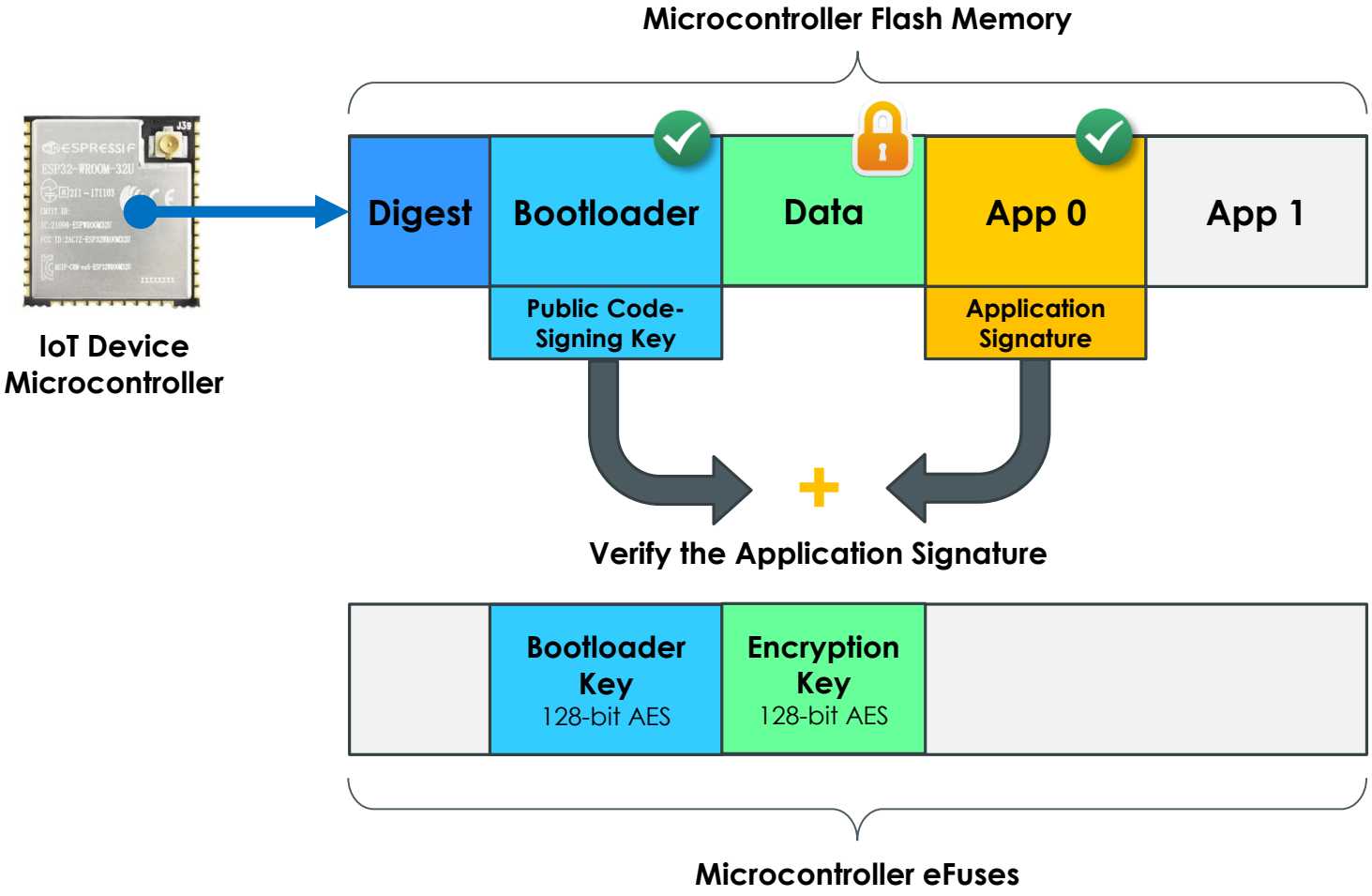
Case Study: IoT Security

Manufacturing Phase: Power-On (Step 4 of 7)



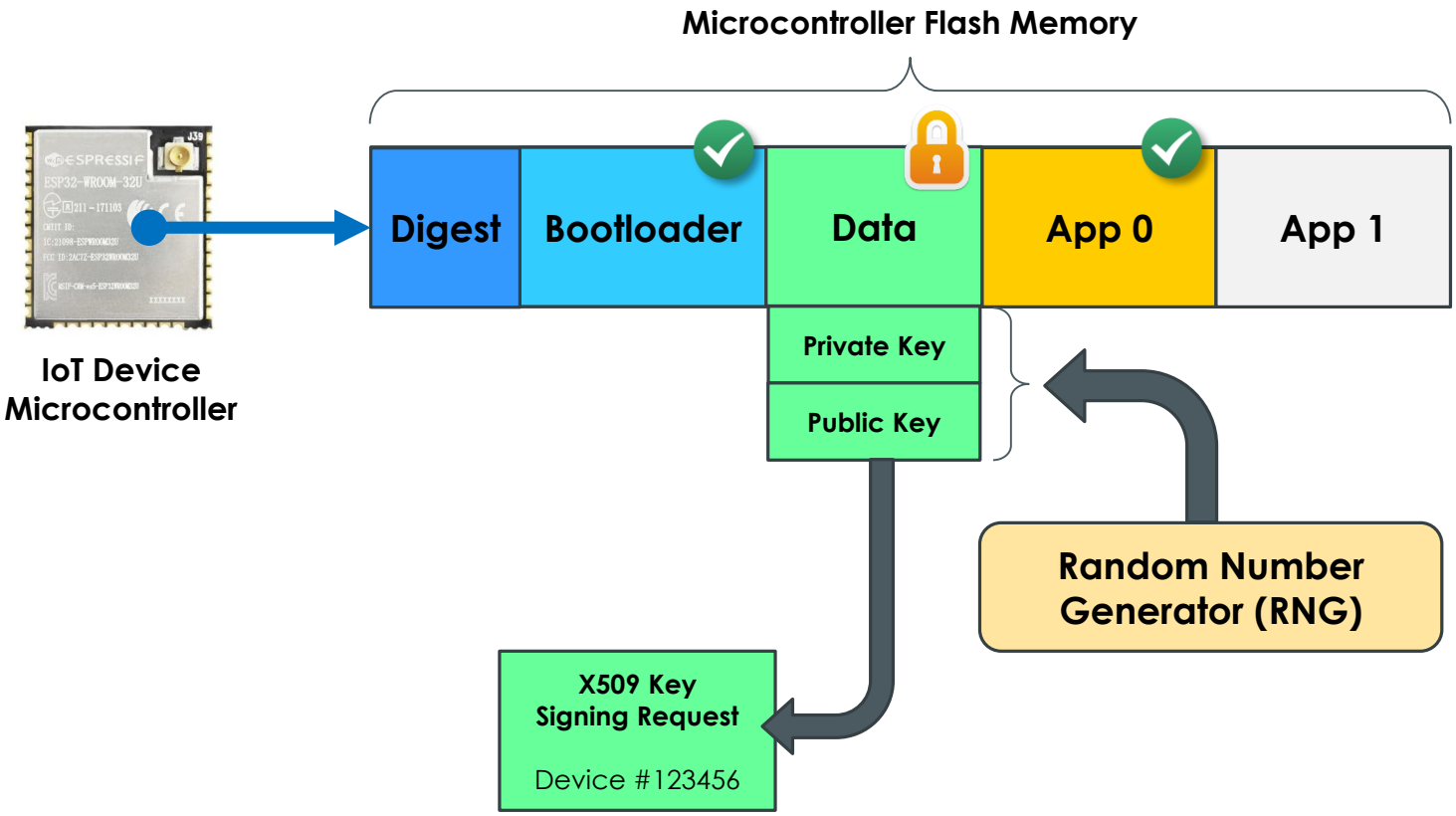
Case Study: IoT Security

Manufacturing Phase: Power-On (Step 5 of 7)



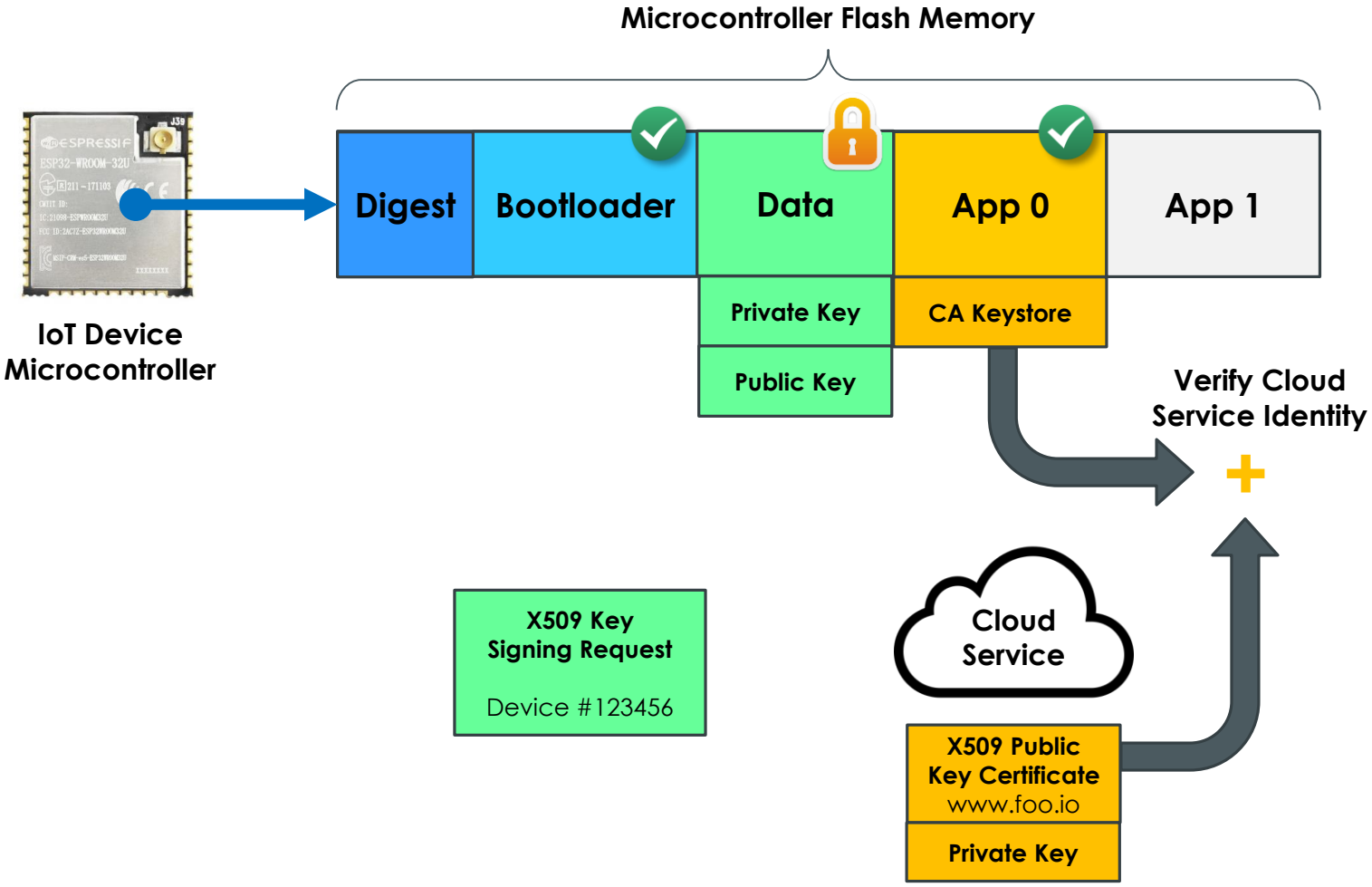
Case Study: IoT Security

Manufacturing Phase: Power-On (Step 6 of 7)



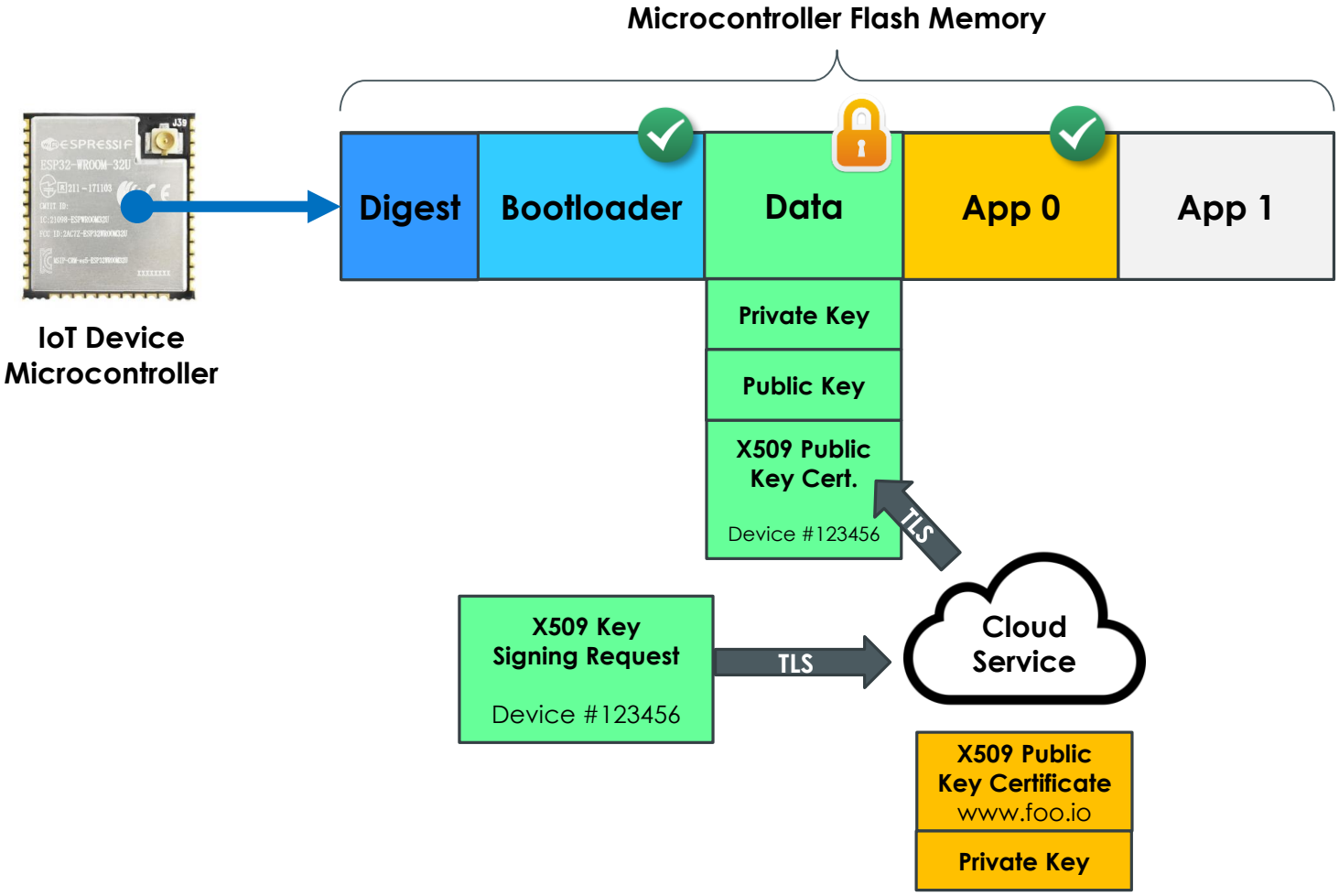
Case Study: IoT Security

Manufacturing Phase: Power-On (Step 7 of 8)



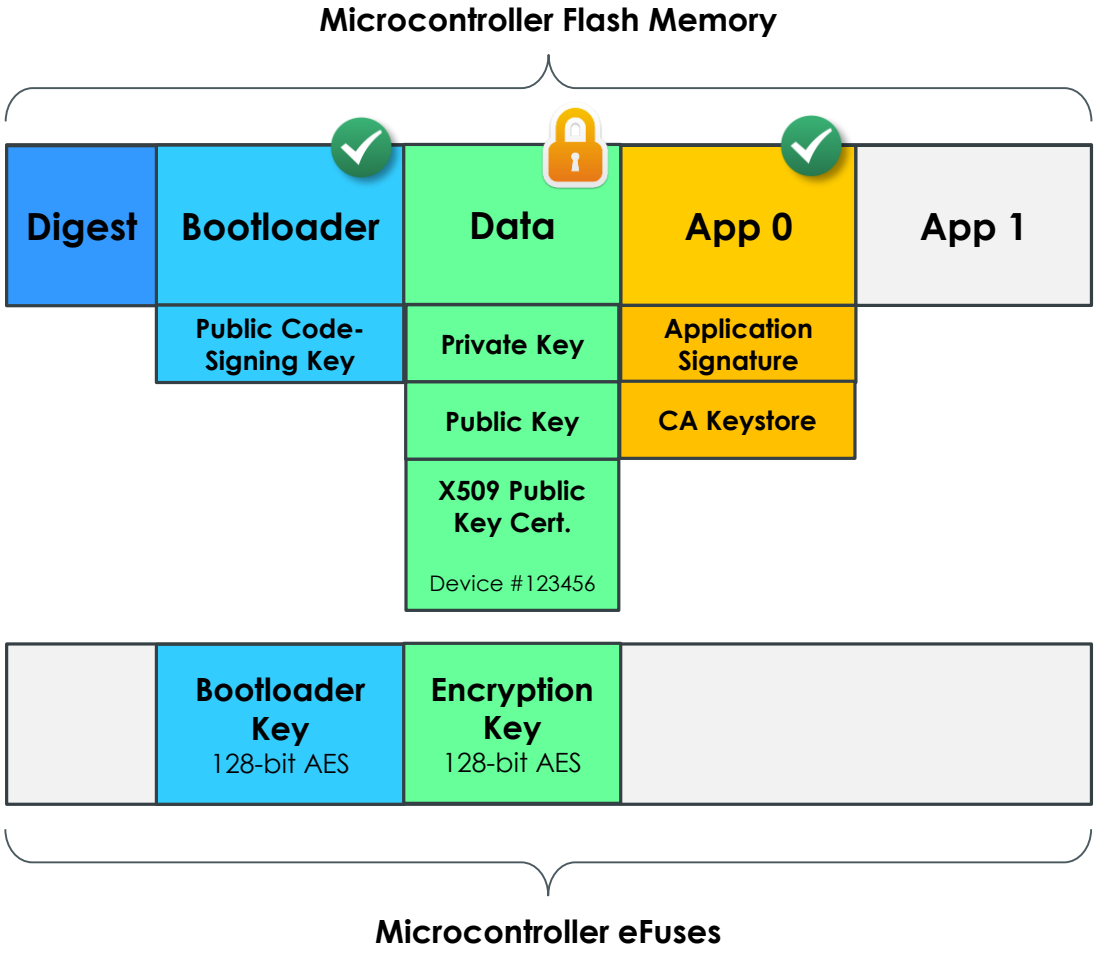
Case Study: IoT Security

Manufacturing Phase: Power-On (Step 8 of 8)



Case Study: IoT Security

IoT Device Microcontroller



Case Study: Door Alarm

Roll over image to zoom in

Mengshen Door and Window Alarm - Wireless Burglar Alarm with 105db Loud Sound and Bright Light, Easy to Install (Includes 1 Alarm and 1 Remote Control)

Brand: Mengshen

★★★★½ 122 ratings | 11 answered questions

Price: CDN$ 19.99 ✓prime

You could get **5% back at Amazon.ca**, Whole Foods Market stores, grocery stores, and restaurants for 6 months upon approval for the **Amazon.ca Rewards Mastercard**. See terms and learn more.

New (2) from CDN$ 19.99 ✓prime

Style: **A-1 Alarm and 1 Remote**

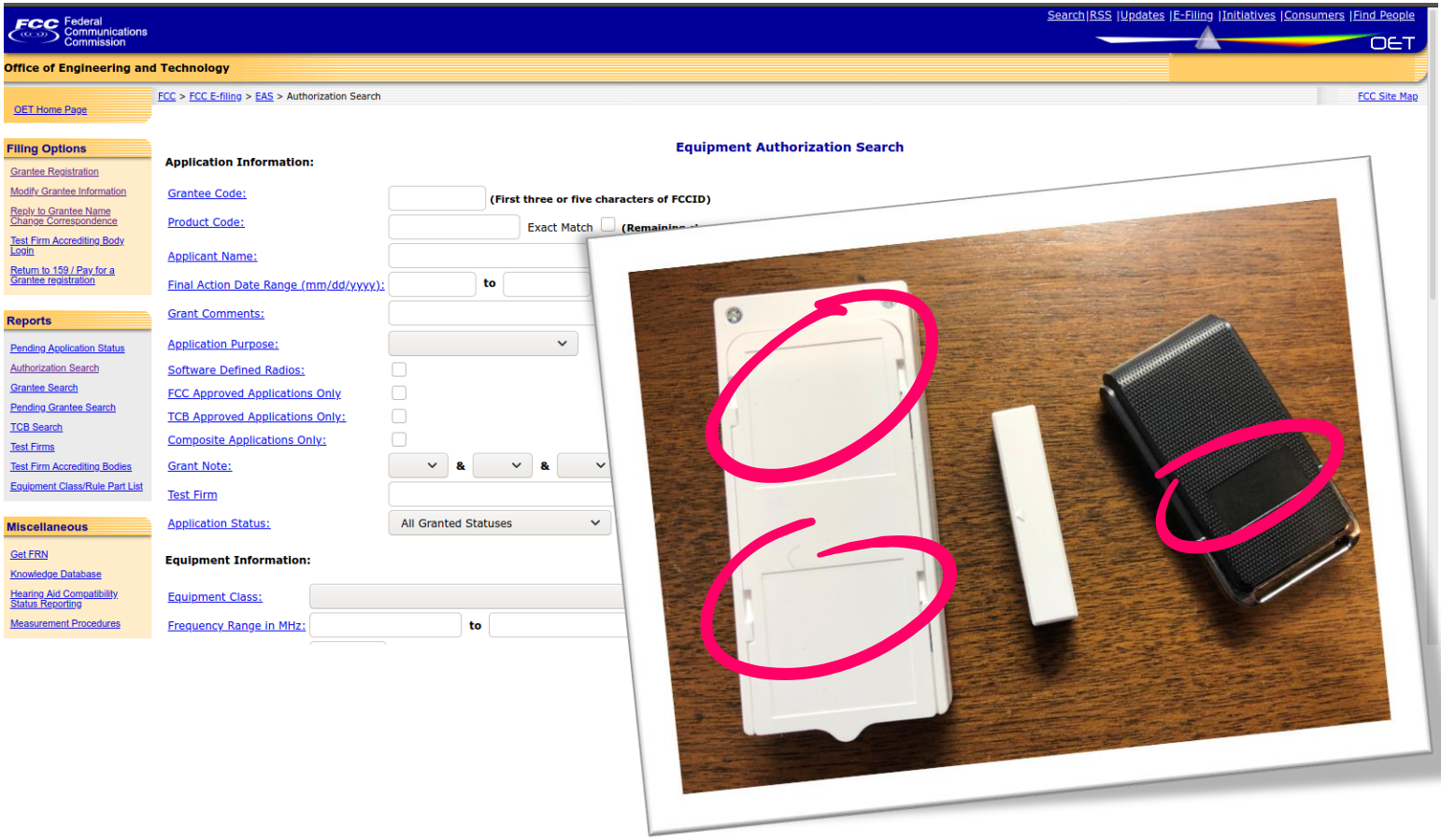
A-1 Alarm and 1 Remote	B-1 Alarm and 2 Remote	C-2 Alarm and 1 Remote
CDN$ 19.99 ✓prime	CDN$ 24.99 ✓prime	CDN$ 29.99 ✓prime

- Voice and Light Alarm: 108db loud alarm and bright light, can effectively scare the thieves and intruders away,
- Magnetic Sensor Alarm: Sensitive, lower false trigger rate,
- Multi-funciton Security Alarm: It has the function of arm, disarm, emergency alarm and doorbell, widly meet your demand,
- Easy Operation: With remote control, it will be very easy to operate,
- Flexible Options: If you need extra remote controls for family member or just for spare. Pls refer to "More Remote Control" in the "Special offers and

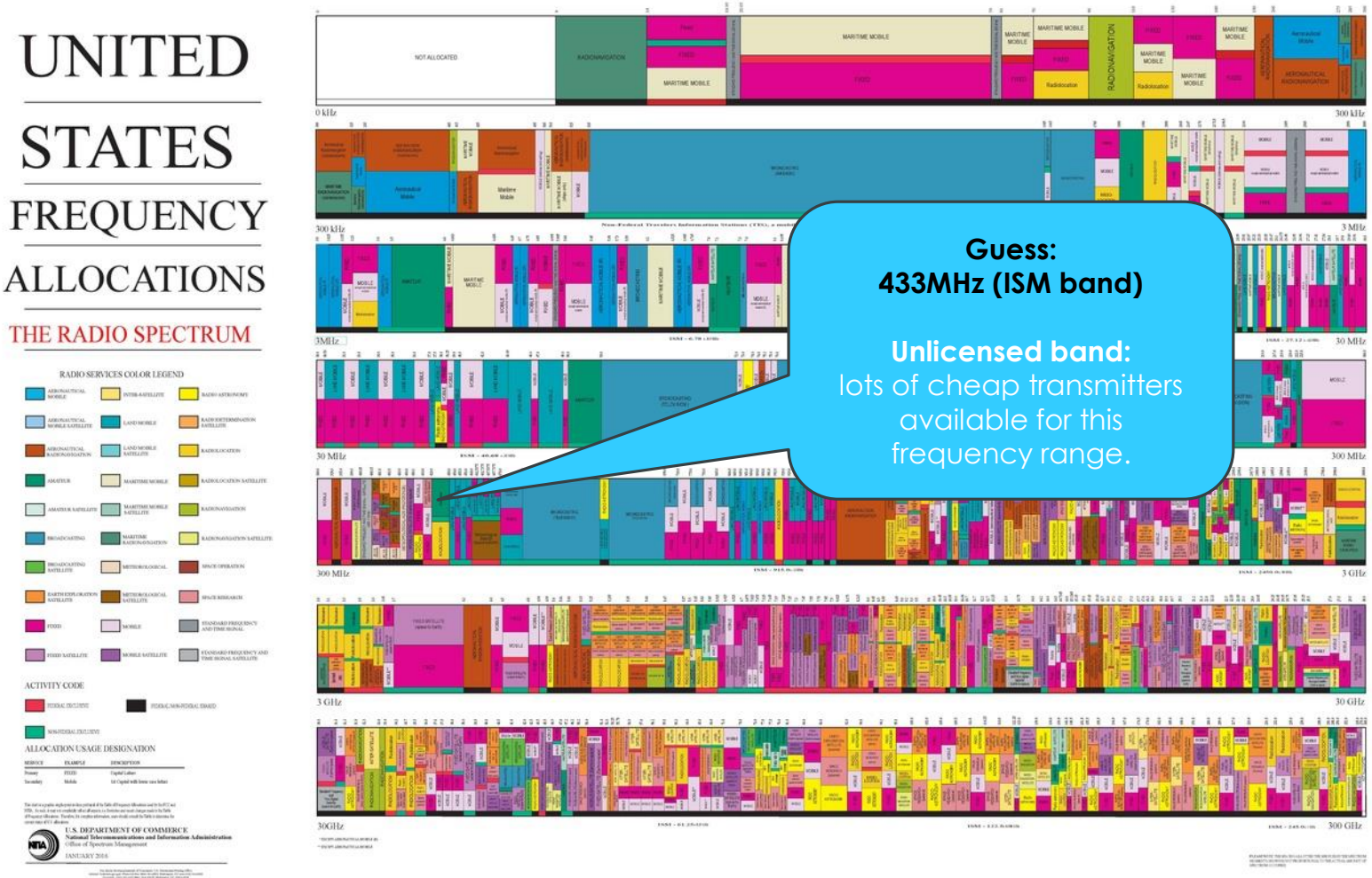
Case Study: Door Alarm

Case Study: Door Alarm

Investigation: FCC.gov (Equipment Authorization Search)

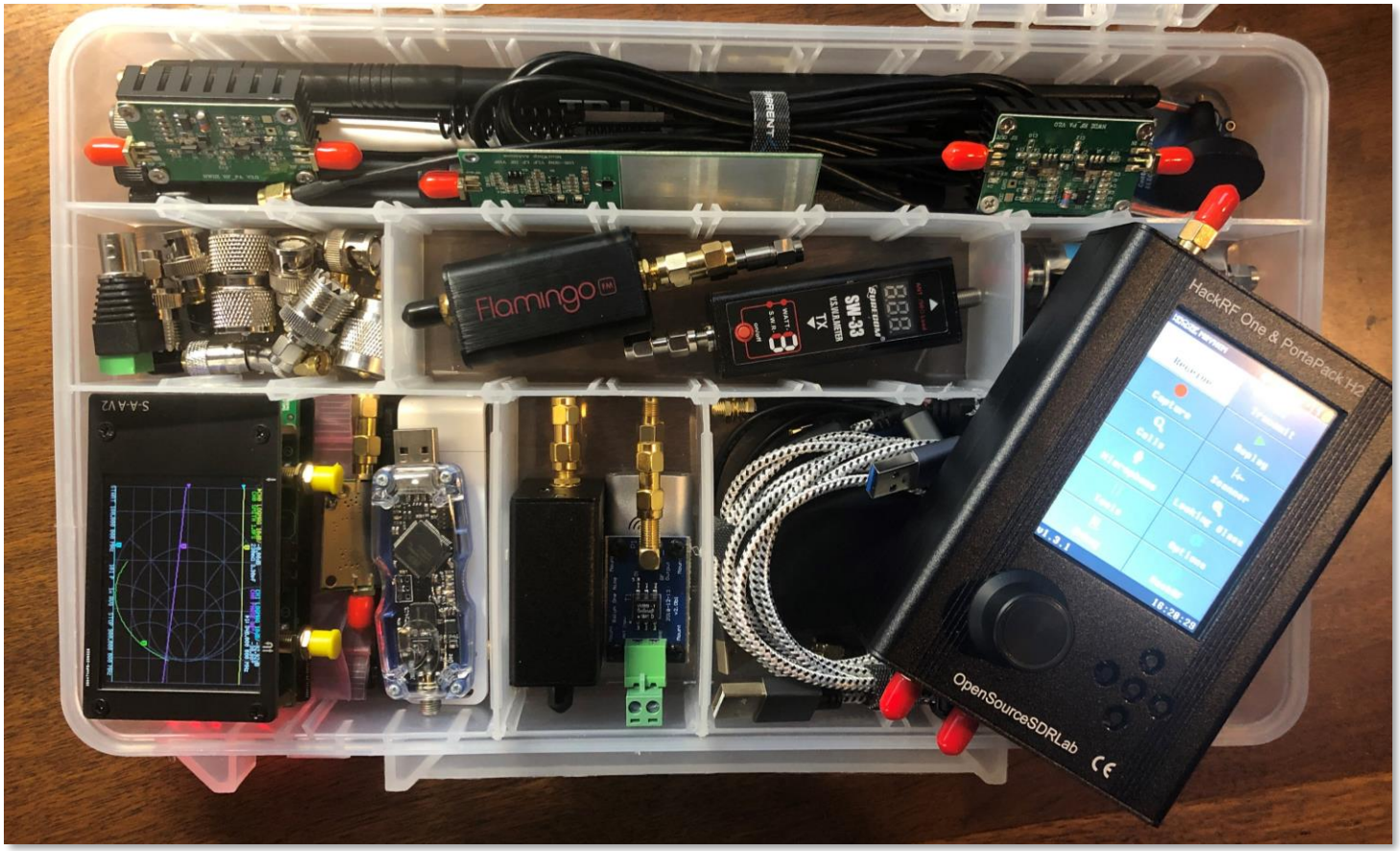


Case Study: Door Alarm



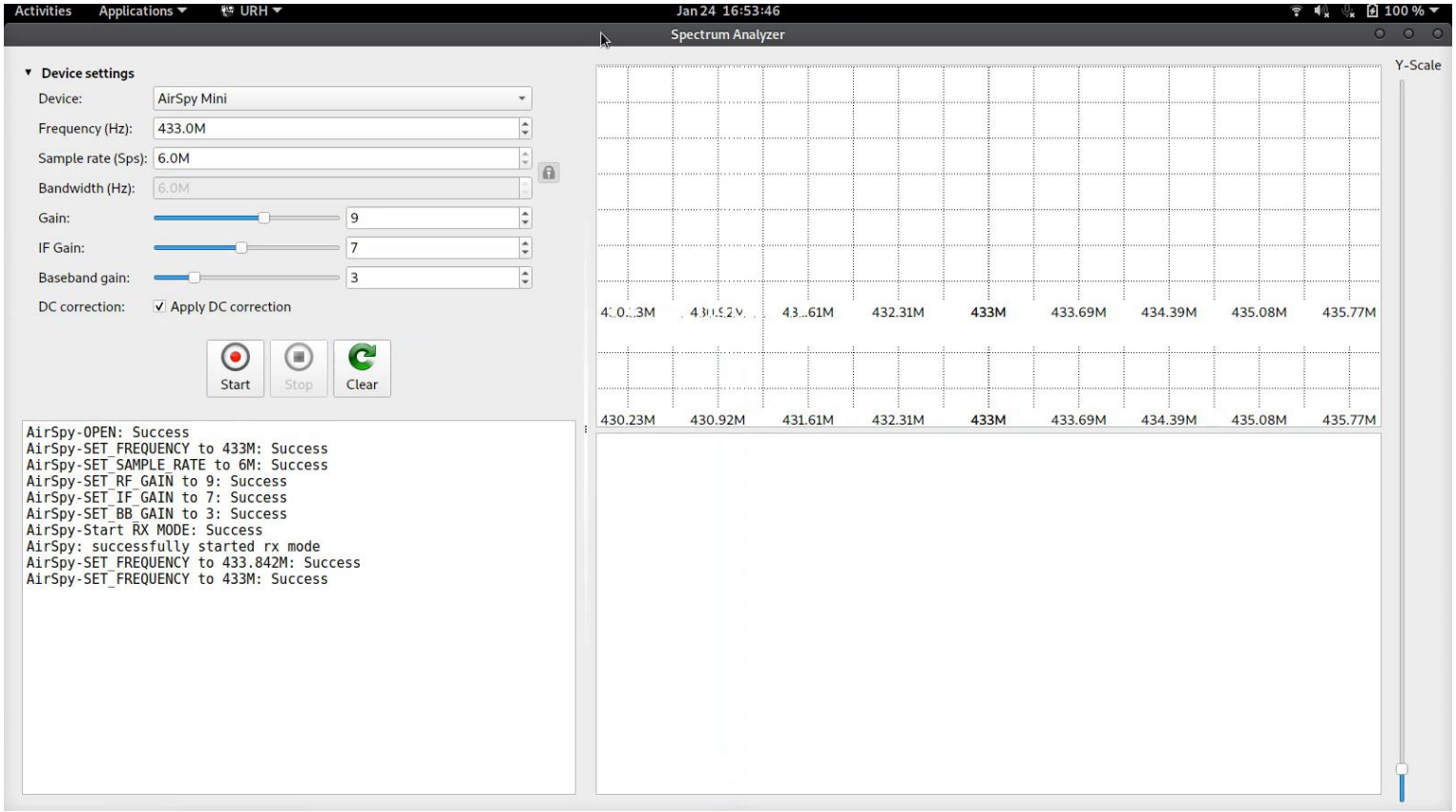
Case Study: Door Alarm

Investigation: Software Defined Radio (SDR)



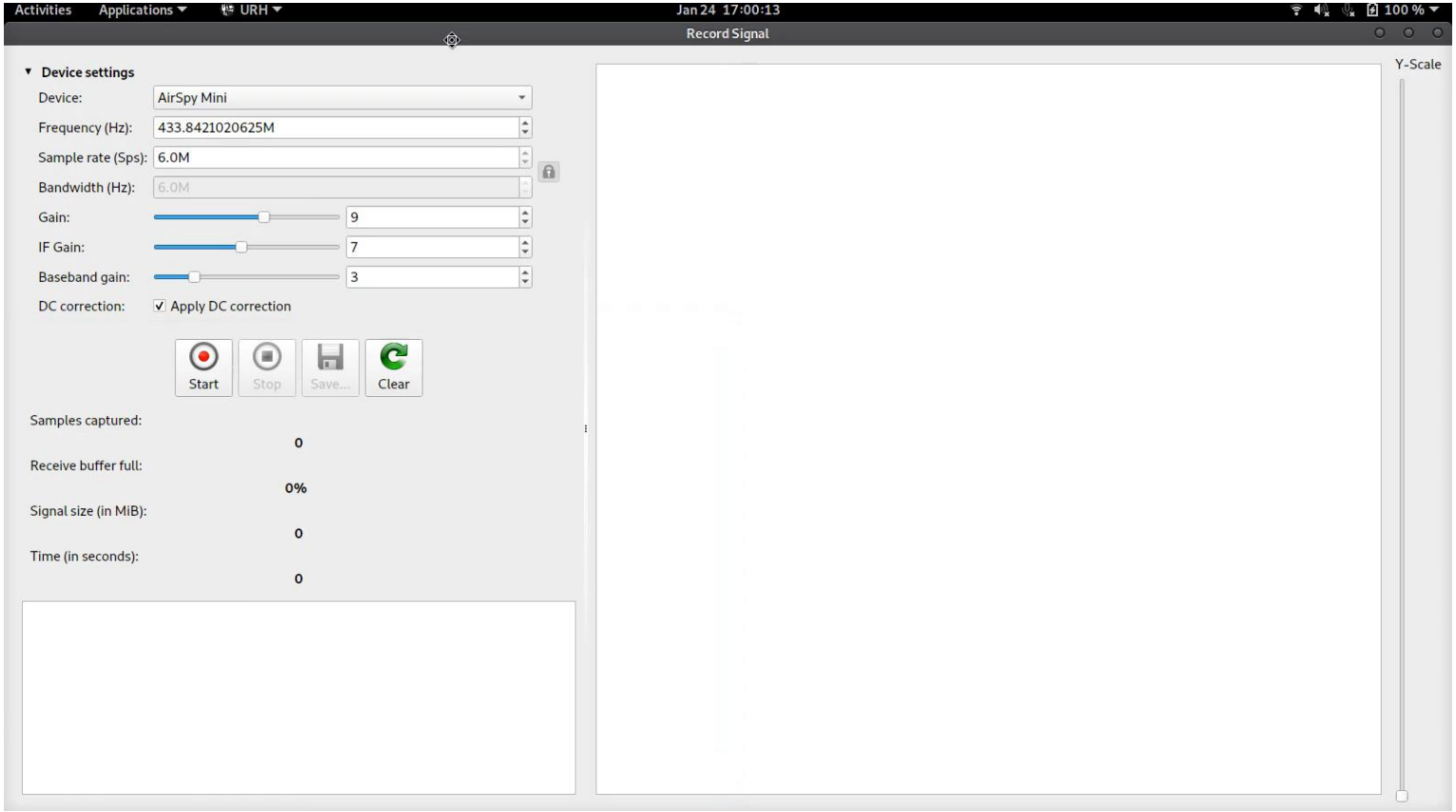
Case Study: Door Alarm

Investigation: Universal Radio Hacker (URH)

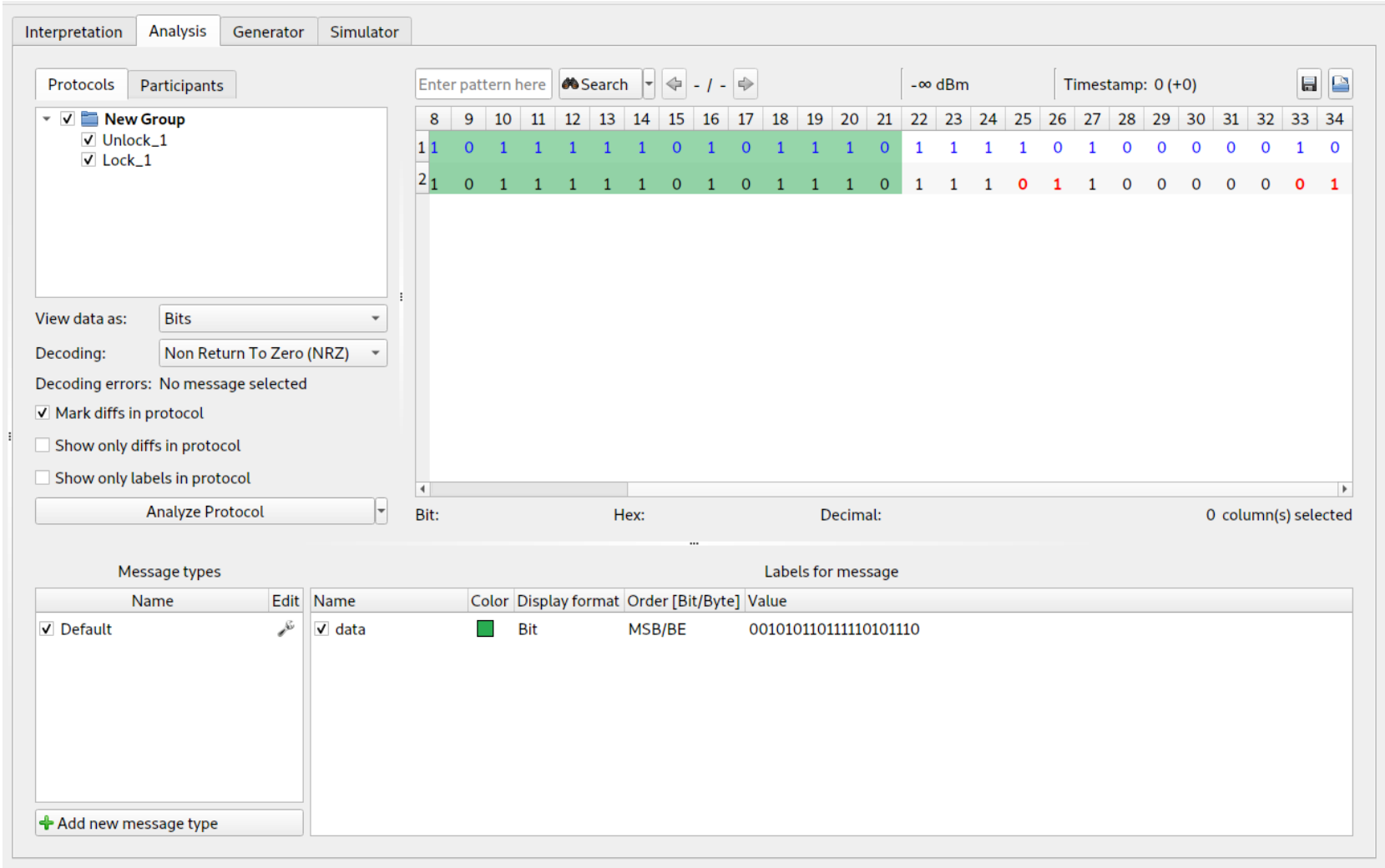


Case Study: Door Alarm

Investigation: Universal Radio Hacker (URH)

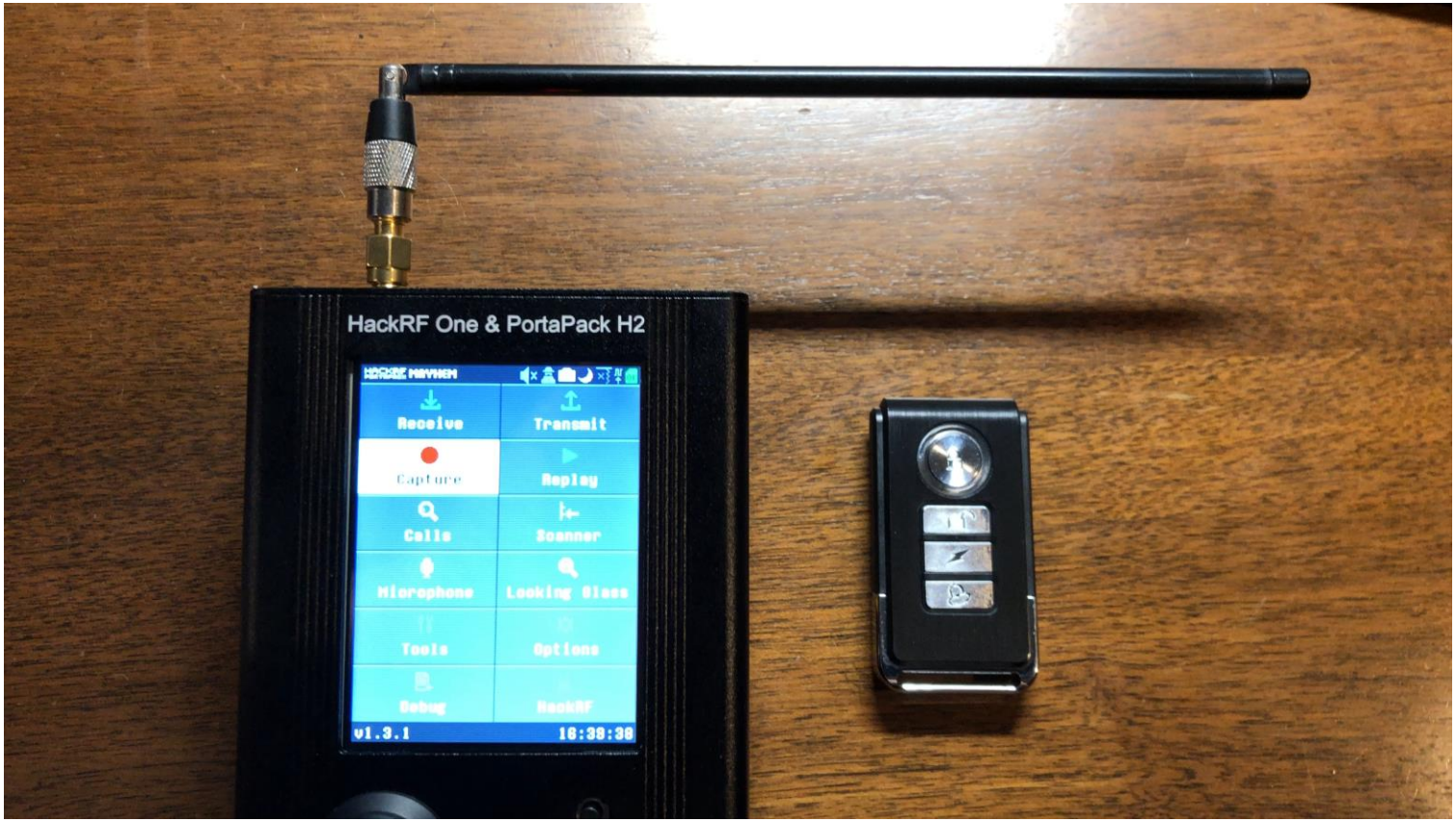



Interpretation
Analysis
Generator
Simulator
1: Complex Signal
Lock_1
Noise: 0.0025
Center: 0.0093
Samples/Symbol: 12000
Error Tolerance: 26
Modulation: ASK
Bits/Symbol: 1
Autodetect parameters
Signal View: Analog
Show Signal as Bits
Y-Scale
0 selected
0.00 ns
-∞ dBm
Filter (moving average)
00101011011111010111011110110000001 [Pause: 0 samples]
2: Complex Signal
Unlock_1
Noise: 0.0023
Center: 0.0297
Samples/Symbol: 12000
Error Tolerance: 31
Modulation: ASK
Bits/Symbol: 1
Autodetect parameters
Signal View: Analog
Show Signal as Bits
Y-Scale
0 selected
0.00 ns
-∞ dBm
Filter (moving average)
00101011011111010111011111010000010 [Pause: 0 samples]


Interpretation
Analysis
Generator
Simulator
Protocols
Participants
New Group
Unlock_1
Lock_1
Enter pattern here
Search
- / -
-∞ dBm
Timestamp: 0 (+0)
8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34
1 1 0 1 1 1 1 1 0 1 0 1 1 1 0 1 1 1 1 0 1 0 0 0 0 0 1 0
2 1 0 1 1 1 1 1 0 1 0 1 1 1 0 1 1 1 0 1 1 0 0 0 0 0 0 1
View data as: Bits
Decoding: Non Return To Zero (NRZ)
Decoding errors: No message selected
Mark diffs in protocol
Show only diffs in protocol
Show only labels in protocol
Analyze Protocol
Bit:
Hex:
Decimal:
0 column(s) selected
Message types
Name
Edit
Default
Add new message type
Labels for message
Name
Color
Display format
Order [Bit/Byte]
Value
data
Bit
MSB/BE
00101011011111010111 0

Case Study: Door Alarm

Investigation: HackRF One + PortaPack H2 + Mayhem Firmware

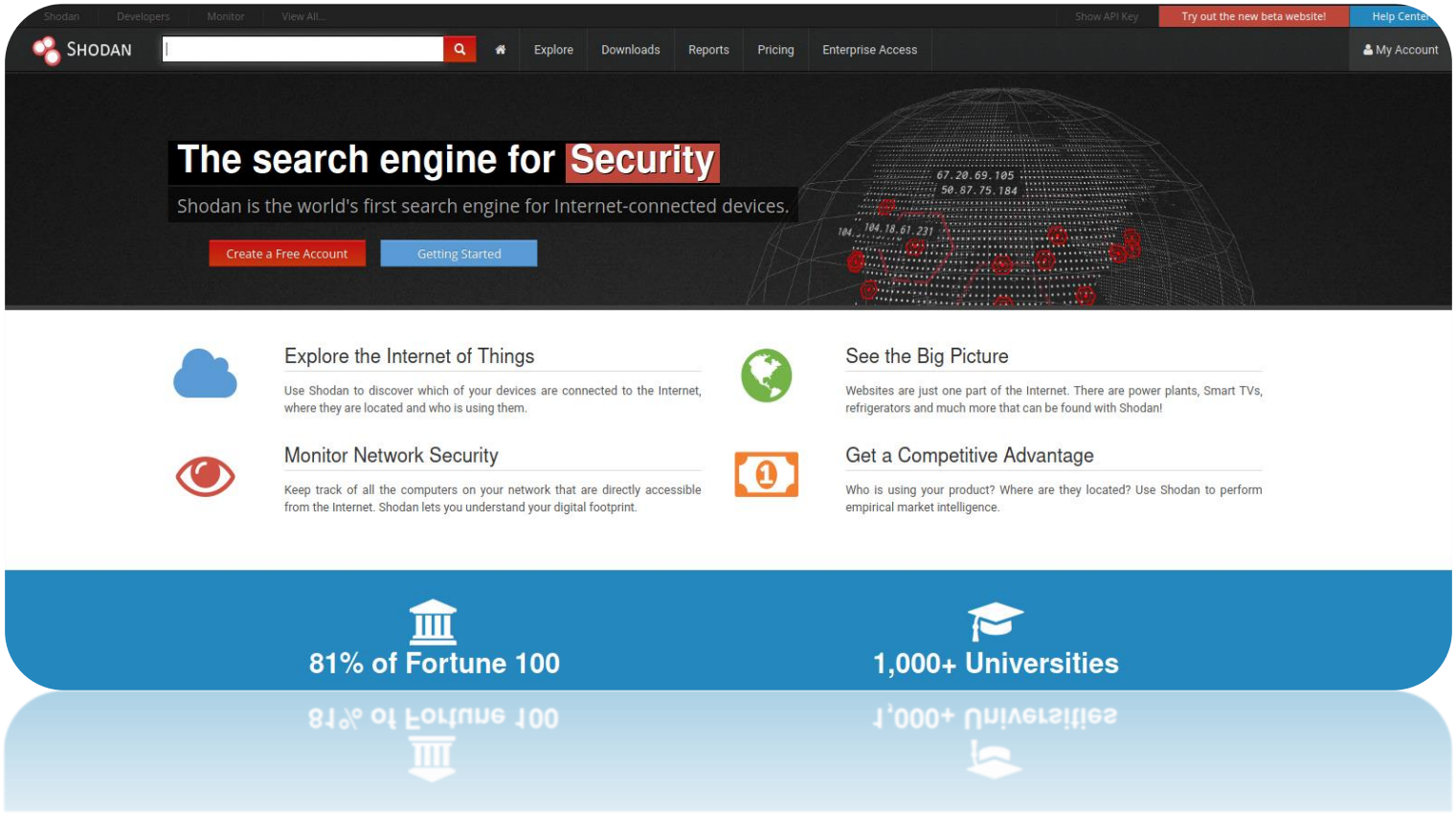


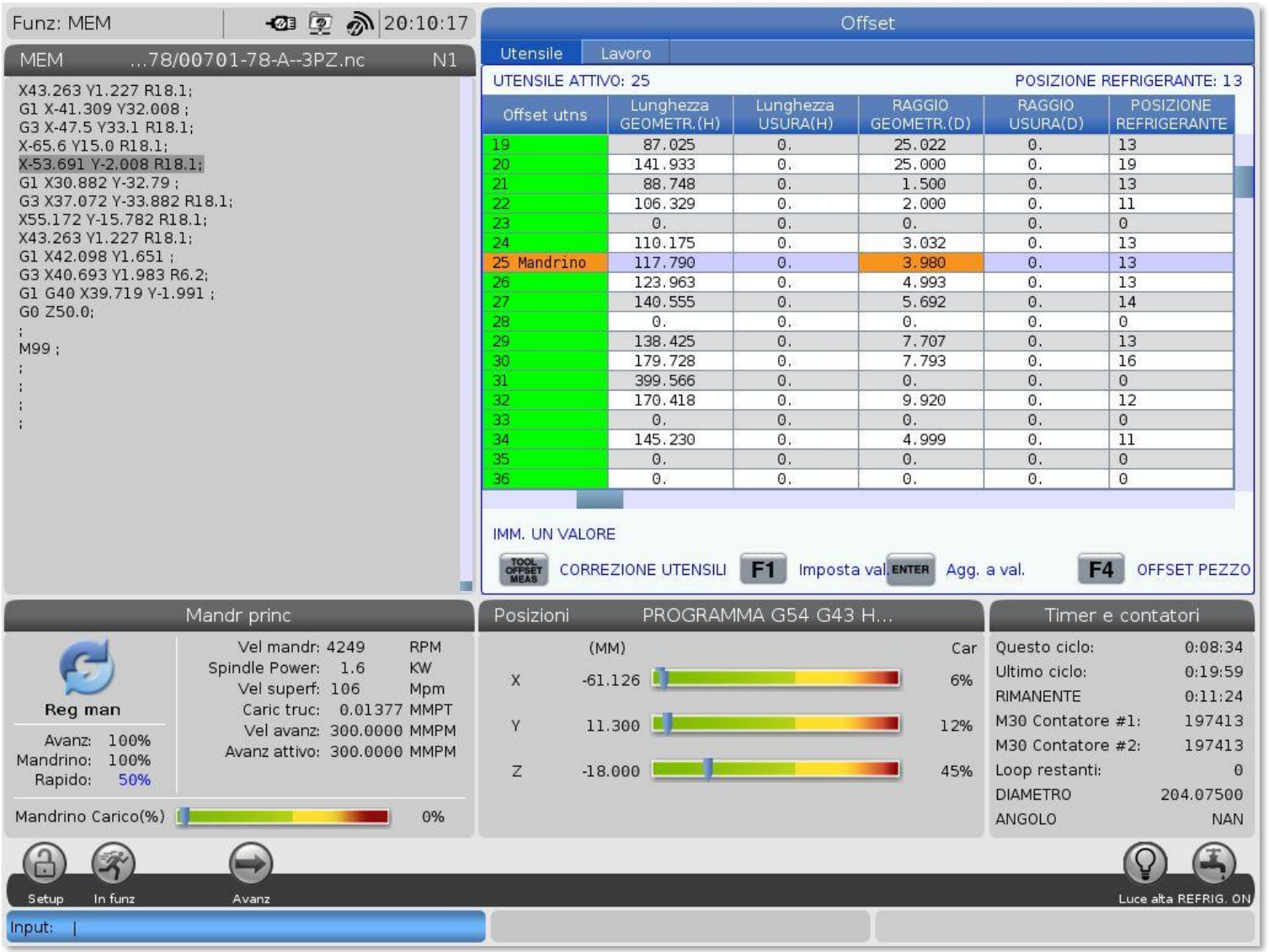
Case Study: Door Alarm

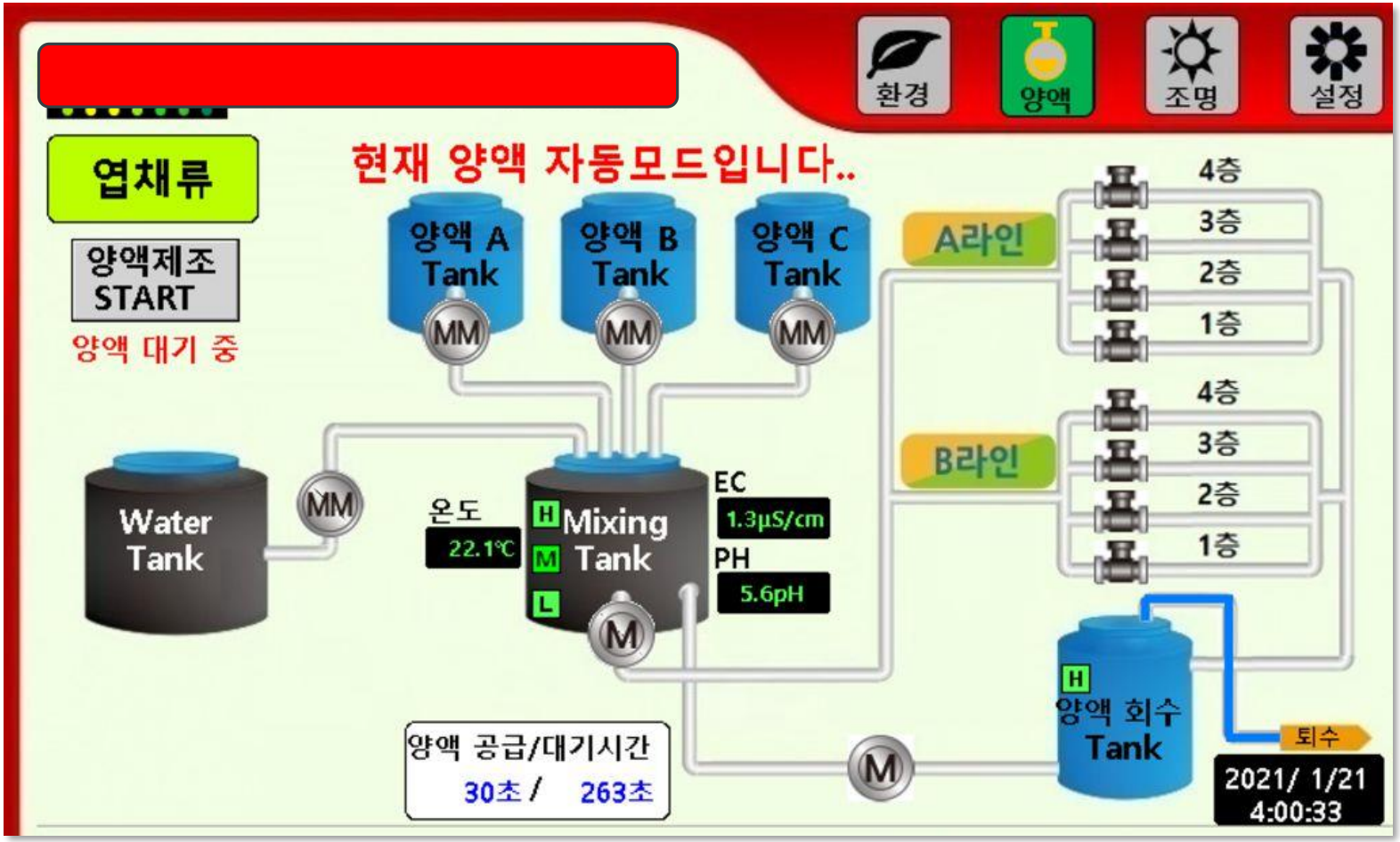
Investigation: HackRF One + PortaPack H2 + Mayhem Firmware

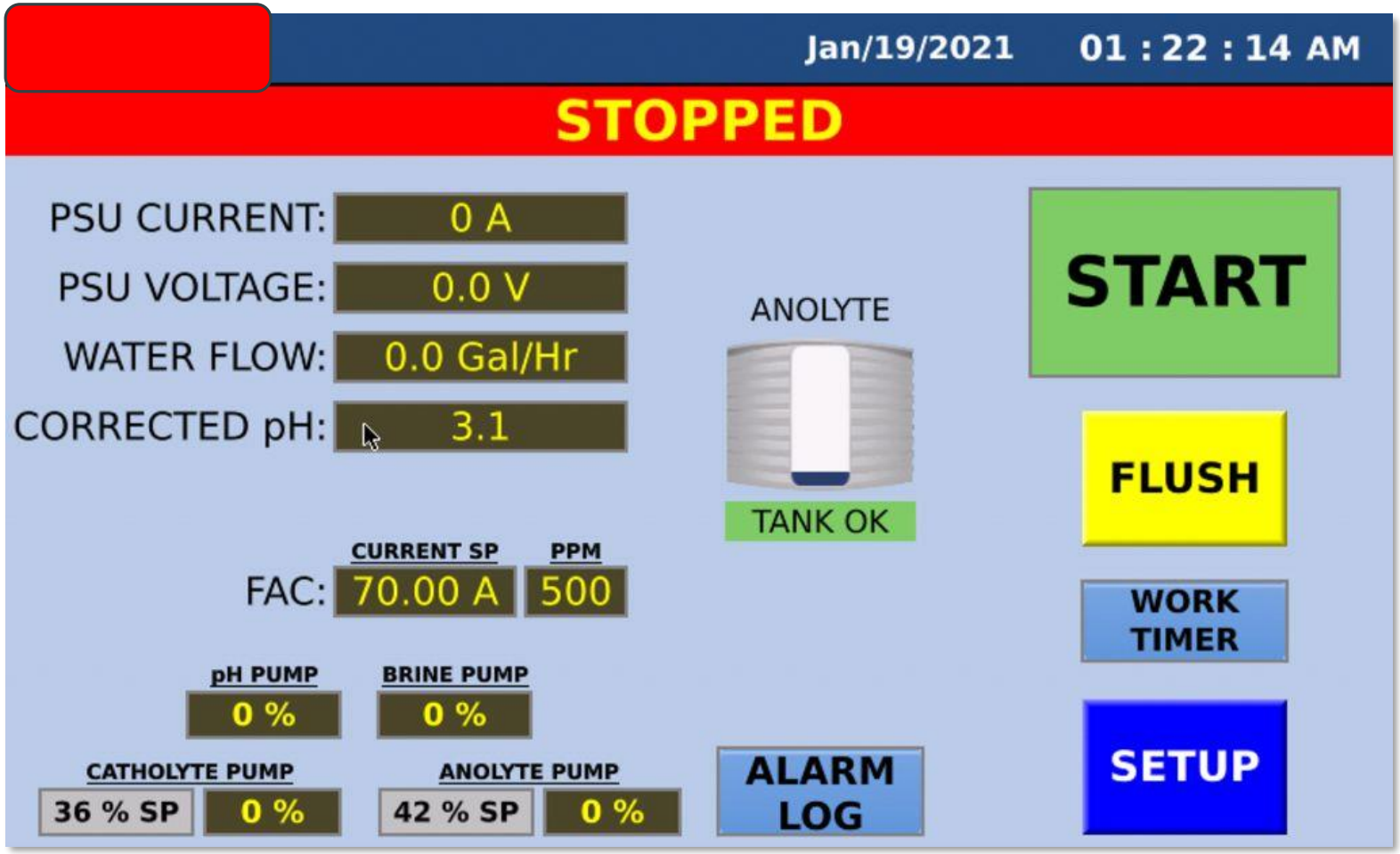
Are industrial systems any better?

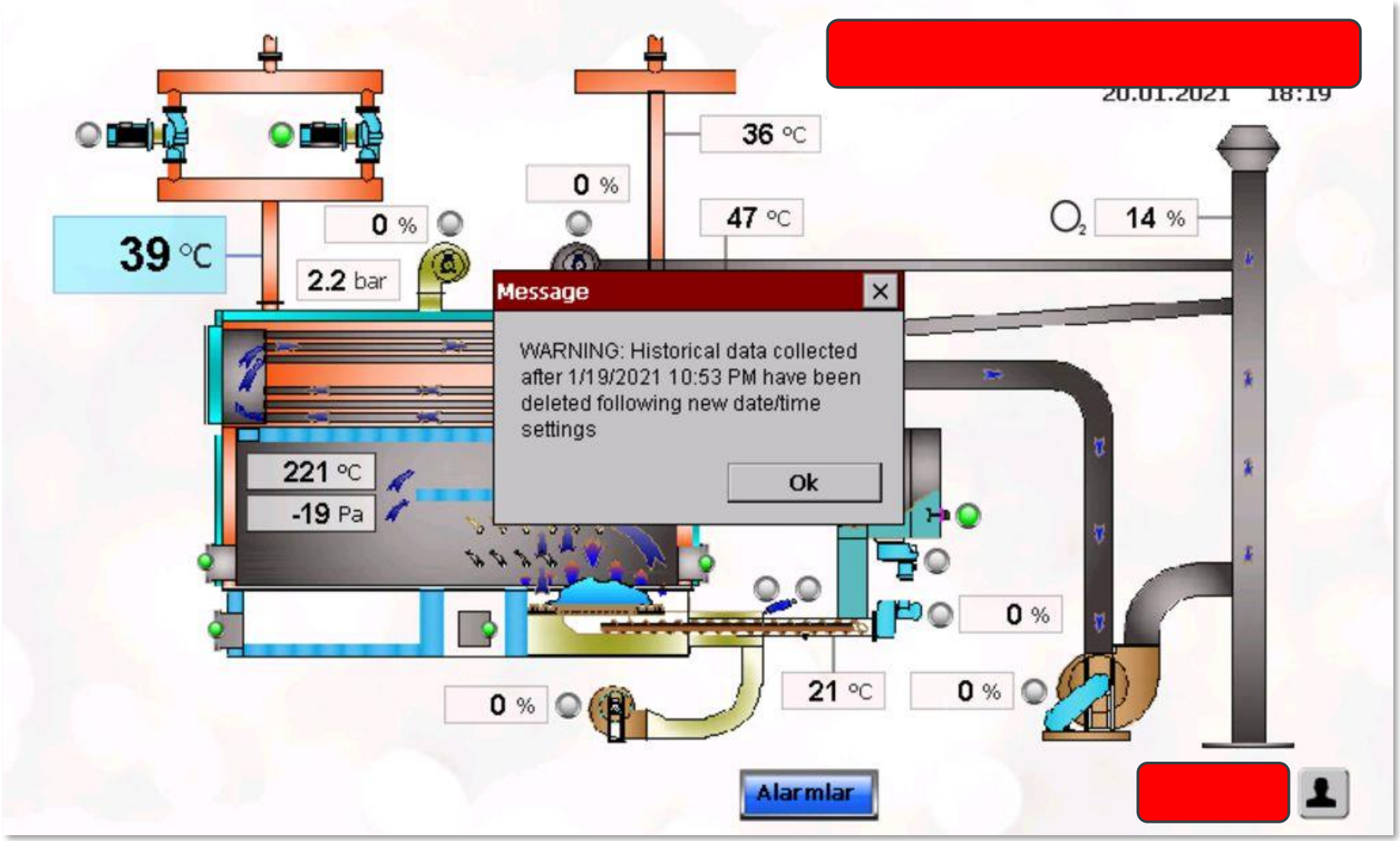
Not really.

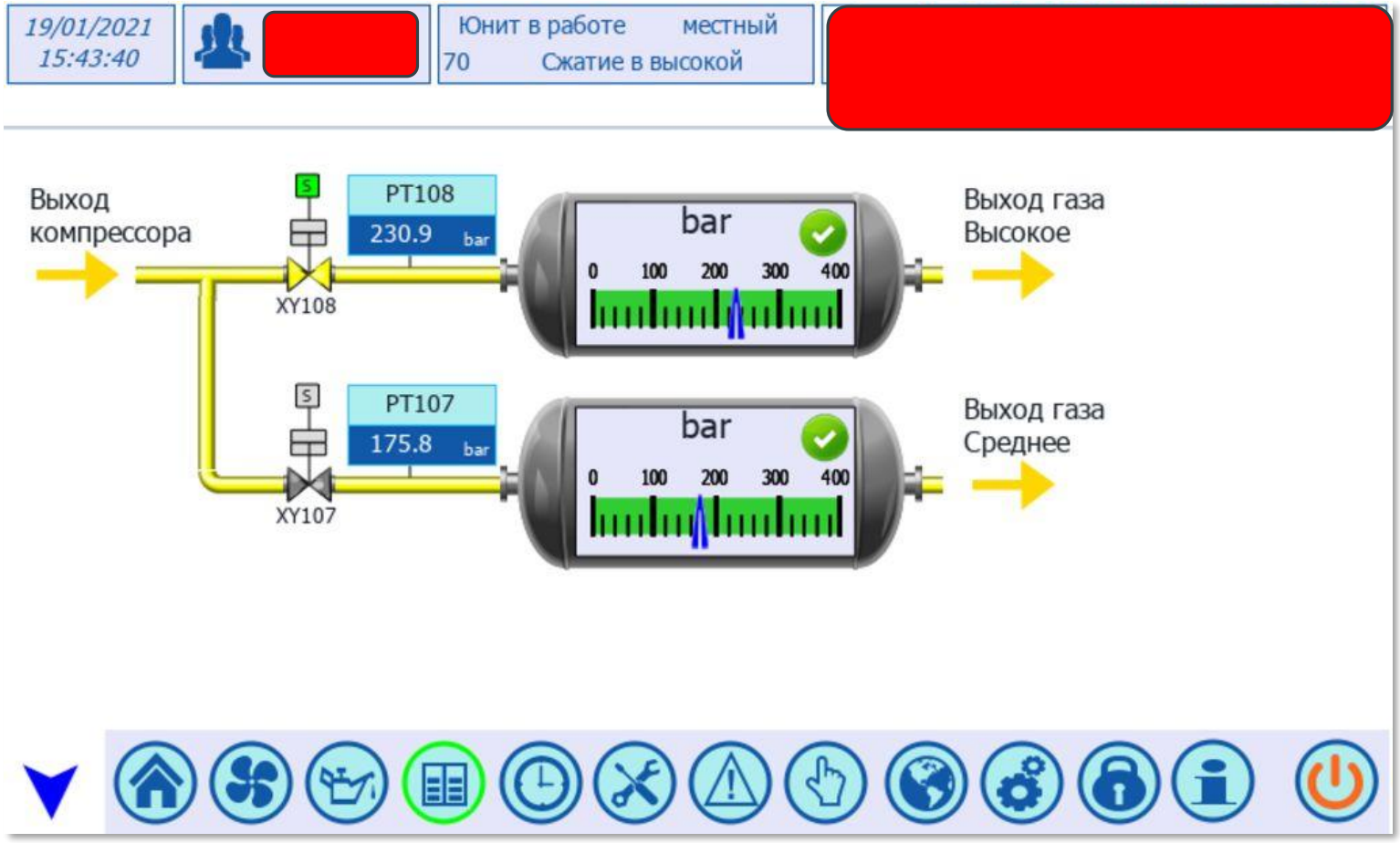


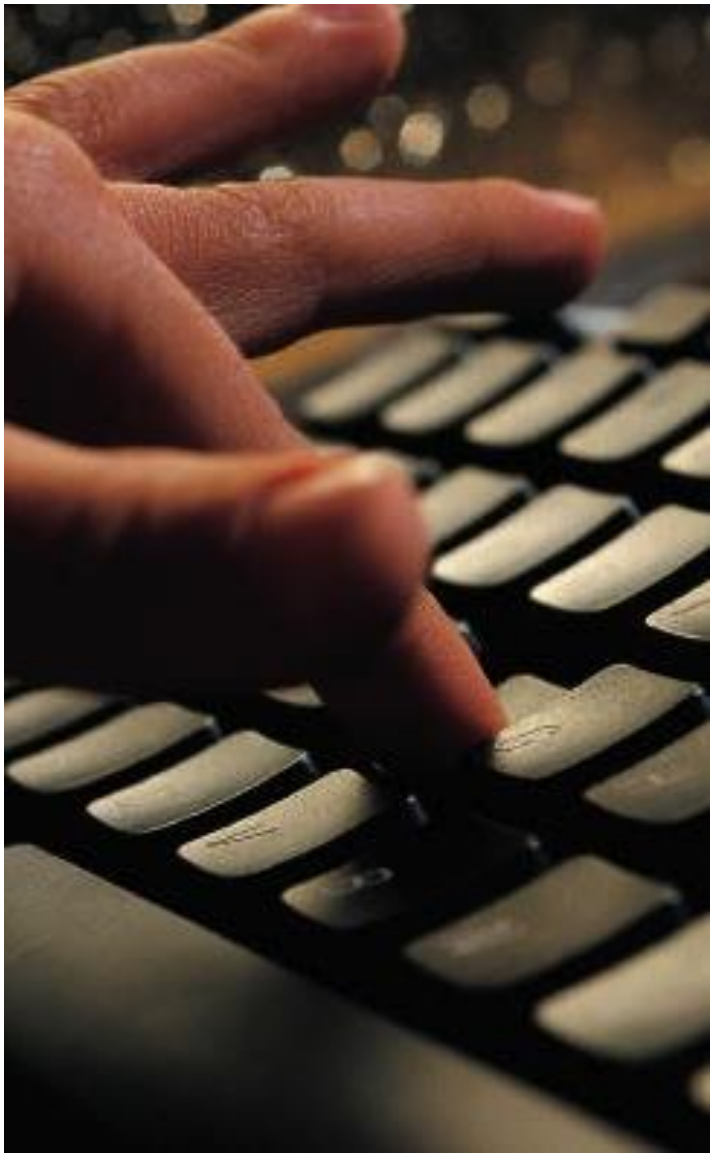

Funz: MEM
20:10:17
MEM ...78/00701-78-A--3PZ.nc N1
X43.263 Y1.227 R18.1;
G1 X-41.309 Y32.008 ;
G3 X-47.5 Y33.1 R18.1;
X-65.6 Y15.0 R18.1;
X-53.691 Y-2.008 R18.1;
G1 X30.882 Y-32.79 ;
G3 X37.072 Y-33.882 R18.1;
X55.172 Y-15.782 R18.1;
X43.263 Y1.227 R18.1;
G1 X42.098 Y1.651 ;
G3 X40.693 Y1.983 R6.2;
G1 G40 X39.719 Y-1.991 ;
G0 Z50.0;
;
M99 ;
;
;
;
;
Offset
Utensile
Lavoro
UTENSILE ATTIVO: 25
POSIZIONE REFRIGERANTE: 13
Offset utns | Lunghezza GEOMETR.(H) | Lunghezza USURA(H) | RAGGIO GEOMETR.(D) | RAGGIO USURA(D) | POSIZIONE REFRIGERANTE
19 | 87.025 | 0. | 25.022 | 0. | 13
20 | 141.933 | 0. | 25.000 | 0. | 19
21 | 88.748 | 0. | 1.500 | 0. | 13
22 | 106.329 | 0. | 2.000 | 0. | 11
23 | 0. | 0. | 0. | 0. | 0
24 | 110.175 | 0. | 3.032 | 0. | 13
25 Mandrino | 117.790 | 0. | 3.980 | 0. | 13
26 | 123.963 | 0. | 4.993 | 0. | 13
27 | 140.555 | 0. | 5.692 | 0. | 14
28 | 0. | 0. | 0. | 0. | 0
29 | 138.425 | 0. | 7.707 | 0. | 13
30 | 179.728 | 0. | 7.793 | 0. | 16
31 | 399.566 | 0. | 0. | 0. | 0
32 | 170.418 | 0. | 9.920 | 0. | 12
33 | 0. | 0. | 0. | 0. | 0
34 | 145.230 | 0. | 4.999 | 0. | 11
35 | 0. | 0. | 0. | 0. | 0
36 | 0. | 0. | 0. | 0. | 0
IMM. UN VALORE
TOOL OFFSET MEAS
CORREZIONE UTENSILI
F1
Imposta val.
ENTER
Agg. a val.
F4
OFFSET PEZZO
Mandr princ
Reg man
Avanz: 100%
Mandrino: 100%
Rapido: 50%
Vel mandr: 4249 RPM
Spindle Power: 1.6 KW
Vel superf: 106 Mpm
Caric truc: 0.01377 MMPT
Vel avanz: 300.0000 MMPM
Avanz attivo: 300.0000 MMPM
Mandrino Carico(%) 0%
Posizioni
PROGRAMMA G54 G43 H...
(MM)
Car
X -61.126 6%
Y 11.300 12%
Z -18.000 45%
Timer e contatori
Questo ciclo: 0:08:34
Ultimo ciclo: 0:19:59
RIMANENTE 0:11:24
M30 Contatore #1: 197413
M30 Contatore #2: 197413
Loop restanti: 0
DIAMETRO 204.07500
ANGOLO NAN
Setup
In funz
Avanz
Luce alta
REFRIG. ON
Input:


환경
양액
조명
설정
엽채류
현재 양액 자동모드입니다..
양액제조
START
양액 대기 중
양액 A Tank
양액 B Tank
양액 C Tank
MM
A라인
B라인
4층
3층
2층
1층
Water Tank
온도
22.1℃
H
M
L
Mixing Tank
EC
1.3μS/cm
PH
5.6pH
양액 회수 Tank
퇴수
양액 공급/대기시간
30초 / 263초
2021/ 1/21
4:00:33


Jan/19/2021
01 : 22 : 14 AM
STOPPED
PSU CURRENT:
0 A
PSU VOLTAGE:
0.0 V
WATER FLOW:
0.0 Gal/Hr
CORRECTED pH:
3.1
ANOLYTE
TANK OK
START
FLUSH
CURRENT SP
PPM
FAC:
70.00 A
500
WORK TIMER
pH PUMP
0 %
BRINE PUMP
0 %
CATHOLYTE PUMP
36 % SP
0 %
ANOLYTE PUMP
42 % SP
0 %
ALARM LOG
SETUP


20.01.2021 18:19
36 °C
0 %
47 °C
O2 14 %
39 °C
0 %
2.2 bar
Message
WARNING: Historical data collected after 1/19/2021 10:53 PM have been deleted following new date/time settings
Ok
221 °C
-19 Pa
0 %
21 °C
0 %
0 %
Alarmlar


19/01/2021
15:43:40
Юнит в работе
местный
70
Сжатие в высокой
Выход
компрессора
PT108
230.9
bar
XY108
bar
0
100
200
300
400
Выход газа
Высокое
PT107
175.8
bar
XY107
bar
0
100
200
300
400
Выход газа
Среднее



Introduction to Cryptography

Confidentiality, Integrity, Authentication, Non-Repudiation

Cryptography

Cryptography literally means "secret writing"

- Cryptography is an old field: existed far before computers, possibly over 2000 years
- Capable of much more than keeping data secret; its main use is in protecting **stored** or **transmitted** data. Indispensable tool in security

Cryptography is a huge field

- We will focus on the concepts and key attributes of commonly used cryptographic algorithms, without dealing with analysis of strength, etc.

Uses of Cryptography

Cryptography helps establish four properties for data:

Confidentiality

- Secrecy of the data
- This is provided by algorithms called **ciphers**

Integrity

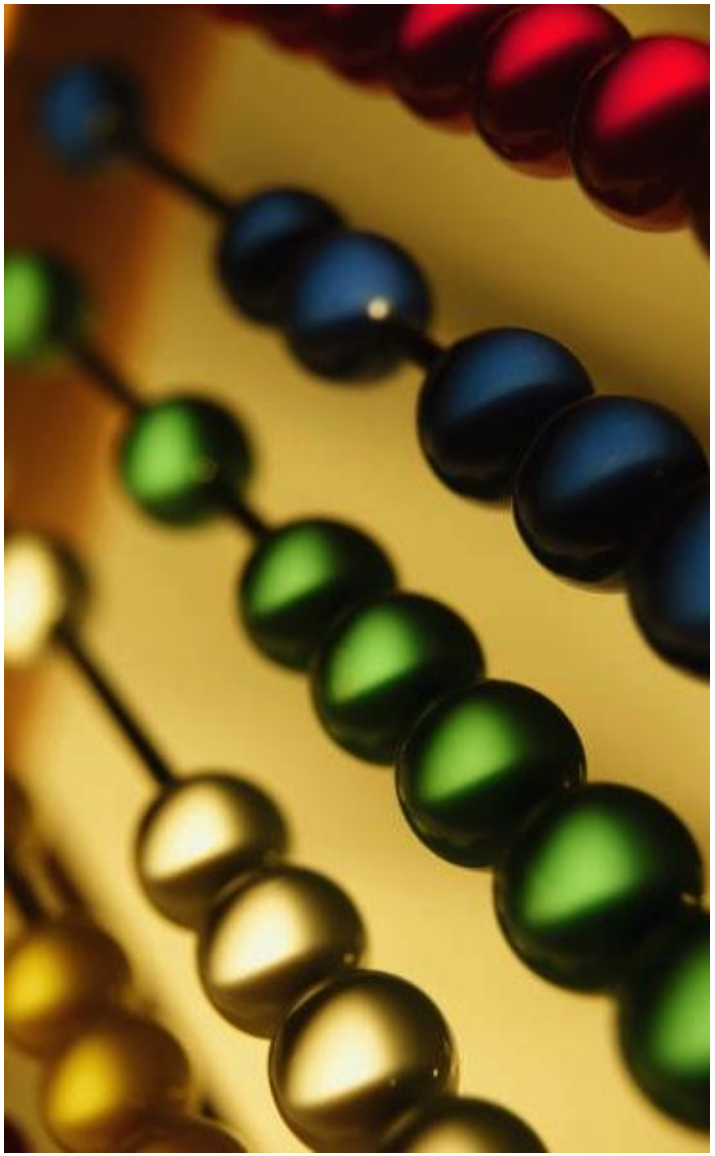
- The trustworthiness of the data
- Provided by **hashes**

Authentication

- Allows a principal (user or machine) to prove their identity, or the origin of a piece of data
- Provided by **signatures** and **Message Authentication Codes** (MACs)

Non-repudiation

- Prevents a principal from denying they performed an action
- Achieved with the help of a **trusted third party**

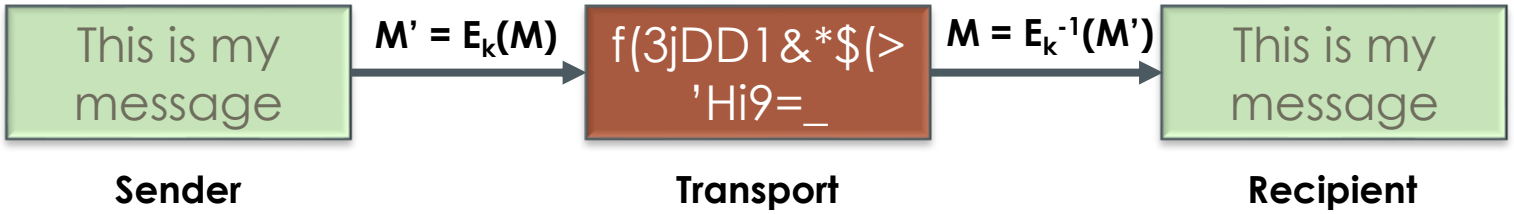


Basic Ciphers

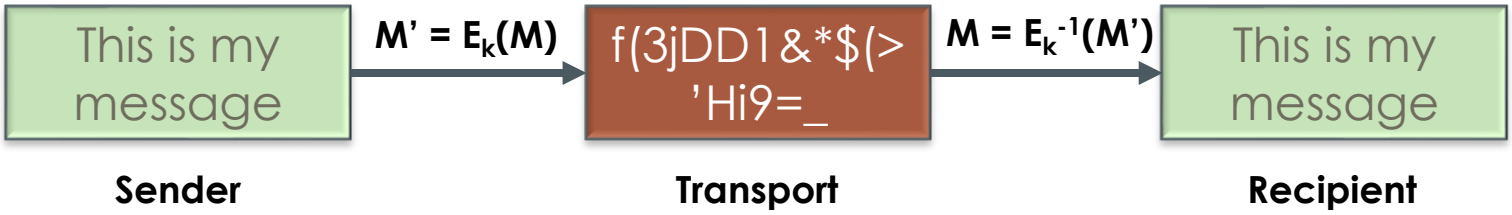
Kerckhoffs' principle, substitution ciphers, polyalphabetic ciphers, Vernam ciphers

Ciphers

A **cipher** is an algorithm that obfuscates information so that it seems random to anyone who does not possess special information called a **key**



Ciphers



Ciphers are based on a class of functions called **trapdoor one-way** functions:

- A **one-way** function is a function that is easy to compute, but whose inverse is difficult to compute
- The **trapdoor** means that, given special information (the key), the inverse becomes easy to compute

Ciphers

Interesting fact: It has never been proven (or disproven) that one-way functions exist.

- Ciphers are based on functions that are believed to be one-way because no one has ever shown an easy way of computing their inverse
- A formal proof that a one-way function exists would also prove that P != NP

Whichever one-way function we choose, the **function itself** must not be the critical secret…

Kerckhoff's Principle

The security of any given encryption system must depend only on the secrecy of the key, **K**, and not on the secrecy of the algorithm

- Algorithms are hard to change: compiled into software, wired into circuits
- Need to be able to rely on using an algorithm for a long time
- Algorithms are easily disclosed: must be distributed in every program / device that might exchange information

Mifare Hack

Mifare RFID smartcards are used in a variety of critical applications (access control, dozens of major public transit systems, etc.)

- Trusted for audit, fare payment, etc.
- In 2008, a team of Dutch researchers reverse-engineered the proprietary encryption used for transmitting data, and hacked the cards

http://www.ru.nl/ds/research/rfid/

Mifare Hack

"The security of Mifare Classic is terrible. This is not an exaggeration; it's kindergarten cryptography. Anyone with any security experience would be embarrassed to put his name to the design. [Mifare] attempted to deal with this embarrassment by keeping the design secret."

- Bruce Schneier

Manufacturer sued to keep the research secret; Dutch court ruled that the potential "damage to [Mifare] is not the result of the publication of the article, but of the production and sale of a chip that appears to have shortcomings."

Ciphers

Two good, well-researched candidates for one-way functions are **factoring** and **discrete log:**

- **Factoring**

  Suppose $z = (x \bullet y)$

  Given **z**, find **x** and **y**.

- **Discrete log**

  Suppose $z = (x^y \bmod m)$

  Given **z**, **x** and **m**, find **y** = $(\log_x z) \bmod m$

Caesar (Shift) Cipher

When Caesar mounted his campaign against the Gauls (modern France), he wanted to communicate with his troops securely

- He contrived a very simple cipher:
  - Take each letter, and replace it with the letter shifted 3 letters to the right in the alphabet
  - If there are no more letters, wrap around to the beginning of the alphabet
  - This is similar to modern day **rot13** used for simple obfuscation
  - Decryption is just the inverse
- This type of cipher is also called a **shift** cipher

Substitution Ciphers

The shift cipher is an instance of a class of ciphers called **substitution** ciphers

- Each plaintext letter is replaced with exactly one ciphertext letter
- The key is the mapping between plaintext letters and ciphertext letters

Example: Assume a five letter alphabet {ABCDE} and a shift of 2: {DEABC}

A=D, B=E, C=A, D=B, E=C

Attacks on Ciphers

When attacking a cipher, may be several goals:

- Get the plaintext corresponding to a ciphertext
- Get the key (and possibly the algorithm if it's not public)

Brute-force attack

- Try all possible keys on some ciphertext until output is an intelligible plaintext

Cryptanalysis

- Aim is to do better than brute-force attack
- Relies on nature or characteristics of the algorithm
- Some knowledge of plaintext characteristics
- May use samples of plaintext-ciphertext pairs

Brute-Force Attack

Key Size	Number of Possible Keys	Time Required at 10^6 tests/sec	Time Required at 10^{12} tests/sec
32 bits	$2^{32} = 4.3 \cdot 10^9$	36 minutes	2.2 ms
56 bits	$2^{56} = 7.2 \cdot 10^{16}$	1142 years	10 hours
128 bits	$2^{128} = 3.4 \cdot 10^{38}$	$5.4 \cdot 10^{24}$ years	$5.4 \cdot 10^{18}$ years
168 bits	$2^{168} = 3.7 \cdot 10^{50}$	$5.9 \cdot 10^{36}$ years	$5.9 \cdot 10^{30}$ years
26 characters (permutation)	$26! = 4 \times 10^{26}$	6.4×10^{12} years	$6.4 \cdot 10^6$ years

Cryptanalysis

When analyzing ciphers, cryptographers categorize the strength of a cipher against several types of attacks (ranked from hardest to easiest):

Ciphertext-Only

- Adversary only has ciphertexts (encrypted messages)

Known-Plaintext

- Adversary has some number of plaintext and ciphertext pairs
- The more pairs it takes to break cipher, the stronger it is

Chosen-Plaintext / Chosen-Ciphertext

- Adversary can pick a plaintext and get corresponding ciphertext or vice-versa
- Adversary can **adaptively** select plaintexts or ciphertexts that help break the cipher

How strong is a substitution cipher?

Breaking Substitution Ciphers

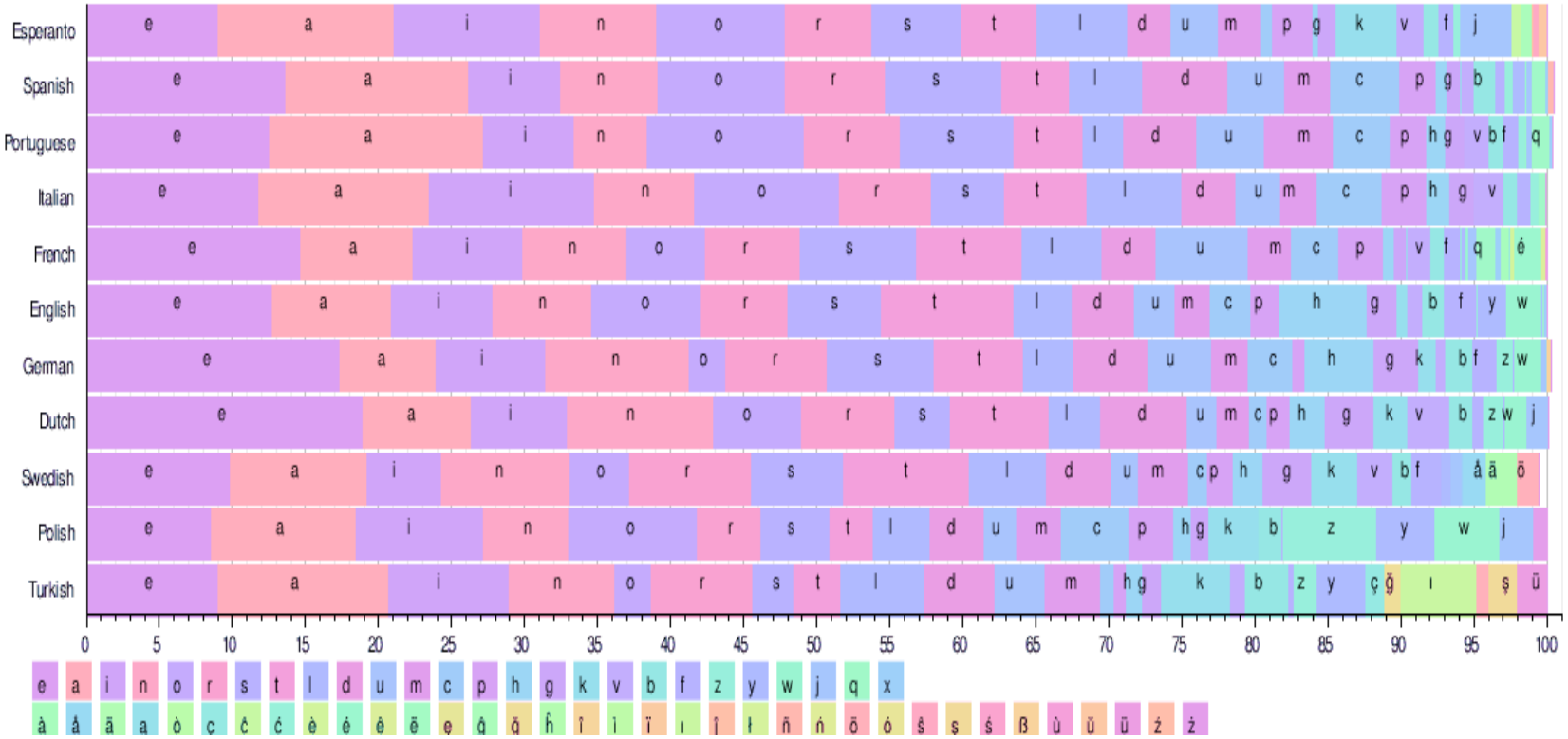
These ciphers are very easy to break with analysis:

- The weakness is that every letter in the plaintext alphabet always gets encrypted to the **same** letter in the ciphertext alphabet

If the attacker knows the original message is in English:

- Probability distribution: **E**=13%, **T**=9.3%, **A**=7.3%, …
- Most common digraphs: **TH**, **HE**, **IN**, …
- Attacker can perform **frequency analysis** on the ciphertext to identify and decode common letters, then match against common English words to recover the plaintext and eventually the key
- These attacks are easy enough to be done by hand

Breaking Substitution Ciphers



Improving Substitution Ciphers

Substitution ciphers do not hide frequency information because every plaintext letter always encrypted to the **same** ciphertext letter

- One way to improve the cipher is to use a **Polyalphabetic Cipher**
  - Instead of having one mapping, have a set of n mappings, and change the mapping with every character
  - When mappings are used up, then repeat with the first one

Polyalphabetic Ciphers

Alphabet:	ABCDE
Key #1:	BEDAC
Key #2:	ACBDE
Key #3:	DACBE

E(“BED”) = EEB

E(“ABACADA”) = BCDDABB

Polyalphabetic Ciphers

Because of this repetition, these poly-alphabetic ciphers are sometimes called periodic ciphers

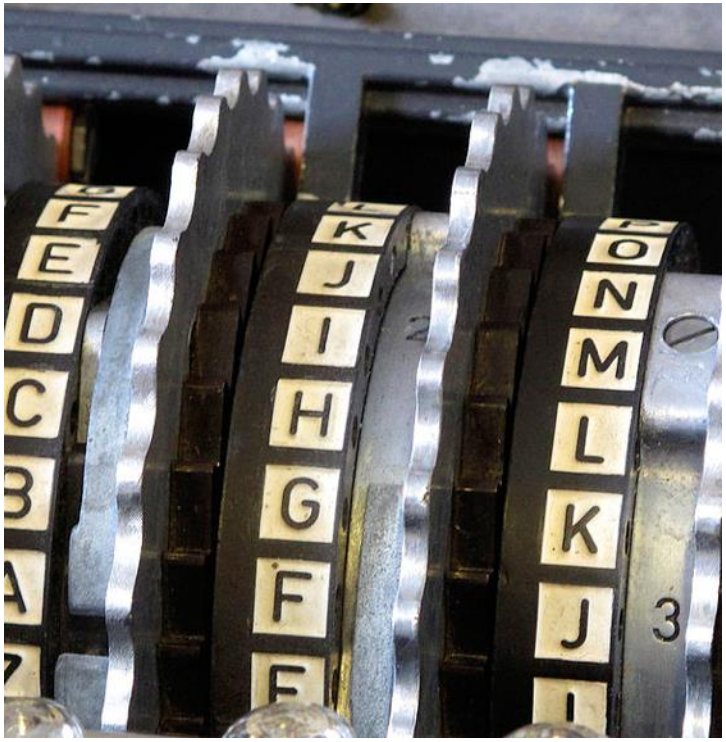
- If the attacker knows, or can somehow guess the period, an attack is possible
- For small **n** (number of keys), only incrementally harder than a plain substitution cipher, but for large **n**, much more difficult
- The attack requires more ciphertext examples, but becomes easier if the adversary has some plaintext/ciphertext pairs

Enigma Machine

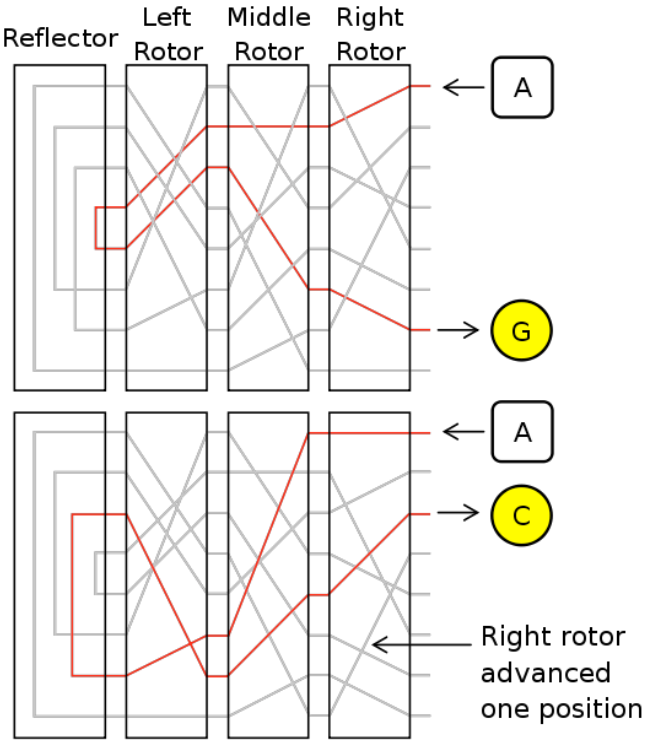
Famous early large-scale mechanization of secrecy

- Series of rotors produce a polyalphabetic cipher with very large **n**
- Permutation mapping of alphabet that changes with each keystroke

Enigma Machine



Source: Wikimedia Commons



One-Time Pad or Vernam Cipher

A **One-Time Pad** (OTP), also called the **Vernam Cipher** after its creator, is a special type of polyalphabetic cipher that never repeats

- A random substitution is used for every character
- Think about it as using an infinite number of keys

One-Time Pad

- The cipher requires the key to be the same length as the message to be encrypted
- The ciphertext is created by computing the bitwise XOR of the plaintext and the key at the binary level
  - A plaintext bit is flipped when key bit is 1
  - A plaintext bit remains the same when key bit is 0

One-Time Pad

XOR

	0	1
0	0	1
1	1	0

Plaintext: 0101

Key: 1001

Encrypted: 1100

Properties of One-Time Pad

If the key is well chosen (*i.e.*, random), the ciphertext is the plaintext with randomly flipped bits

- For a message containing **n** bits of information, OTP adds exactly **n** bits of randomness, creating a completely random ciphertext
- **Theoretically unbreakable**

Disadvantages of One-Time Pad

- Key length = message length
  - Key overhead of 100% is generally not acceptable
- Each key can be used **once** (hence the name)
  - Key must be sent separately, for every message sent (serious problem for distribution infrastructure)
  - Synchronization problem if messages are lost or reordered
  - If any key is used to encrypt more than one message, security is reduced significantly
  - Impractical for most applications
- Cipher is **malleable**
  - Bit flip in ciphertext, flips only one bit in plaintext
  - Requires combining with integrity check to avoid tampering
- Need a good source of randomness for the key

Strength of One Time Pad

How strong is OTP against the following attacks?

- **Ciphertext-only**
  - Proven to be information theoretically secure
  - If used properly, impossible to break
- **Known-Plaintext**
  - Very weak
  - Just XOR CT with PT to reveal the key
  - Only need one pair
  - Of course, key is not supposed to repeat
- **Chosen-Ciphertext/Plaintext**
  - Since it's weak against a Known-CT/PT attack, it is weak against this attack

Stream Ciphers vs. Block Ciphers

Practical Ciphers

- Fixed length keys that are much shorter than the message
  - Do not depend on message length
- Efficient for encryption and decryption
- Ciphertexts should be computationally difficult to decrypt without the key
  - Note: "computationally difficult" is a moving target, as computers are getting more and more powerful
- Two type of ciphers
  - Symmetric key
  - Public (assymmetric) key

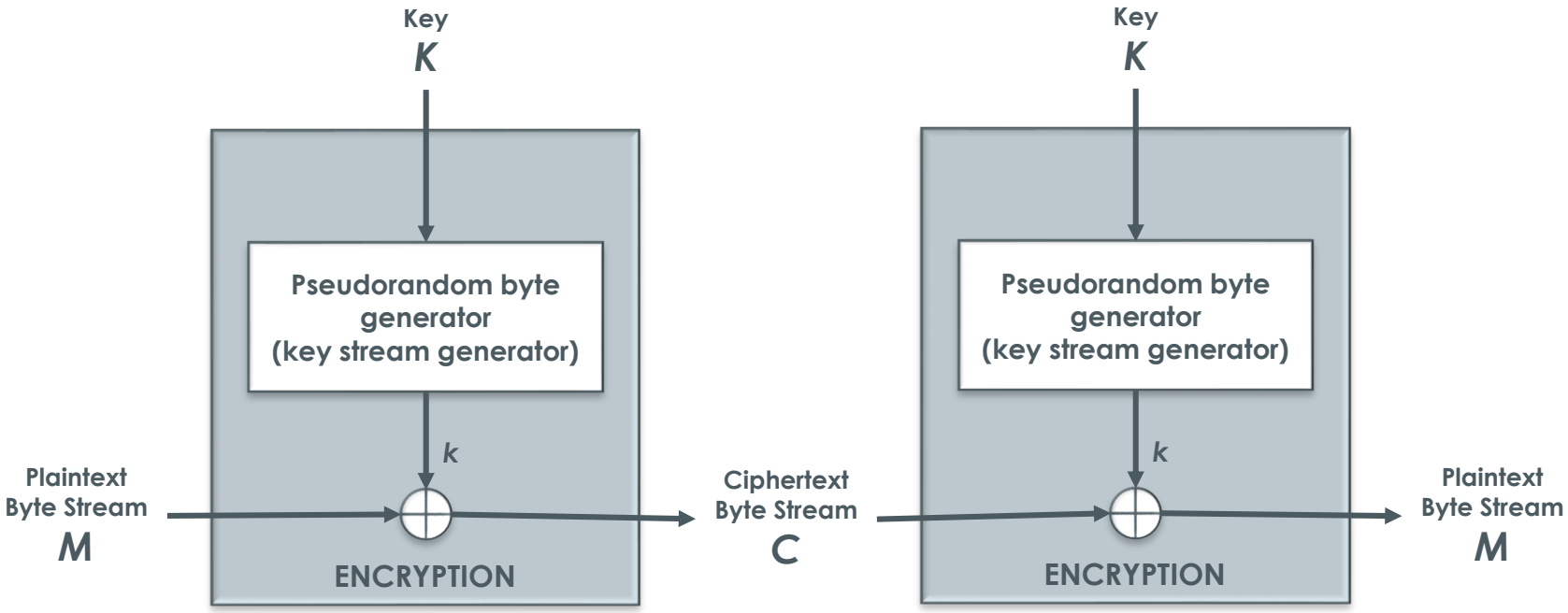
Symmetric Key Ciphers

- Symmetric key ciphers use the same key to encrypt and decrypt data
- There are two types of symmetric key ciphers
  - Stream Ciphers
  - Block Ciphers

Stream Ciphers

- These are similar to OTP's
  - Instead of truly random key bits, a key is used to generate a pseudo-random sequence of bits
  - The bits are then XOR'ed with the plaintext
- Plaintext is encrypted a bit at a time, making it useful for streaming applications
  - *e.g.*, voice or video
- These ciphers suffer from synchronization problems, if any bits are lost, the entire stream may be corrupted

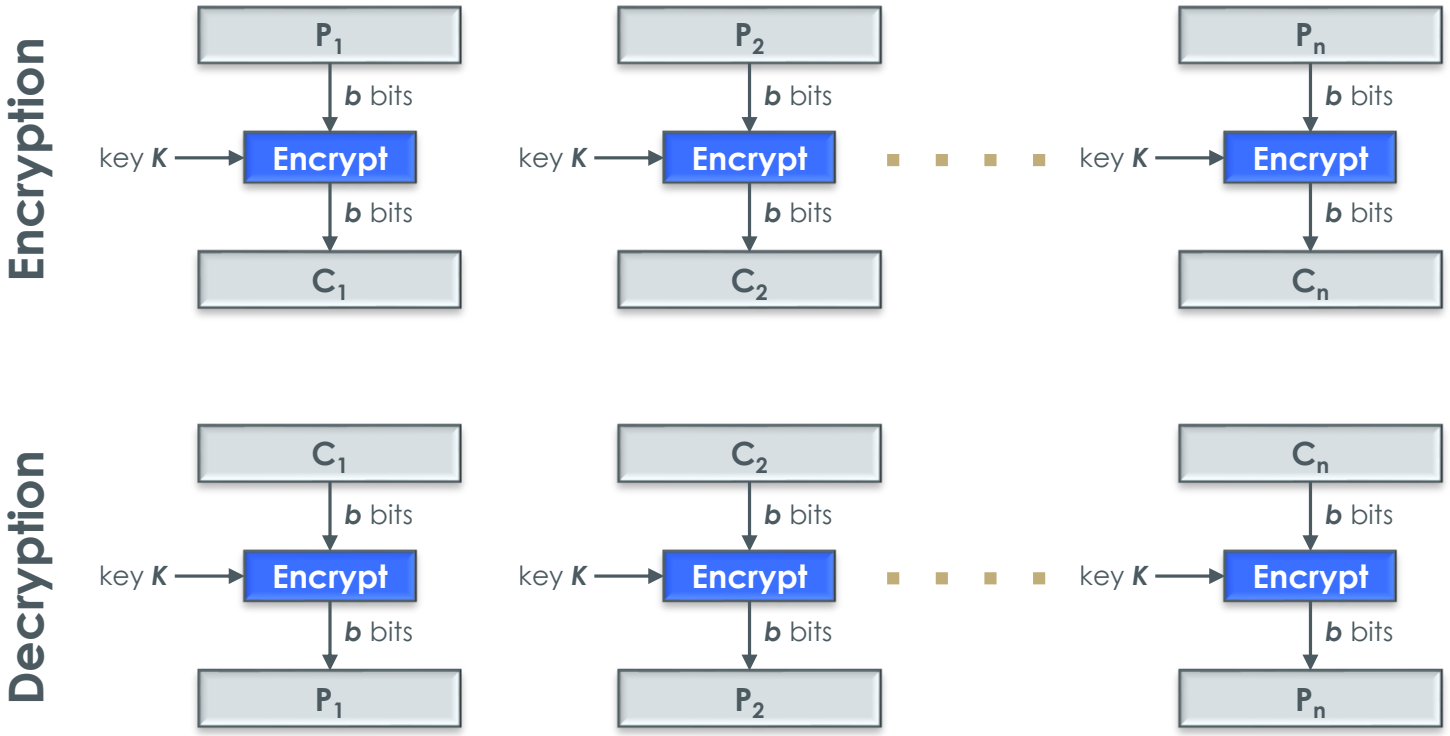
Stream Ciphers



Block Ciphers

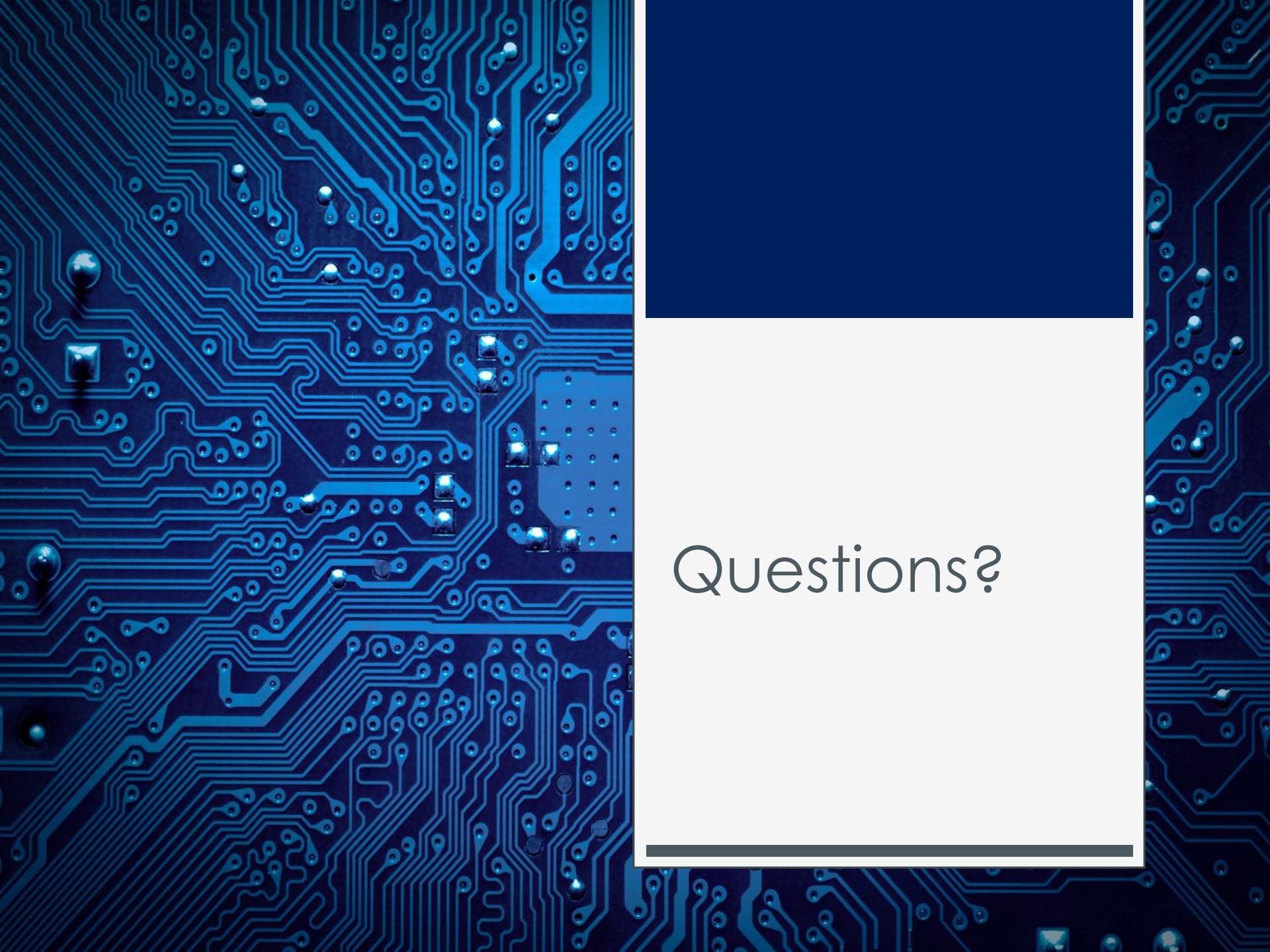
- Encrypt a block of plaintext at a time
- Usually 64 bits or a multiple
- Plaintext is divided into blocks and each is encrypted separately
- The last block might need to be “padded” to make it a full block length

Block Ciphers



Stream Ciphers vs. Block Ciphers

- Stream ciphers are generally simple and very fast
- Block ciphers are more common than stream ciphers
  - Unfortunately, the reasons are not entirely logical
  - In the past, most stream ciphers were proprietary, so they could not be analyzed, and therefore people could not be confident of their security
  - In contrast, there are many publicly available and well-studied block ciphers


Questions?