



MAY 11-12

BRIEFINGS

# Grand Theft House: RF Lock Pick Tool to Unlock Smart Door Lock

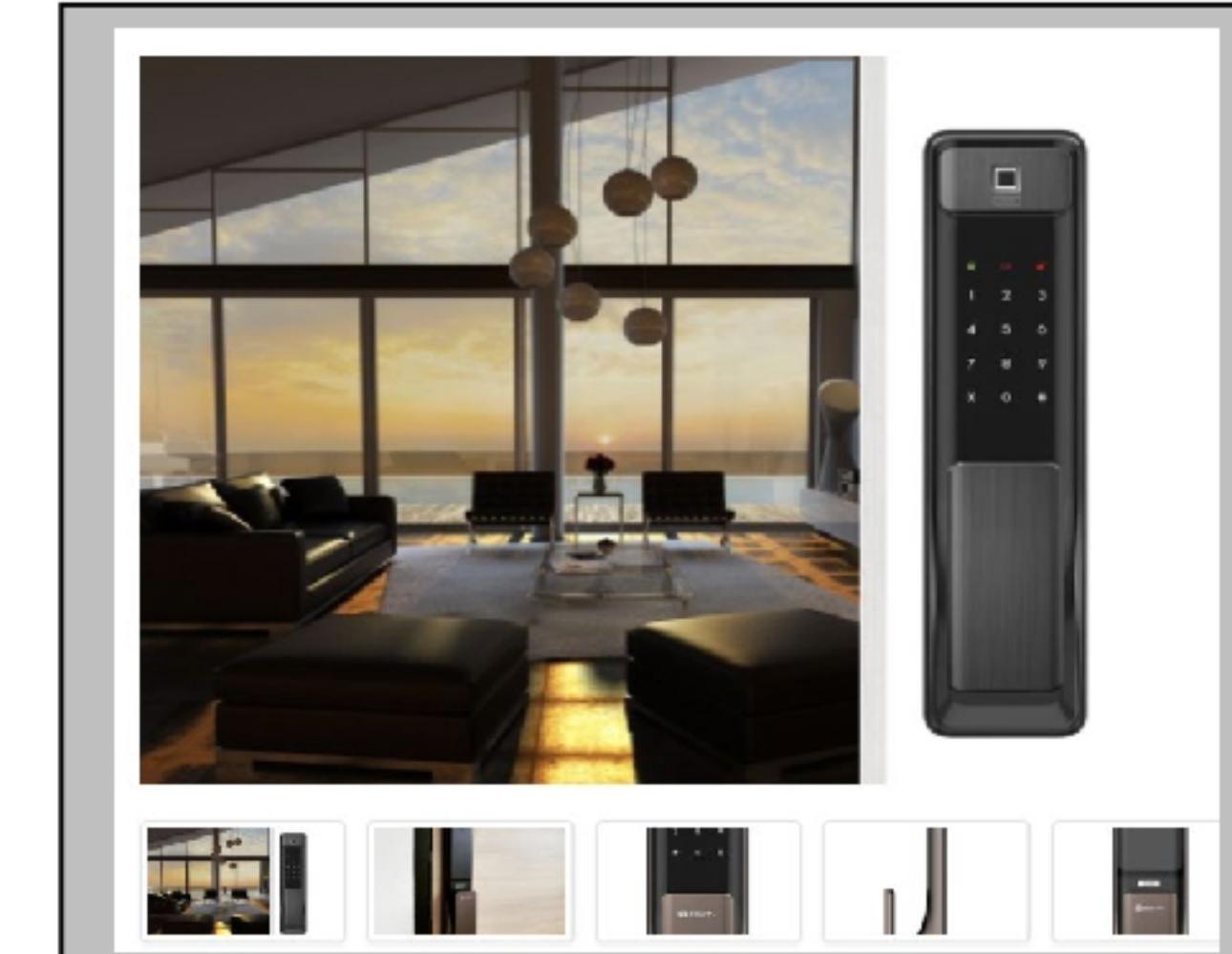
Seungjoon Lee, Kwonyoup Kim

# Prologue



## , DDL RF Module- 4 wires (Power 2 /open 2)

Model		
Transmission Method		Camera- 4 wires, CCTV- 2 wire Interphone- 4 wires , Sensor- 2
Power Source		100V-240V~, 50/60Hz
Consumption		Stand-by : 3.5 W Max : 15 W
Intercom system		HANDS FREE type (Voice swit
Monitor		7" FULL TOUCH LCD
Video subscription time		Individual door camera monitor Call connection / Calling: 60 ±5
Wiring Distance	UTP CAT 5e. Standard (Ø 0.5)	Camera / CCTV / Interphone < 28 meters (MAX) Resistance under 10Ω per 100
Operating Temp.		0 ~ +40 °C (32°F ~ 104°F)



Registration Status (Korea National Radio Research Agency)

Application	'17	'18	19	tot.
447MHz Module		2	2	4
alarm detector	1	6	-	7
<b>Digital Doorlock</b>	<b>23</b>	<b>16</b>	<b>18</b>	<b>57</b>
Guidance for disabilities	10	8	14	32
Thermal Detector	5	3	-	8



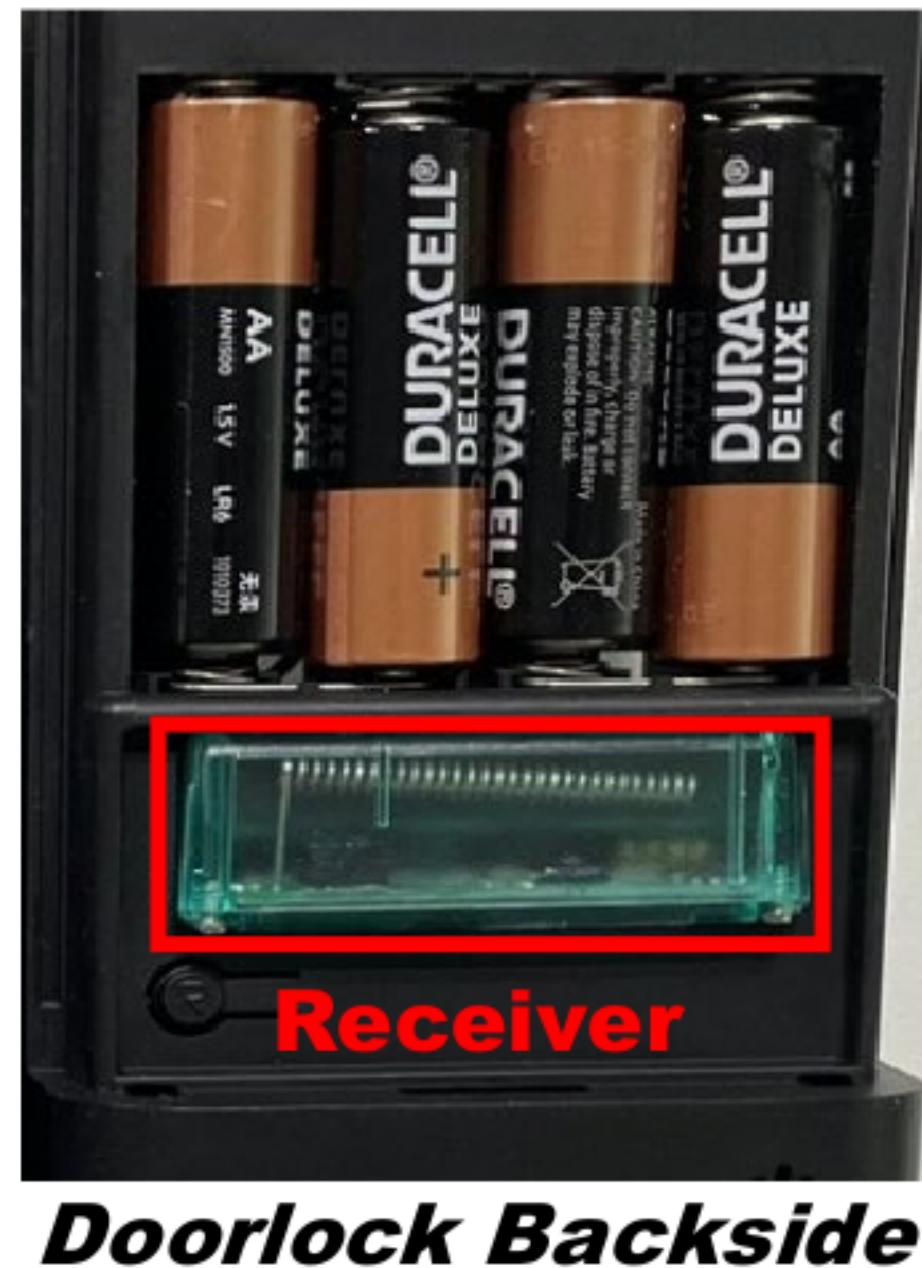
Korea Table of Frequency Allocations(Low-power wireless device for security & safety systems)

Application	Band	Power
Guidance equipment for the visually impaired	235.300MHz (Fixed)	< 10m W
	358.500MHz (Mobile)	
Transportation support system for disabilities	235.3125MHz	< 100m W
	...	
Security system, Anti-theft system, fire alarms, etc.	447.2625 MHz ~447.5625 MHz	< 10m W

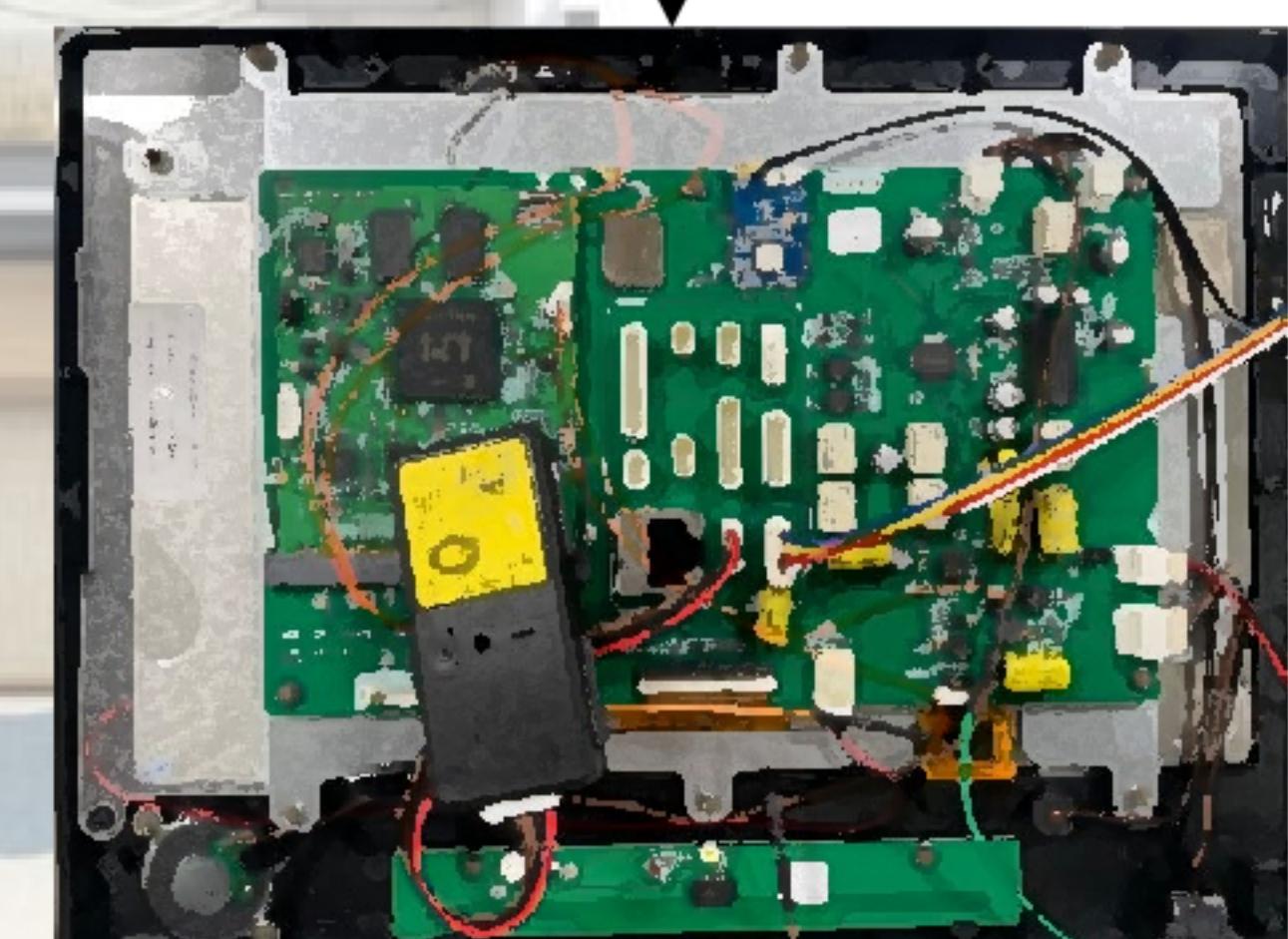
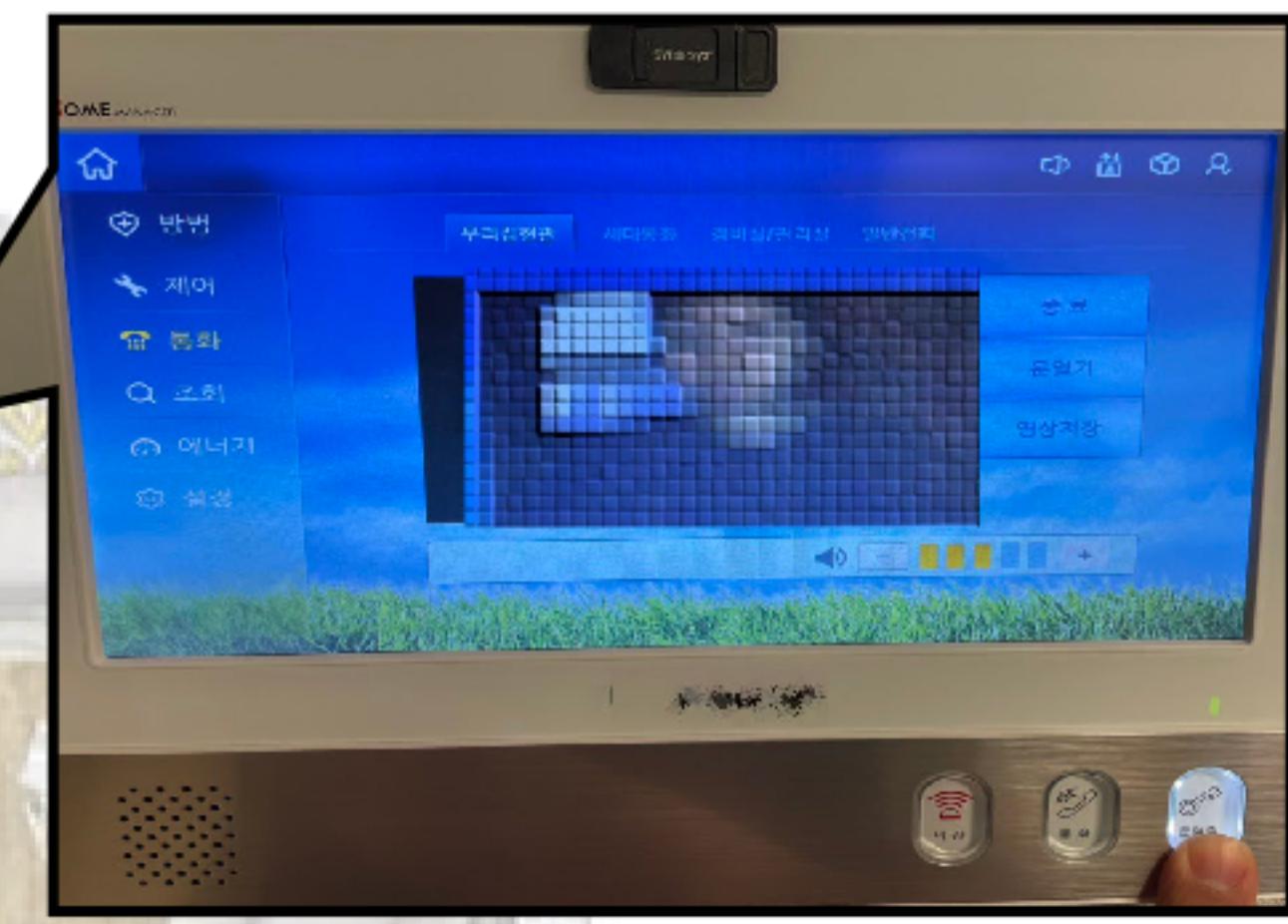
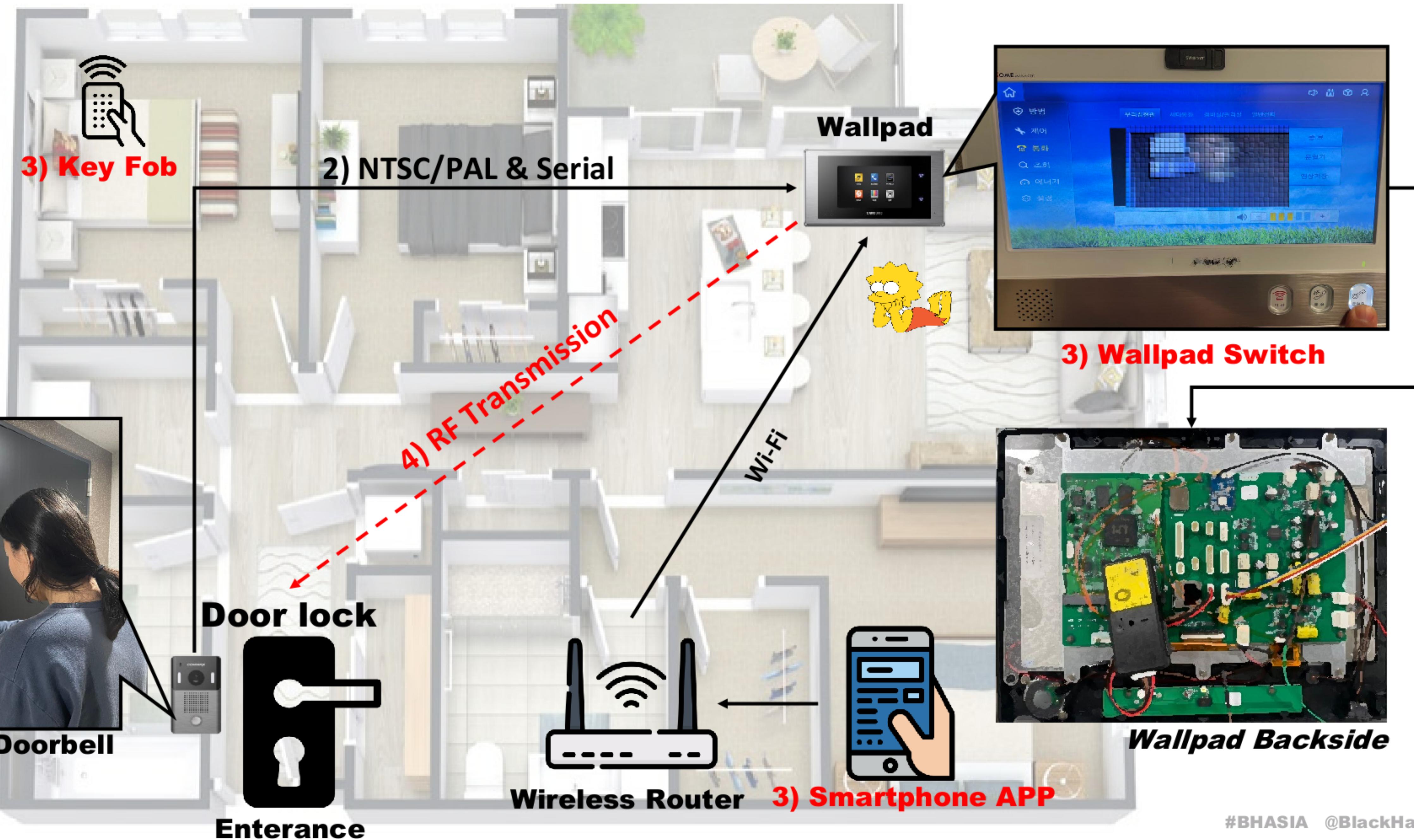
235.3	447.5625
(0.0375W)	(0.0375W)
358.5	358.5375
(0.3W)	(0.3W)

# Wireless Door Lock System

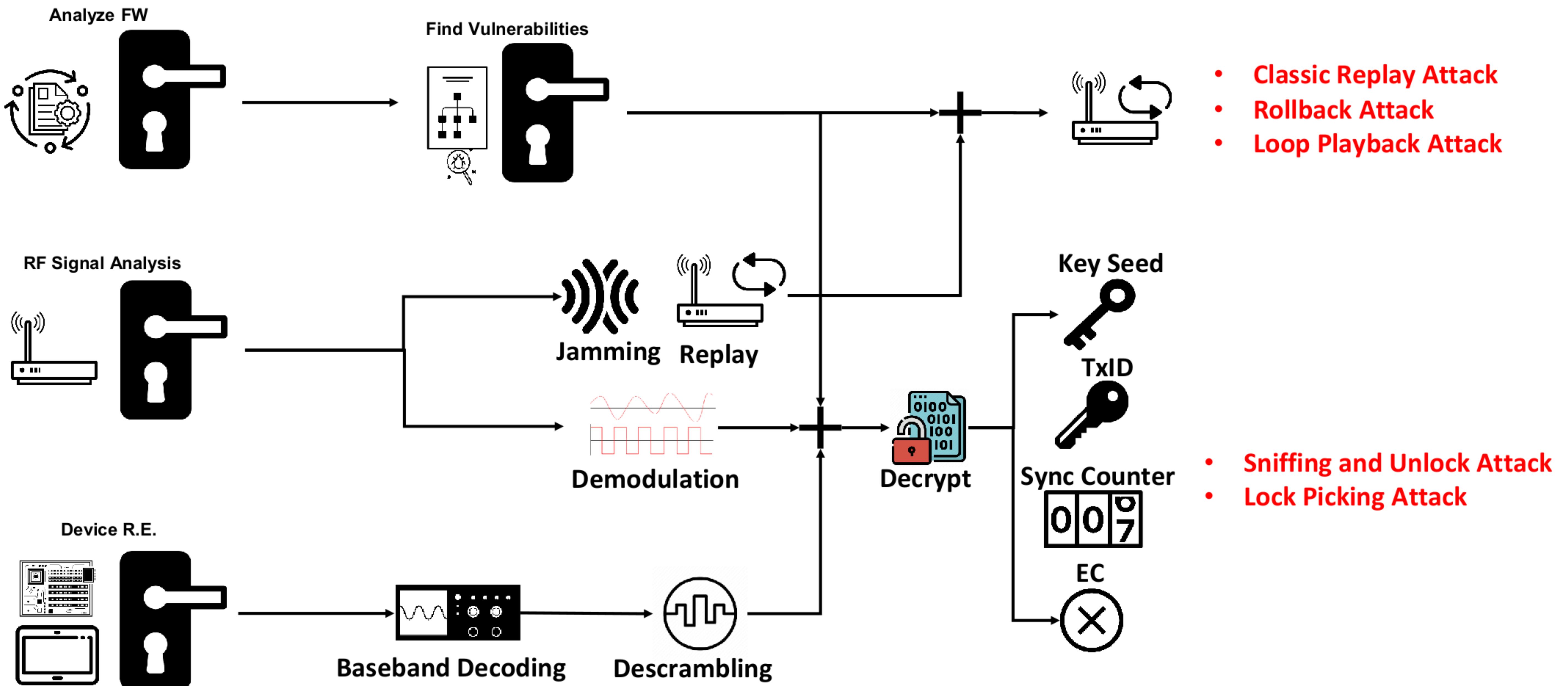
- It can be used in a variety of settings, including homes, offices, and other commercial or industrial.



**Doorlock Backside**



# Summary



- Classic Replay Attack
- Rollback Attack
- Loop Playback Attack

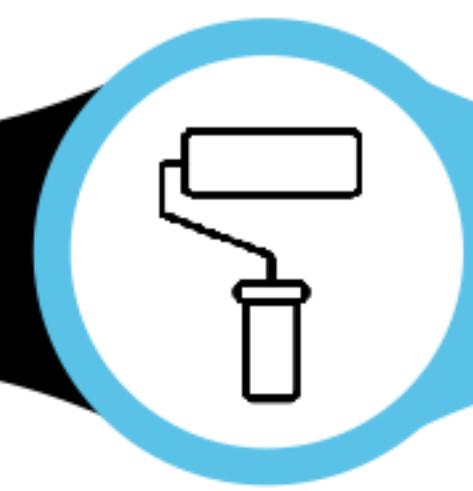
- Sniffing and Unlock Attack
- Lock Picking Attack

# Agenda



## Background

- Wireless Door Lock System
- Hardware Teardown
- Door Lock RF Signal Basic



## Rolling Code and Replay

- Rolling code nutshell
- Roll Jam/ Roll Back Attack
- Loop Back Attack
- Evaluation



## Encryption & Authentication

- Door Lock RF Packet Encryption
- Key Generation
- Packet Confusing
- TxID Matching and Authentication



## Gadget Preparation

- Direct Mode and Synchronization
- Proprietary Baseband Encoding
- Build Receiver and Transmitter



## The Art of Lockpicking

- Sniffing and Unlock Door Locks
- Resynchronization
- Force Synchronization
- Lock Picking Time Analysis

# Types of Door Lock RX

## Door Lock Mainboard MCU

- ❑ Usually use one of “16Bit RL78” or “32Bit ARM” or “8bit STM”
- ❑ usage : Ten-key pad, E-Mortise Control, Authentication

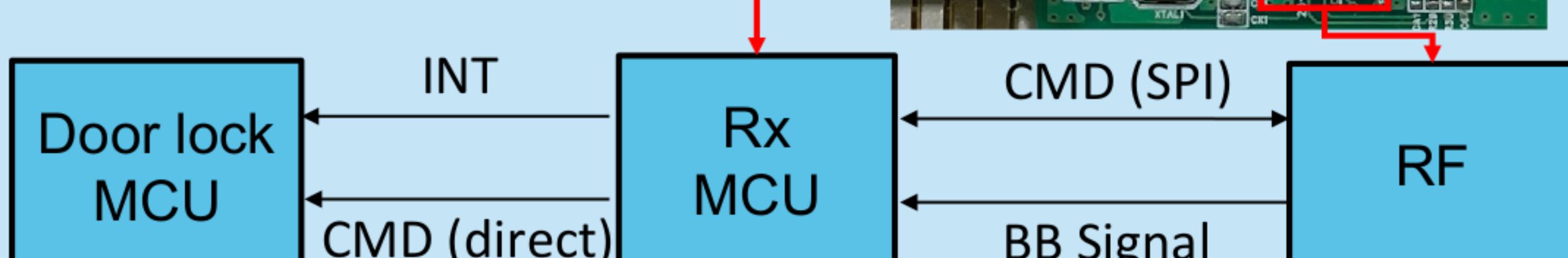
## RX MCU

- ❑ Usually use one of “8bit 8051” or “8bit PIC” or “8bit STM” or “16bit PIC”

## RX MCU

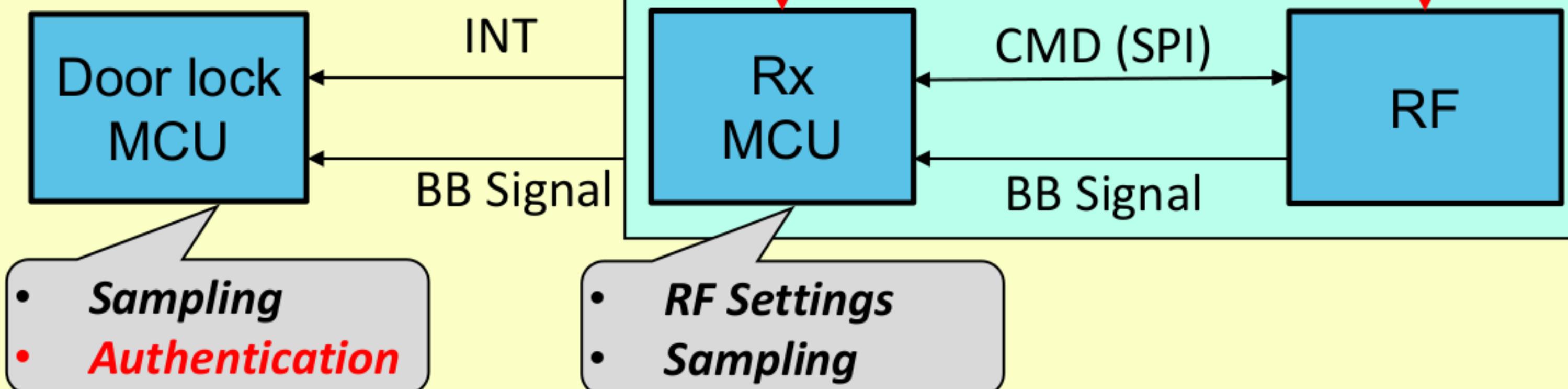
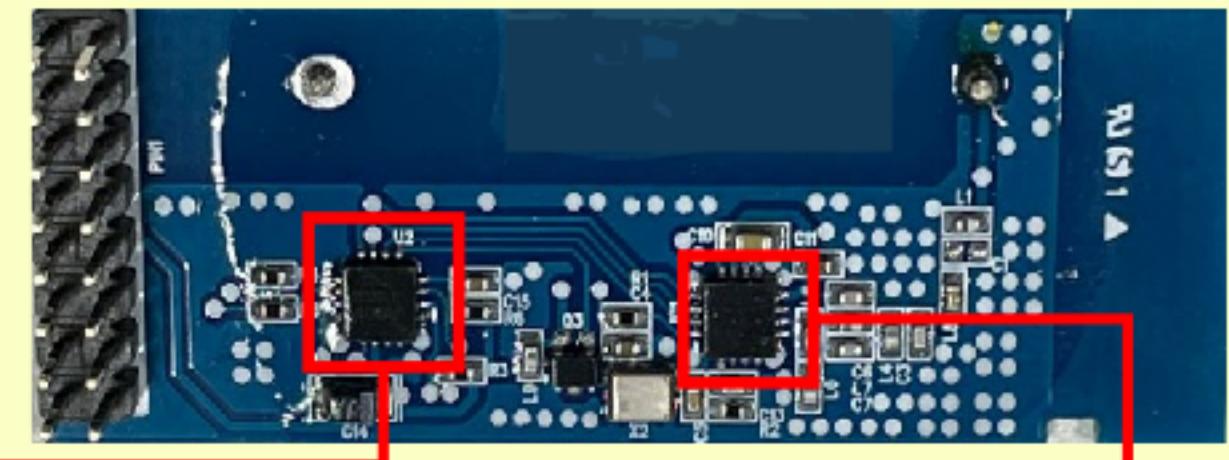
- ❑ Usually use transceiver, TH71120 /CMT 2219 / CMT 2300
- ❑ Transceiver IC used but Simplex Communication

## Receiver-based Auth. (RX MCU + RF IC Dongle)



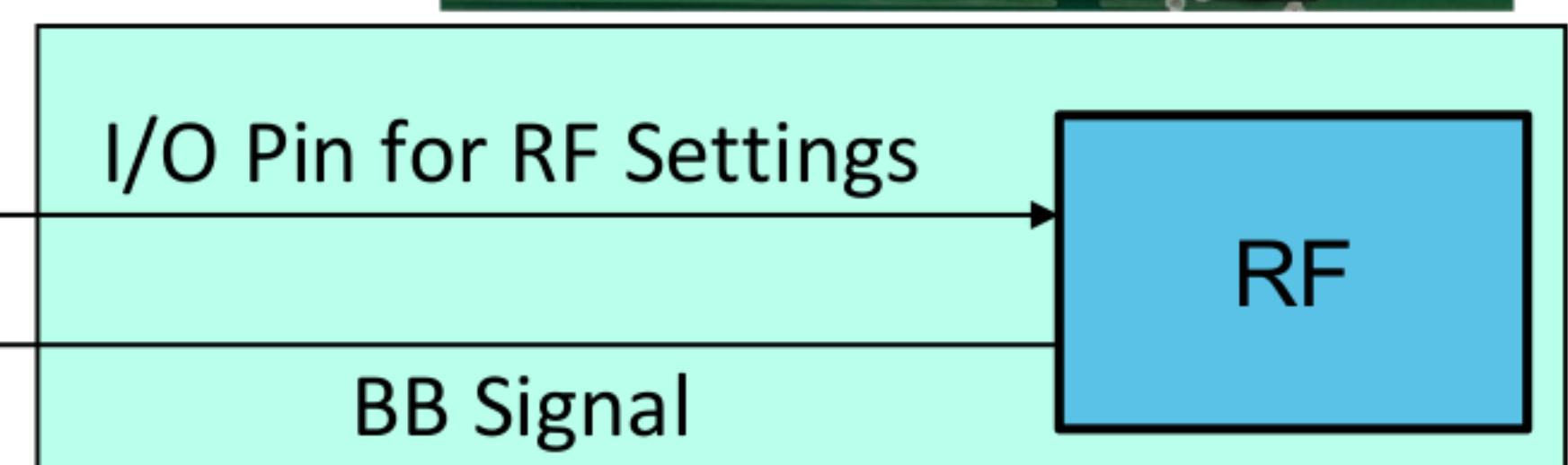
- **RF Settings**
- **Sampling**
- **Authentication**

## Mainboard-based Auth. (RX MCU + RF IC Dongle)



- **Sampling**
- **Authentication**
- **RF Settings**
- **Sampling**

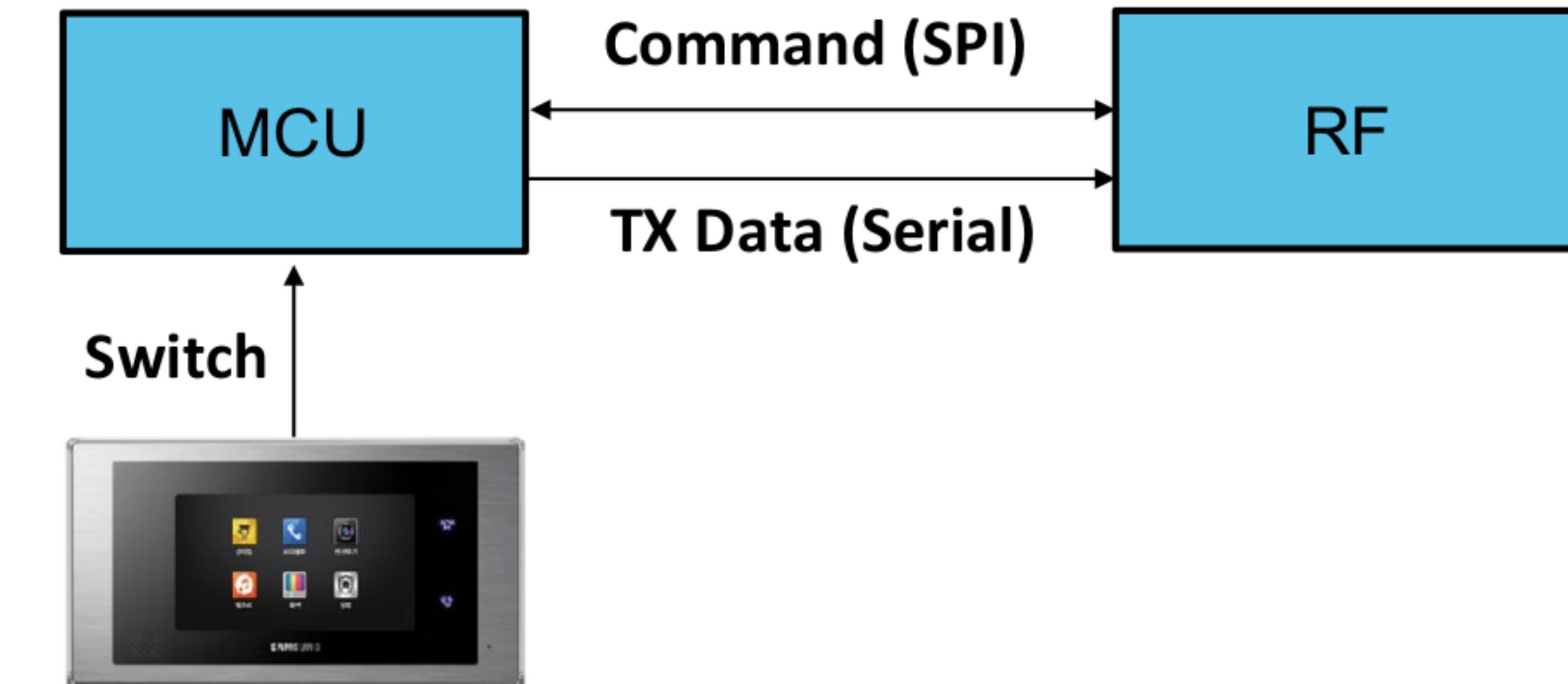
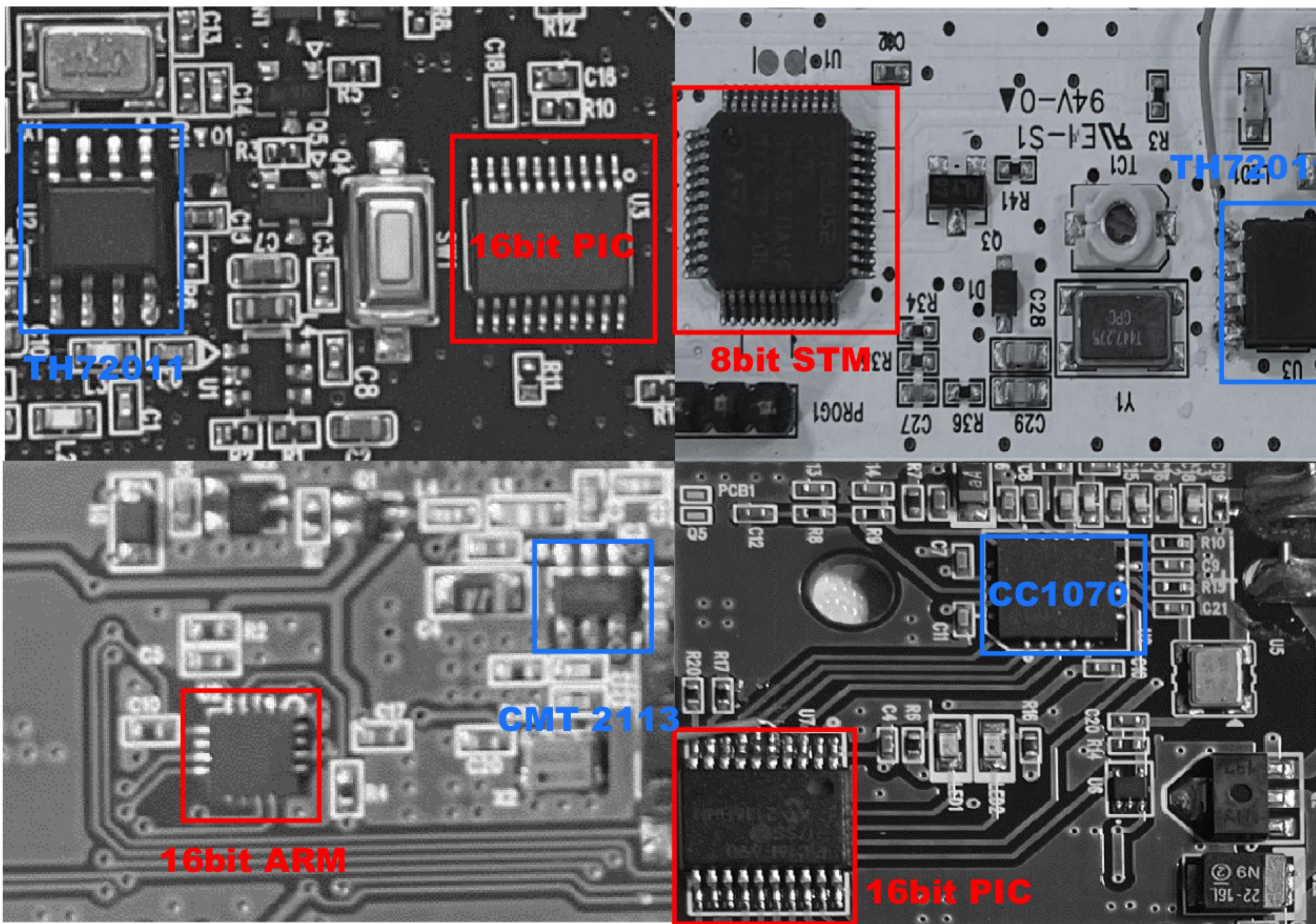
## Mainboard-based Auth. (RF IC Only Dongle)



- **Sampling**
- **RF Settings**
- **Authentication**

# Types of Door Lock TX

## □ Simplex Communication



## Feature of Transmitter HW

- **MCU**
  - 16Bit ARM / 16bit PIC /8bit PIC /8bit STM
  - Digital Encoding
  - Message Encryption
- **RF Chip**
  - Tx Only, FSK Modulation
  - Low-cost(<\$3) Discontinuous Phase Type (CC1070 /TH72011 /CMT 2113)



# **Understanding Rolling Code and Variant Replay Attack**

---

**Concise summary and Applied to Door Lock**

To ensure Secure Rolling Code transmission

1. No transmission is ever repeated

- Each transmitted message should have **different contents**
- Receiver should **ignore messages** that have already been sent
- Keep track of the last used code
  - ❑ But, re-synchronization should be considered (securely)

2. The packet contents are virtually impossible to predict, even if previous messages are known

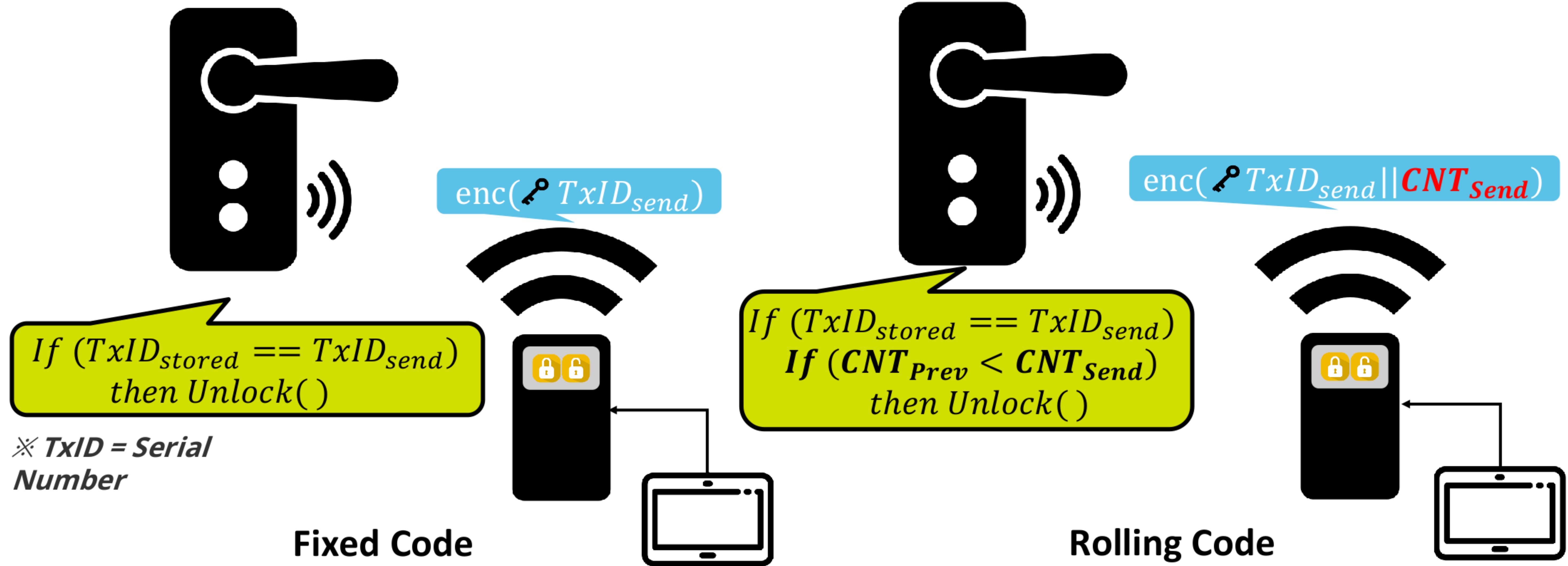
- Protect the confidentiality of the rolling code (Encryption Algorithm)
  - ❑ "TxID" and "rolling counter" are the information that needs to be kept confidential
  - ❑ it can only be read by the intended recipient

3. Prevent some unauthorized access

- Filtering mechanism, a unique serial number(TxID) is used to achieve
  - ❑ TxID should not be guessable and should not appear in a sequential format

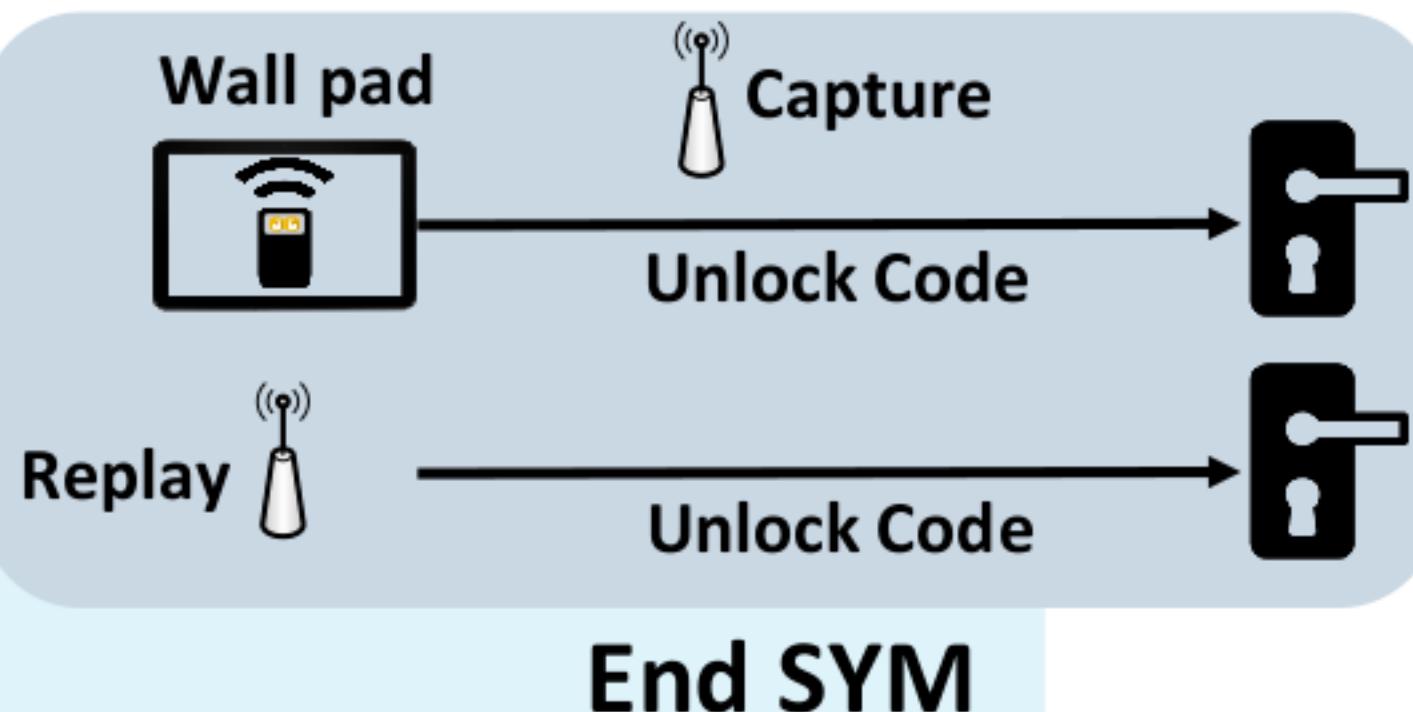
# Rolling Code in essence

It is used to protect the packet from being replayed

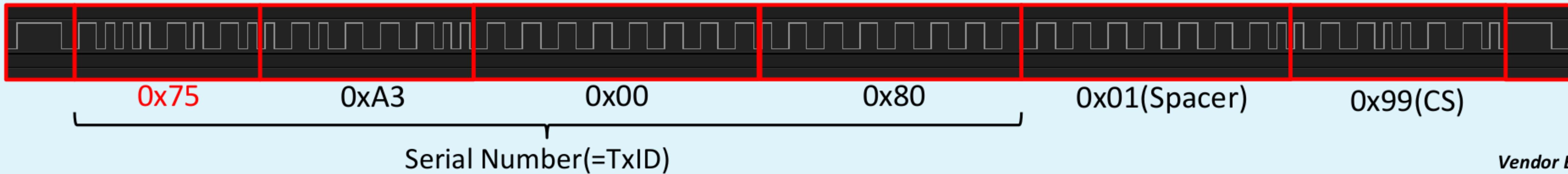
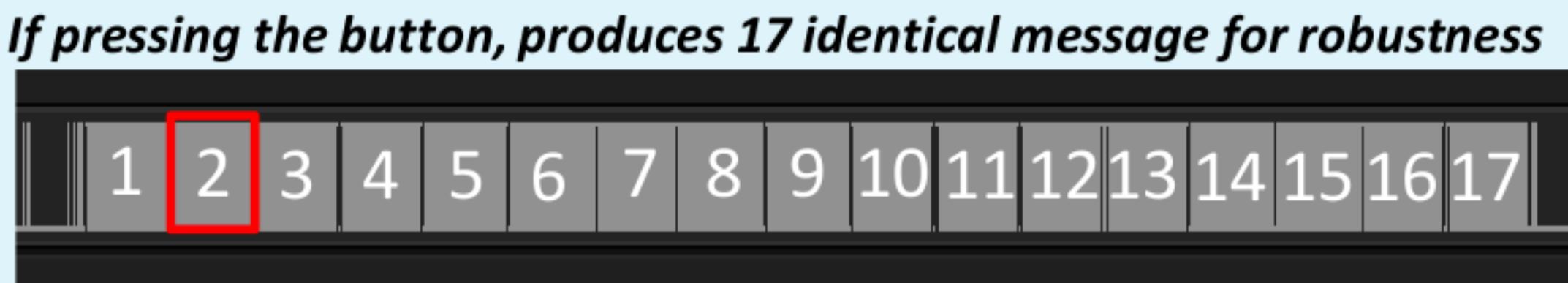


# Fixed Code on Door locks

- Fixed code is very weak in “classic replay attack”



**SYNC symbol**



Received 5-Byte in door lock mainboard(volatile)

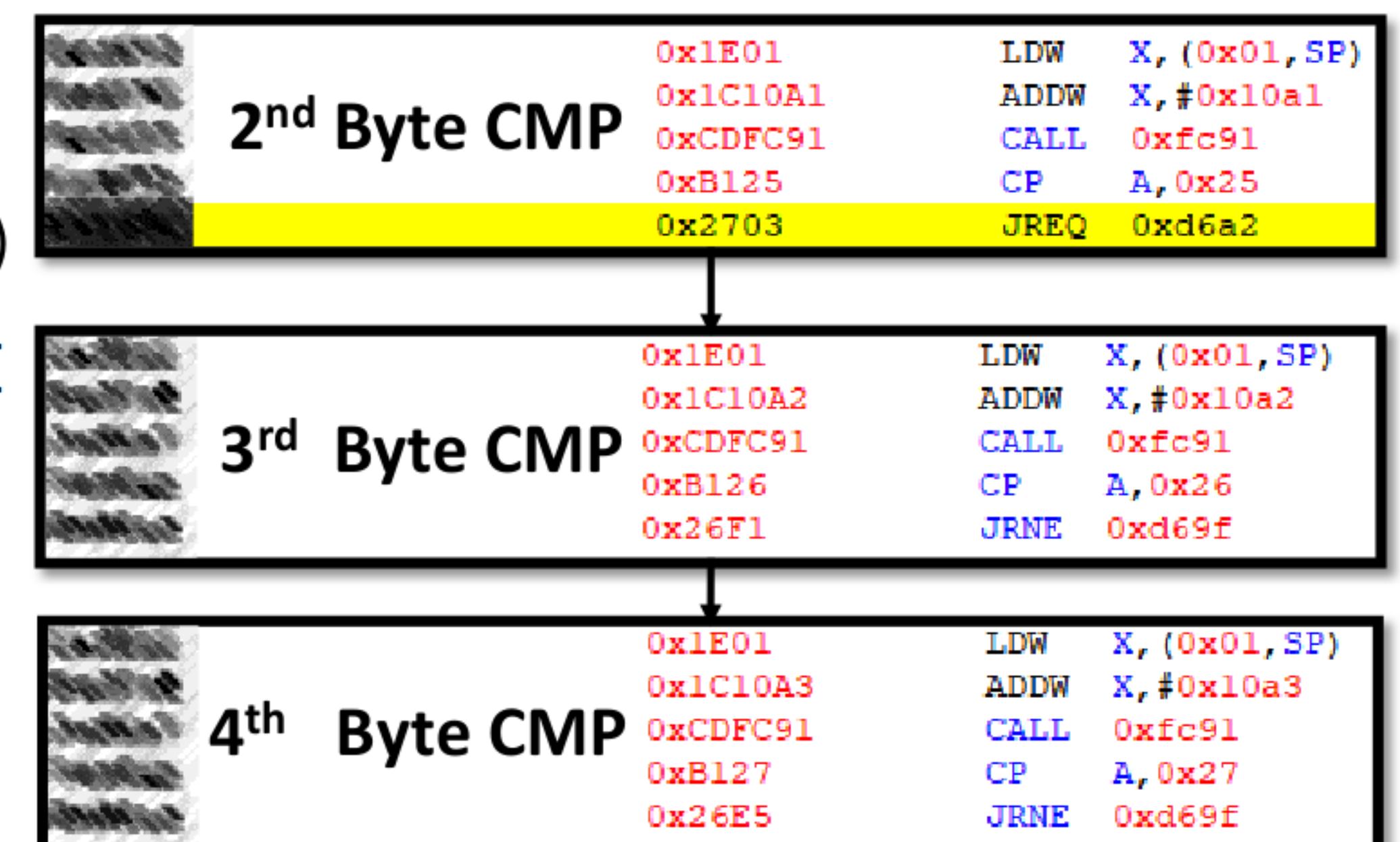
00 00 00 00 00 01 00 02 00 50 00 00 00 02 00 00 .....P.....  
 00 00 00 00 7A 22 C9 86 01 00 EB 00 40 01 08 C5 .....z".....@...

Registered 4-Byte TXID in door lock mainboard(non-volatile)

7A 22 C9 86 75 00 00 00 00 00 00 00 00 00 00 00 z" ..u.....  
 FF 4F 52 4B FF FF FF FF FF FF 00 00 00 00 00 00 .ORK.....

0x89	PUSHW X
0x97	LD XL,A
0xA605	LD A,#0x05
0x42	MUL X,A
0x1F01	LDW (0x01,SP),X
0x1C10A0	ADDW X,#0x10a0
0xCD91	CALL 0xfc91
0xB124	CP A,0x24
0x260C	JRNE 0xd69f

**Rest 3bytes comparison**



The comparison of the value of the first byte stored

# Roll Jam Attack (DEFCON23)

- Variant of Replay Attack
- No time stamp, difficult to prevent unused code replay attack

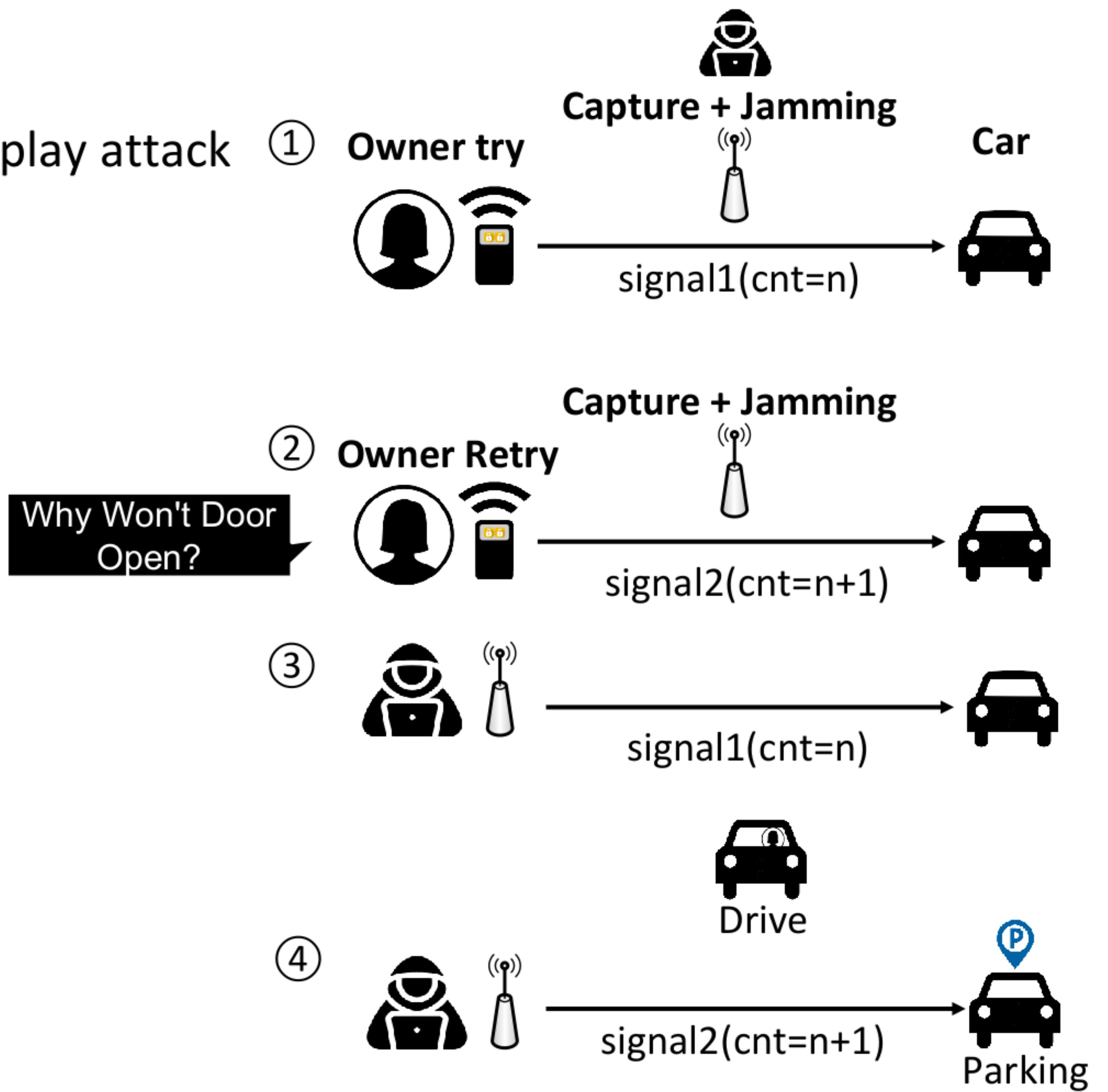
## RollJam Concept

Eve capture unused code(signal)

Replay unused code later

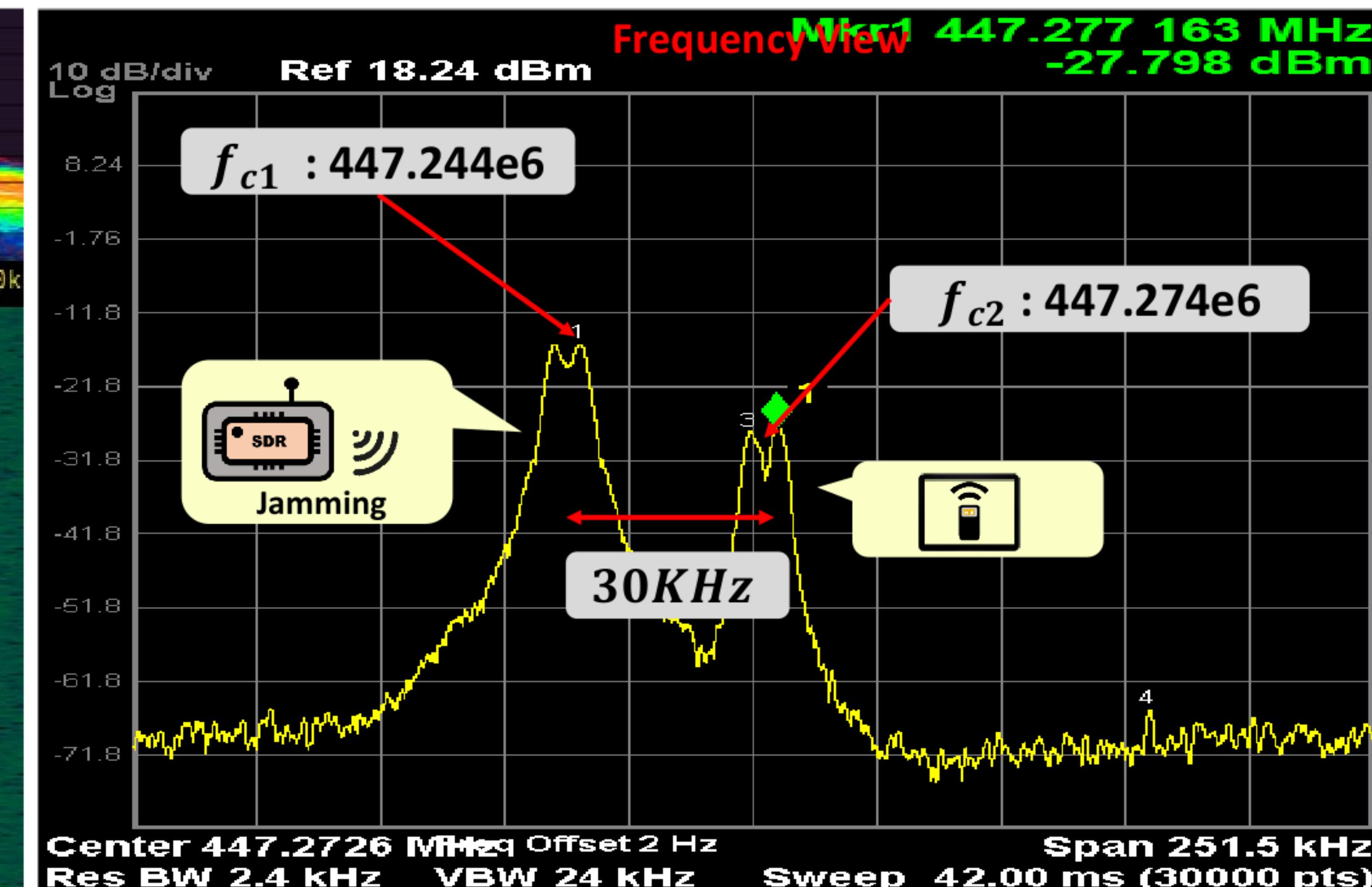
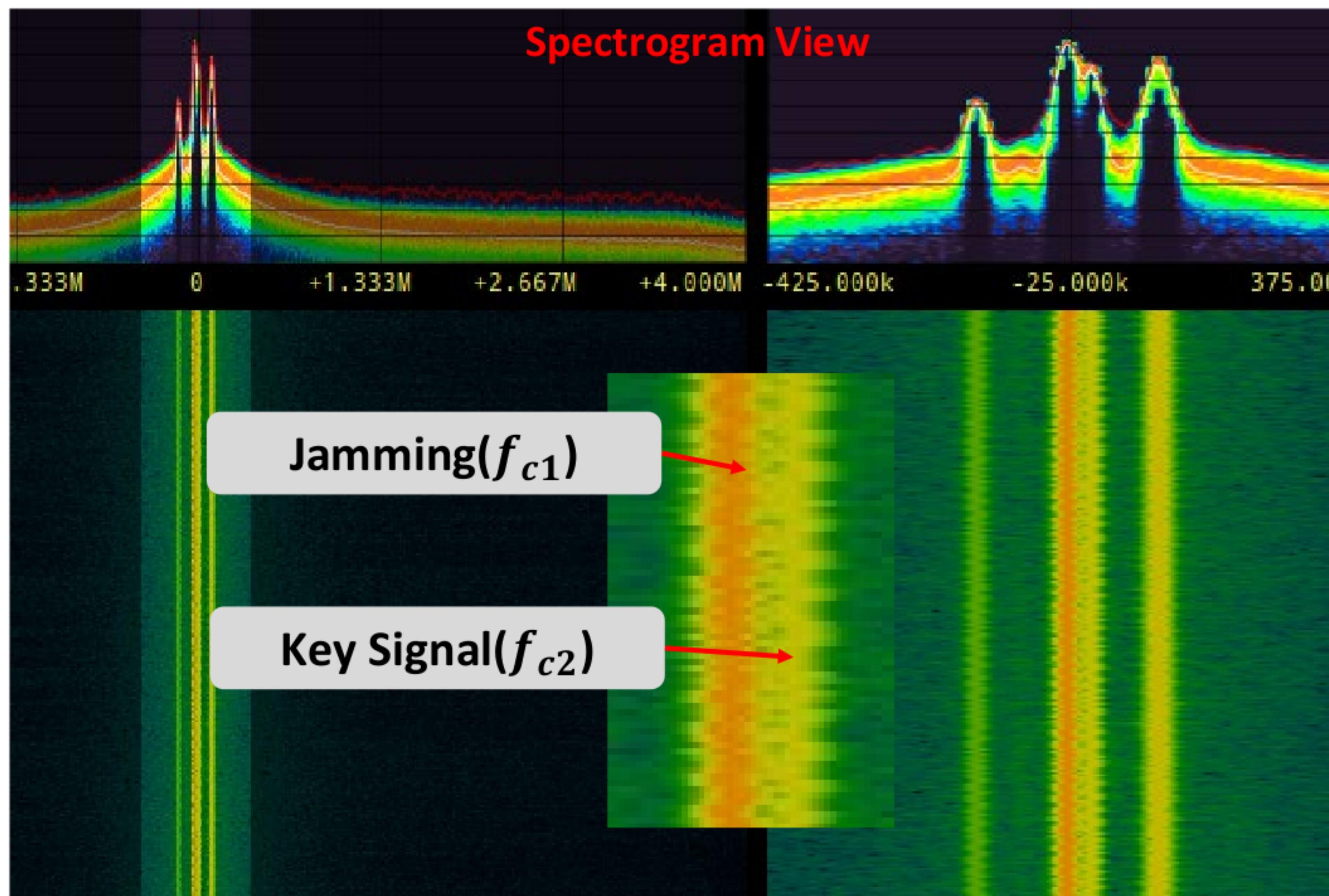
## Process

- ① Capture signal1 + Jamming
- ② Capture signal2 + Jamming
- ③ Capture Signal1 Replay (for her)
- ④ Capture Signal2 Replay (for carjack)

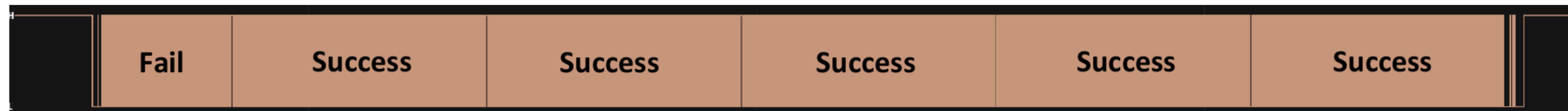


# Jamming & Capture on Doorlock

- The important point is that the jamming signal should effectively interfere with the original signal without completely overpowering it. (< 30KHz gap is best)



The captured signal during jamming



# RollJam Attack on Door Lock

❑ Eve capture unused code(signal), Replay Later

❑ Process

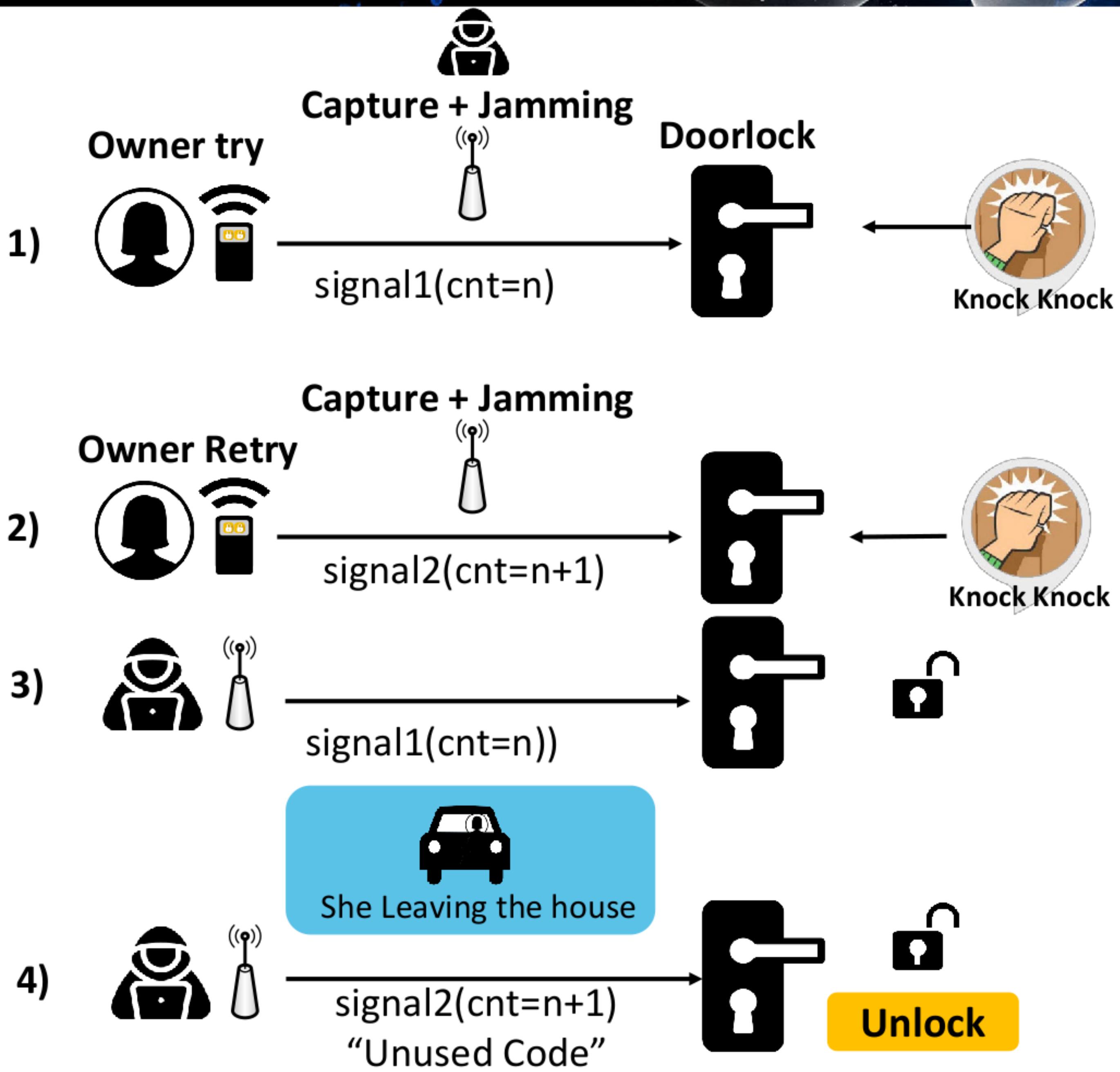
- 1) Capture signal1 + Jamming
- 2) Capture signal2 + Jamming
- 3) Capture Signal1 Replay (for her)
- 4) Capture Signal2 Replay (for theft)

❑ Two drawbacks of RollJam

- ❑ An attacker has to be precise
  - the timing is crucial.

❑ The attack can be launched once

- If the attacker wants to gain access to the same door lock again, they would need to start the process all over again from the beginning.



Because signal 1 has been used, but signal 2 has not been used

# RollBack Attack (BHUSA2022)

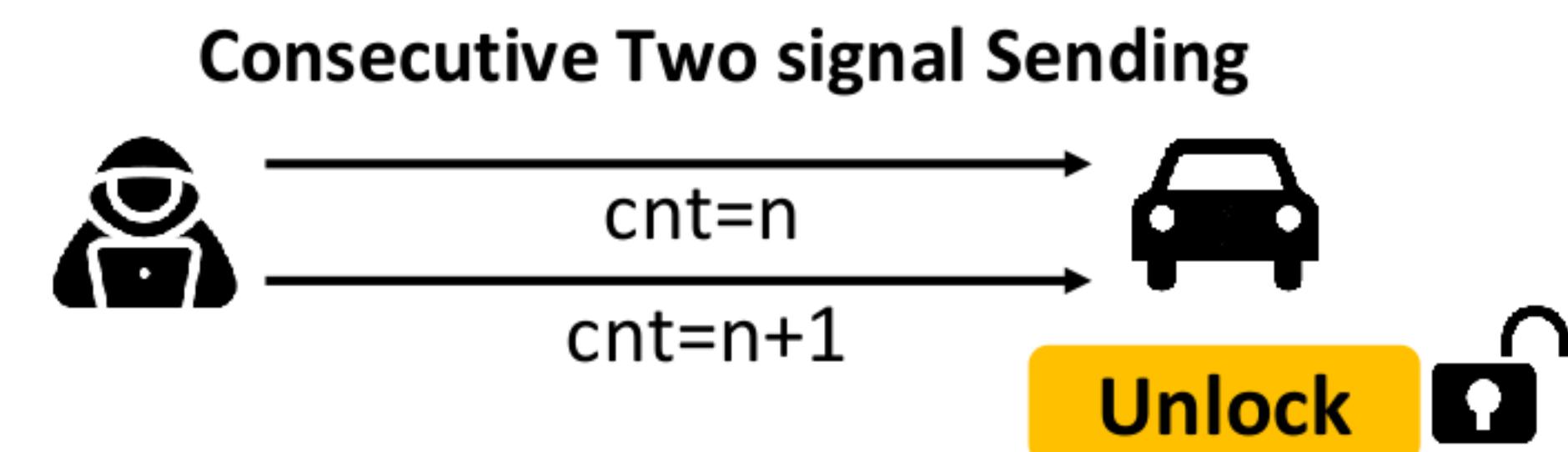
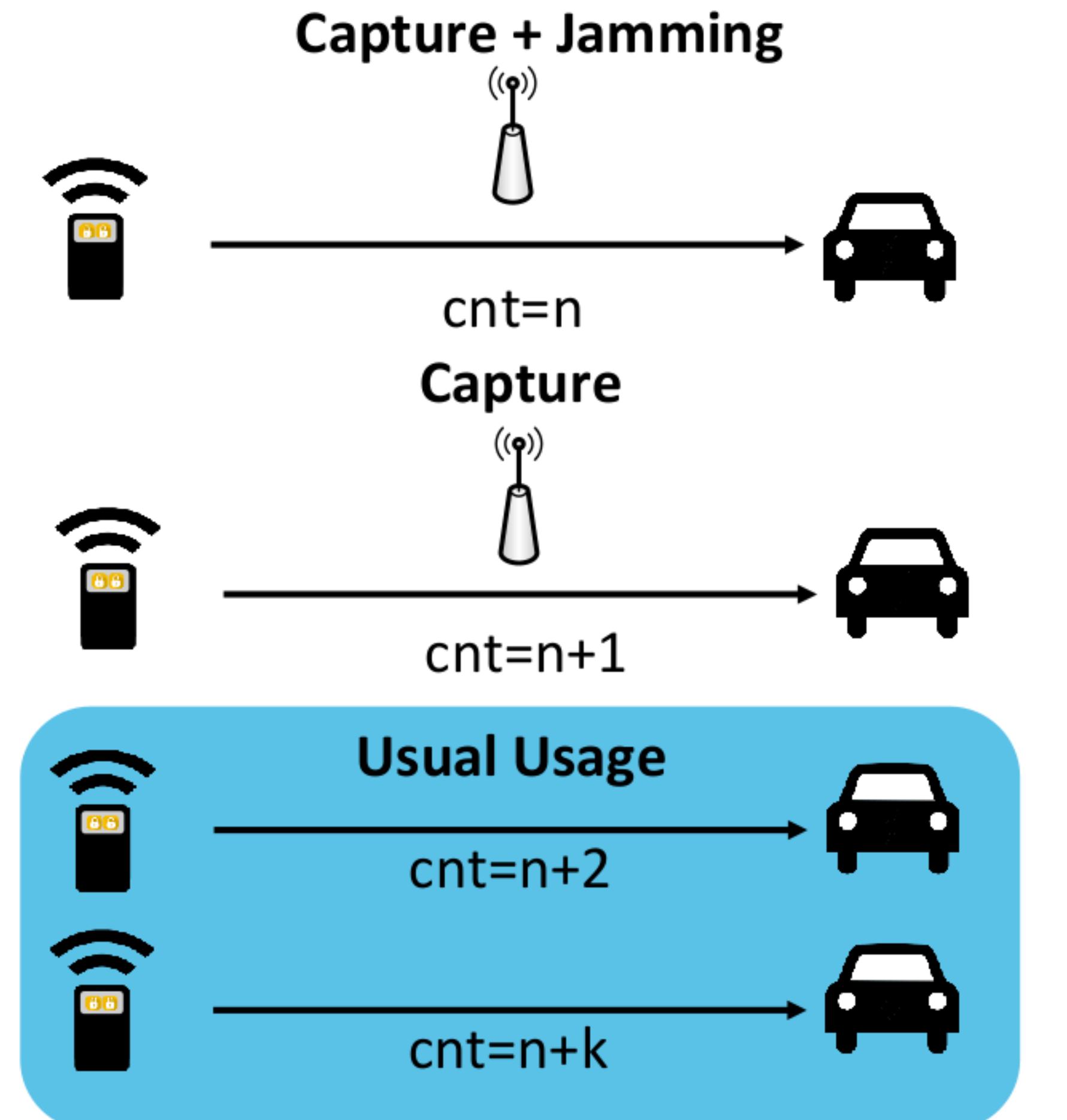
## Rollback Attack Process

- 1) Send Unlock signal
- 2) Capture + Jamming
- 3) Send Unlock signal
- 4) Capture
- 5) Owner uses the key fob as usual(many times)
- 6) replay the two consecutive signal

## Characteristic

- At any time in the future
- As many times as desired

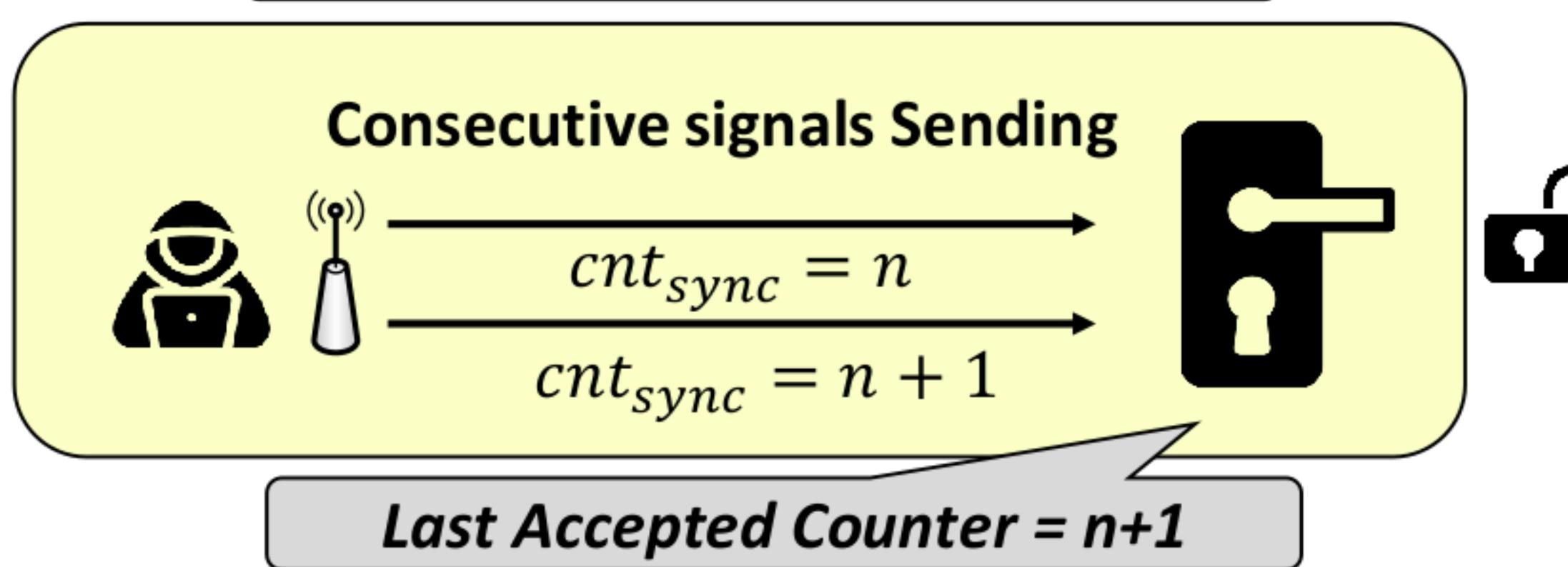
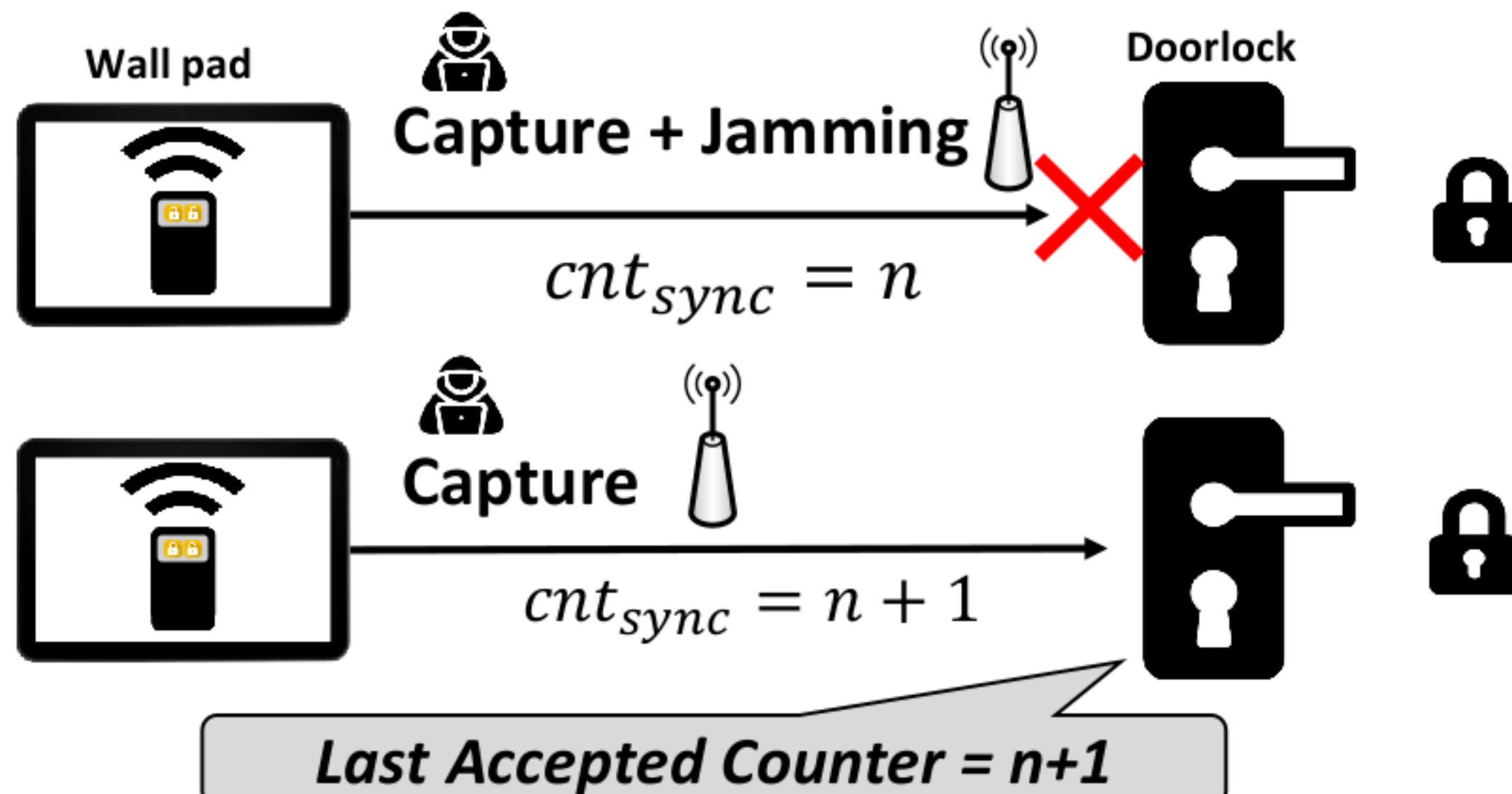
 Attacker  
 Victim



note: vehicle re-synchronize to a previous code

# Root cause of RollBack on Door Lock

- If a received message is valid, then the counter value always be stored on memory



001060	FF FF FF 00 00 00 00 00 00 00 00 00 00
001070	00 00 00 00 00 00 00 00 00 00 00 00 00
001080	95 A5 28 AL .. 00 00 00 00 00 00 00 00
001090	B6 B4 00 00 00 00 00 00 00 00 00 00 00
0010A0	2C 08 9E AL .. 9F FF FF 00 00 00
0010B0	FF 00 00 00 .. .. .. .. .. .. .. .. .. ..

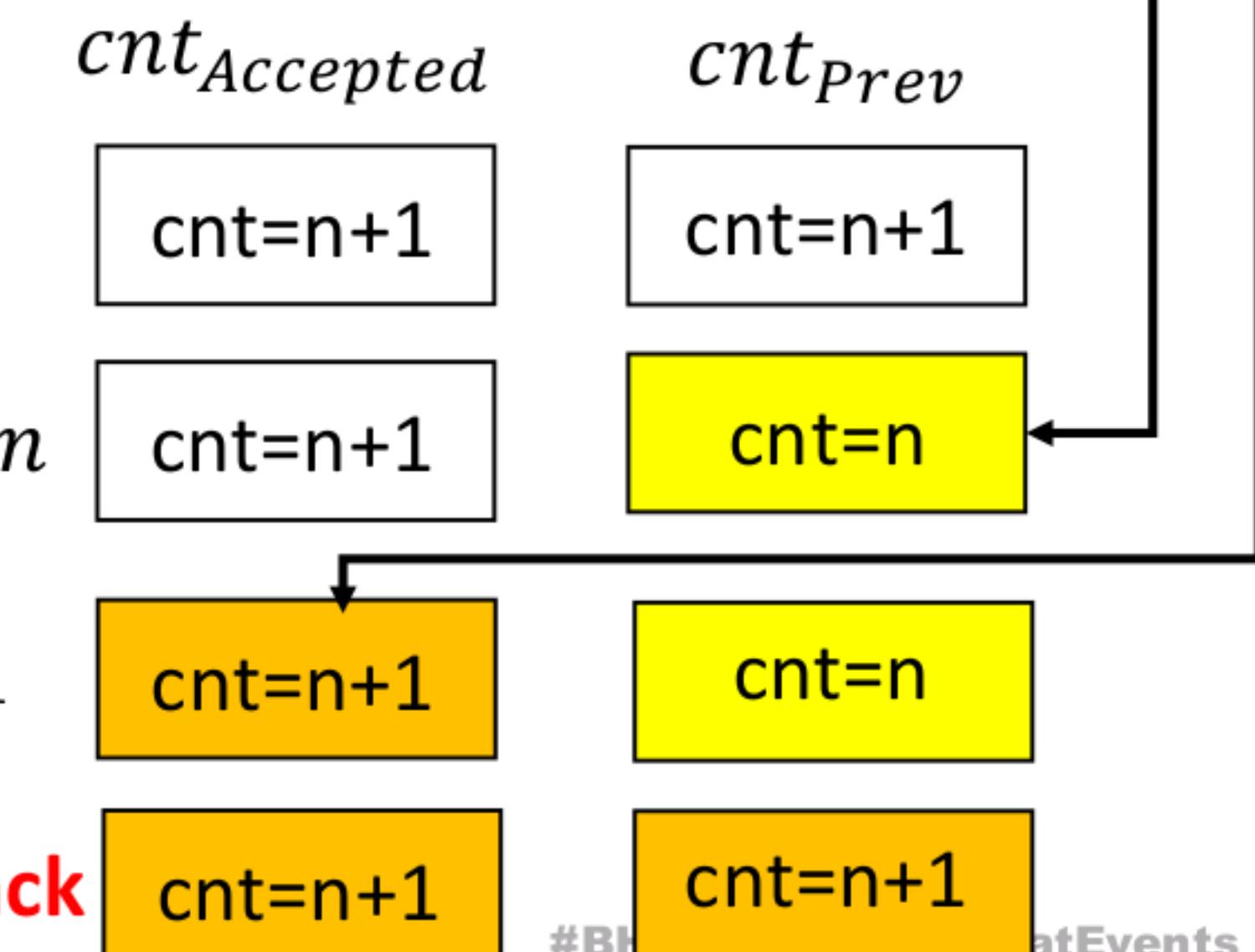
*cntAccepted (Last Accepted Counter)*

*cntPrev (Previous Counter)*

```

Input cntsync
Variable cntPrev // received counter
Variable cntAccepted // Last Accepted Counter
if Message is valid then
    if cntsync > cntPrev then // If valid counter
        cntAccepted ← cntsync // update Last Accepted Counter
        cntPrev ← cntsync // received counter stored
        Doorlock Open()
    else then
        cntPrev ← cntsync // received counter stored ??? Why?

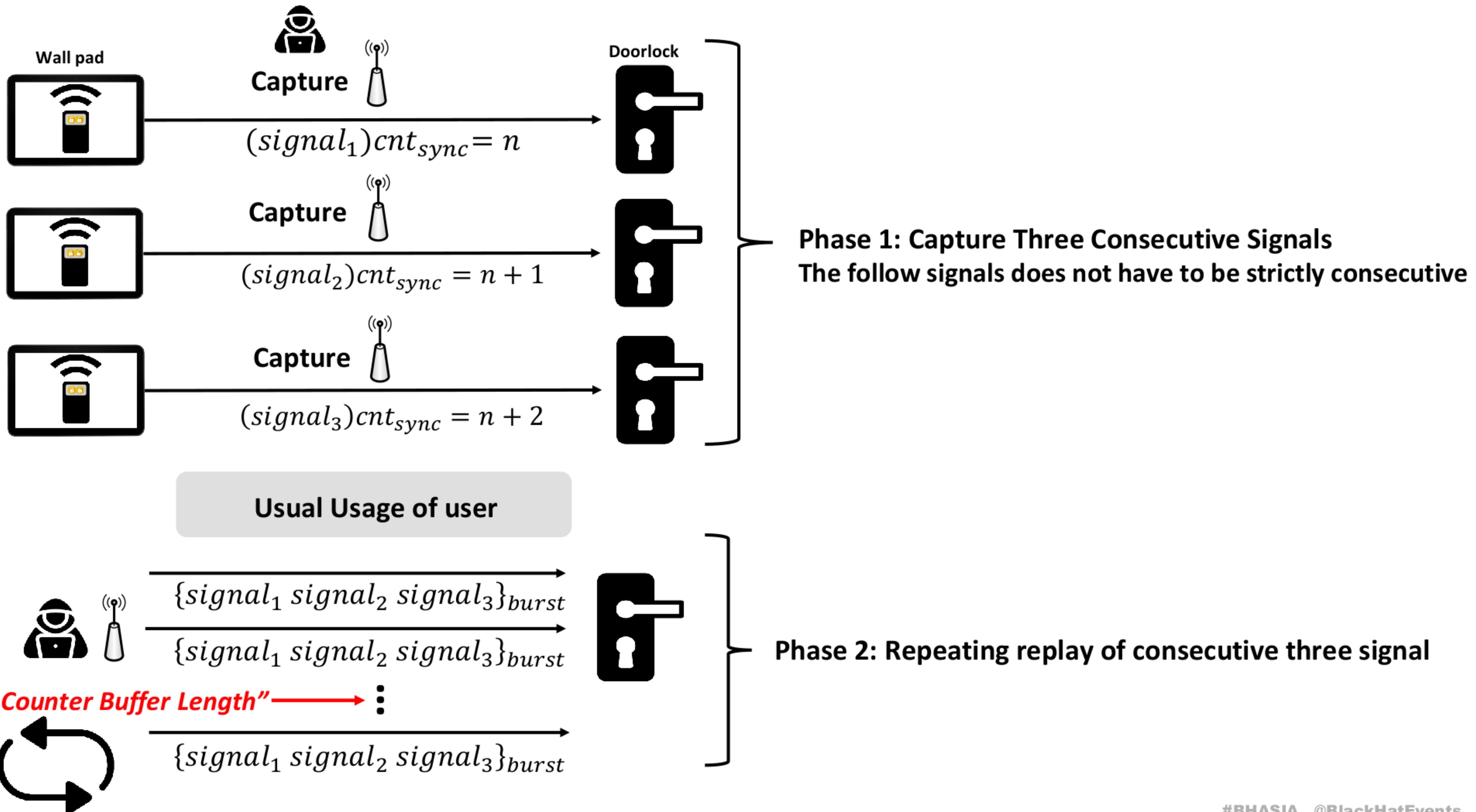
```





# Rollback Attack on Door Lock Demo Video

# Loop Playback Attack (Variant)

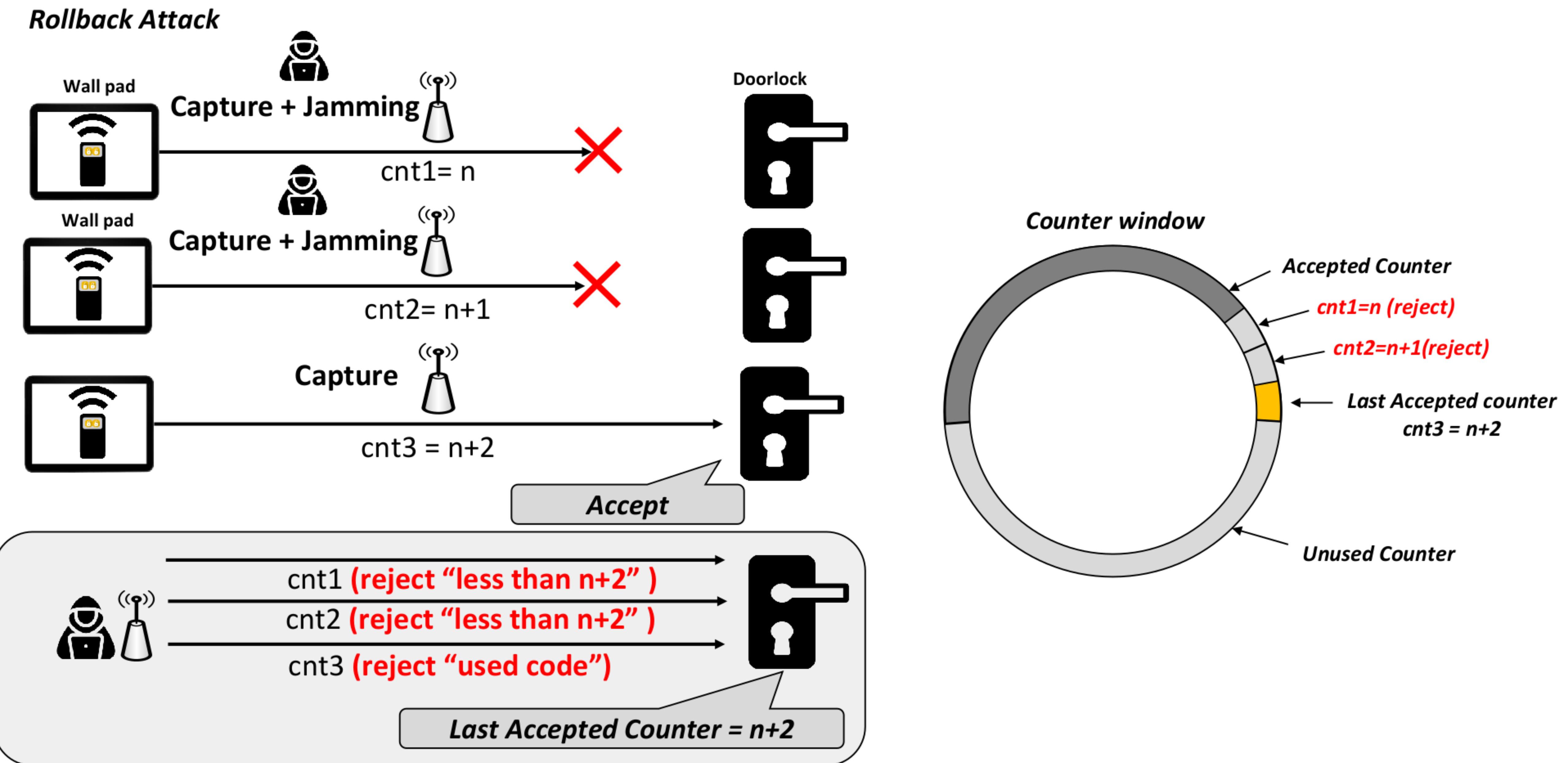




# Loop Playback Attack Demo Video

# Rollback attack won't fly!

- ❑ Most of door locks keep track of the last used code, and never accept used counter



# Evaluation : Replay Attack

- Evaluation on a limited set of door locks(2021~2022)

Vendors	Models	Classic Replay	Rolljam	Rollback	Loop Playback	Remarks	
A	A-1	✓	✗	✗	✗	Consecutive two signals	
B	B-1	✓	✗	✓	✗	10 Minute	
	B-2	✓	✗	✓	✗		
	B-3	✓	✗	✓	✗		
	B-4	✓	✗	✓	✗		
C	C-1	✓	✗	✓	✓	X Very weak	
	C-2	✓	✗	✓	✓		
	C-3	✓	✗	✓	✓		
D	D-1	✓	✗	✓	✓	X Weak	
	D-2	✓	✗	✓	✓		
E	E-1	✗				one signal	
F	F-1	✗					
G	G-1	✗					
H	H-1	✗					
I	I-1	✓	✗	✓	✓		
J	J-1	✓	✗	✓	✓		

✗ Very weak

✗ Weak

✓ Moderate

✓ Probably safe



# **Protect the confidentiality of the code**

---

## **Unveiling the Vulnerabilities in Door Lock RF Encryption Design**

# Principles of Secure Rolling Code

To ensure Secure Rolling Code transmission (The three critical properties)

## 1. No transmission is ever repeated

- Each transmitted message should have **different contents**
- Receiver should **ignore messages** that have already been sent
- Keep track of the last used code
  - But, re-synchronization should be considered

## 2. The packet contents are virtually impossible to predict, even if previous messages are known

- Ultimately, the system should be designed to make it difficult for an attacker to guess and replicate the message
  - "Serial number(=TxID)" and "sync counter" are the information that needs to be kept confidential
  - It can only be read by the intended recipient

## 3. Prevent unauthorized access

- Filtering mechanism, a unique serial number(TxID) is used to achieve
  - TxID should not be guessable and should not appear in a sequential format

## ❑ Type1: Pre-programmed cipher key

### Type1-1 Fixed Key

- ❑ The implementation is simple and cost-effective
- ❑ same cipher key is used across multiple transmitters, increased security risk

### Type1-2 Random Key

- ❑ Random like generated cipher keys is used, better protection
- ❑ Lost or damaged, a new transmitter cannot be used with the receiver, if not have a learning mechanism

## ❑ Type2: Derives the encryption key by using received data during normal operation

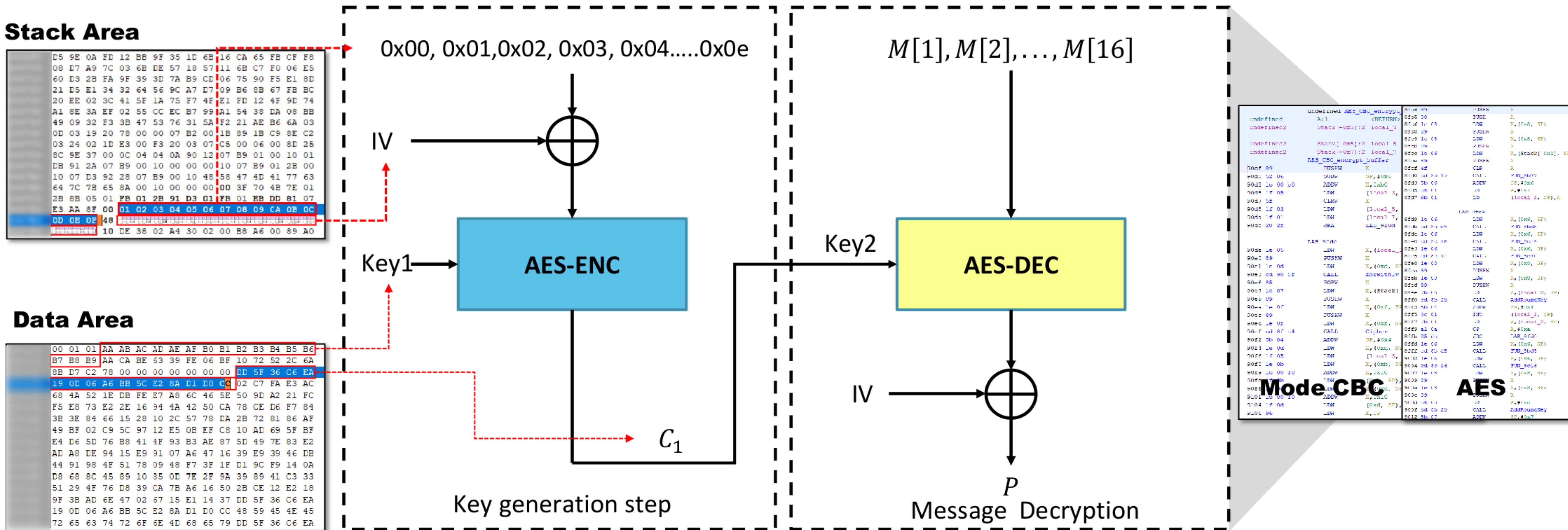
- ❑ It is more secure than Fixed Key, and more flexible than random key
- ❑ This method requires additional security measures for enhancing

## ❑ Type3: Transmit key generation seed value at learning time

- ❑ The receiver uses this seed value to derive the same encryption key
- ❑ During normal operation, attacker will not have any information about the encryption key used

# AES-based Door Lock RF Encryption

- Using hardcoded values for the IV and plaintext in every key generation operation
  - The values are hard-coded in code area



# The Flaws in Key Generation

- Derives the encryption key by using received data during normal operation

If M[0] '1' is even

Received Packet 17-Bytes

AA CA BE 63 39 FE 06 BF 10 72 52 2C 6A 8B D7 C2 78

AA = b'1010\_1010 (number of bit 1 is EVEN)

Generated Key1 16-Bytes

= AA AB AC AD AE AF B0 B1 B2 B3 B4 B5 B6 B7 B8 B9

00	01	01	AA	AB	AC	AD	AE	AF	B0	B1	B2	B3	B4	B5	B6
B7	B8	B9	AA	CA	BE	63	39	FE	06	BF	10	72	52	2C	6A
8B	D7	C2	78	00	00	00	00	00	00	00	DD	5F	36	C6	EA
19	0D	06	A6	BB	5C	E2	8A	D1	D0	CC	02	C7	FA	E3	AC
68	4A	52	1E	DB	FE	E7	A8	6C	46	5E	50	9D	A2	21	FC

If M[0] '1' is odd

Received Packet 17-Bytes

BC 0E 2C 19 35 44 1B F7 52 1D 43 6D 0A 10 C6 20 DA

BC = b'1011\_1100 (number of bit 1 is ODD)

~ 0xBC = 0x43 (Bit inversion)

= 43 44 45 46 47 48 49 4A 4B 4C 4D 4E 4F 50 51 52

If M[0] '1' is even

then key1 =  $M[0], M[0] + 1, M[0] + 2, \dots, M[0] + 15\}$

If M[0] '1' is even

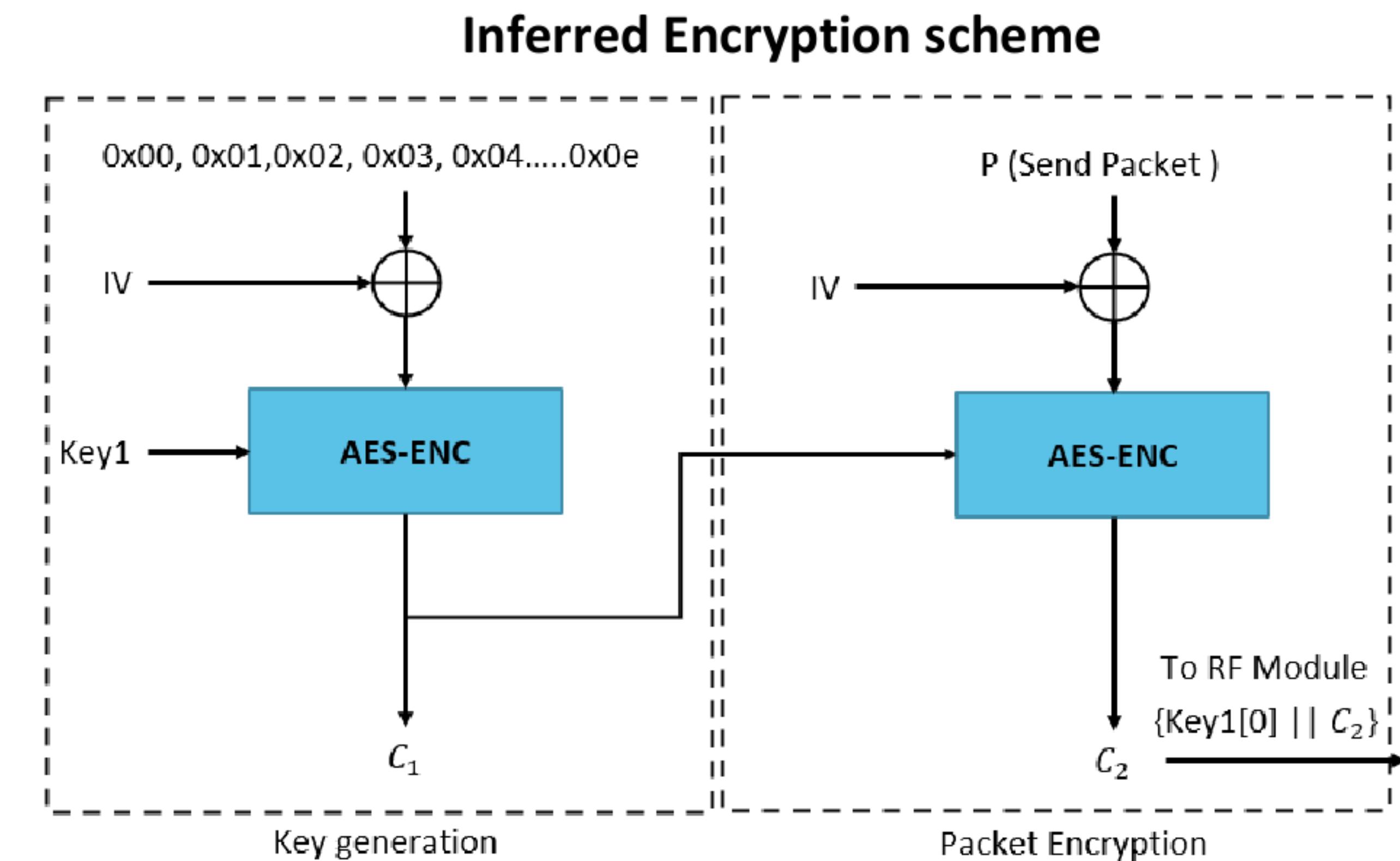
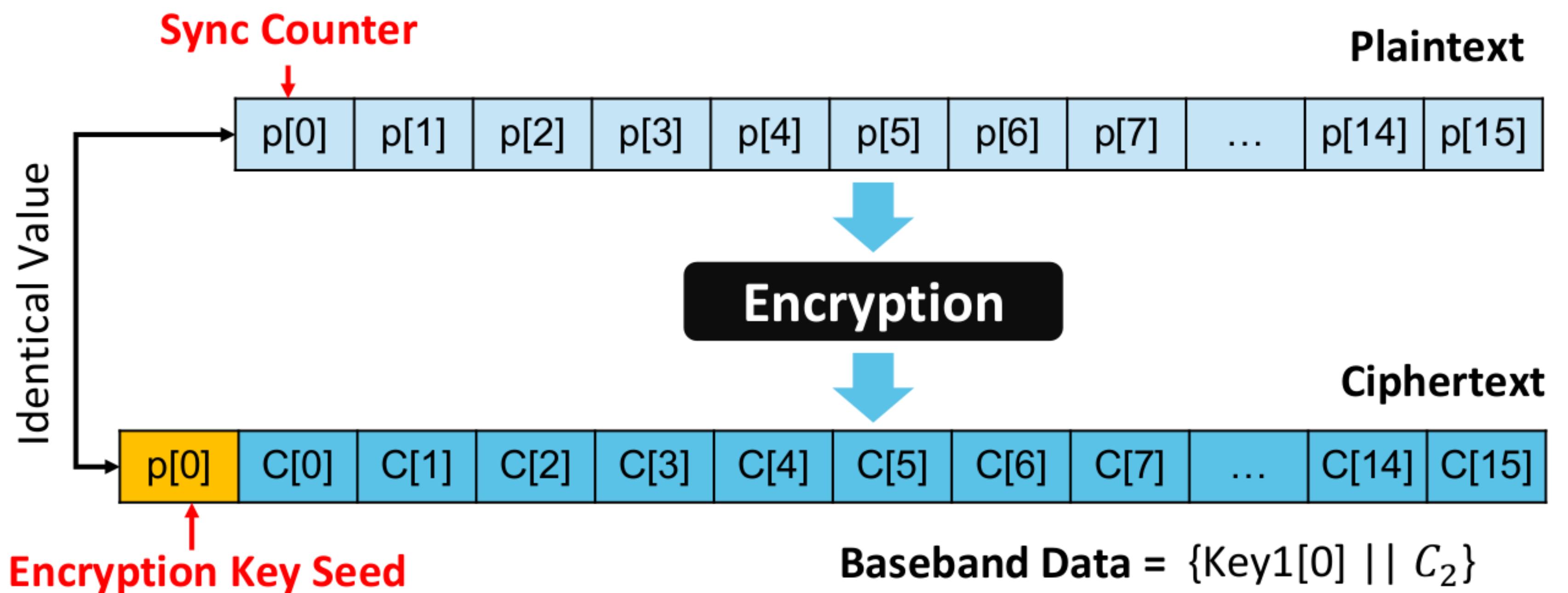
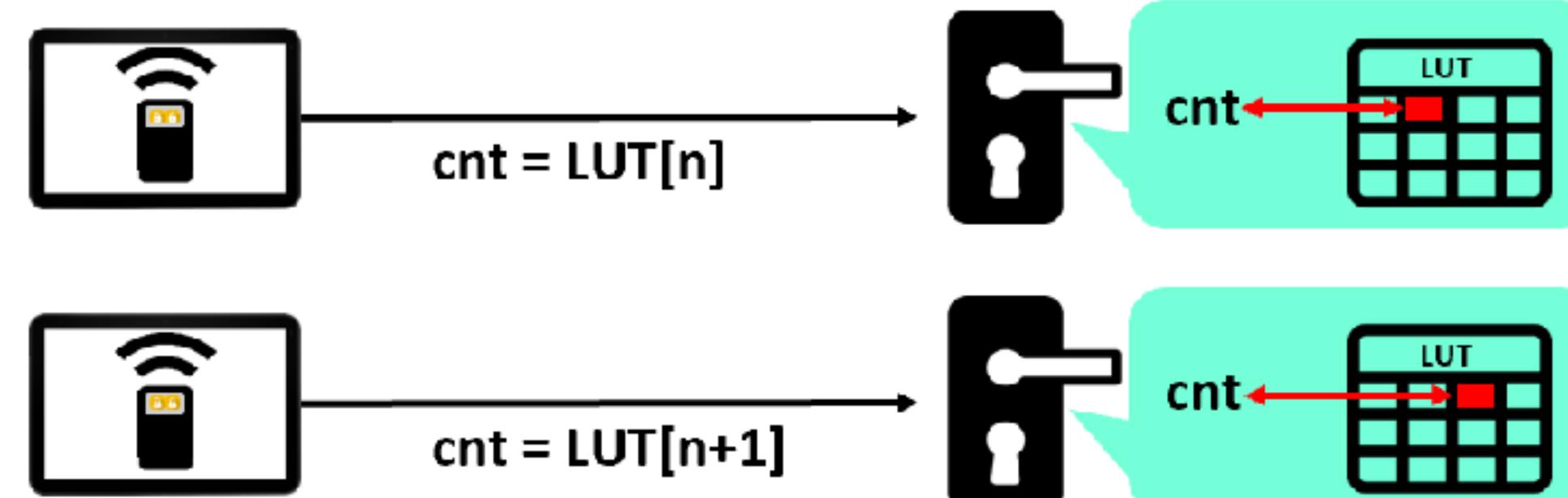
then key1 =  $\sim M[0], \sim M[0] + 1, \sim M[0] + 2, \dots, \sim M[0] + 15\}$

Little trick, confusion to an attacker



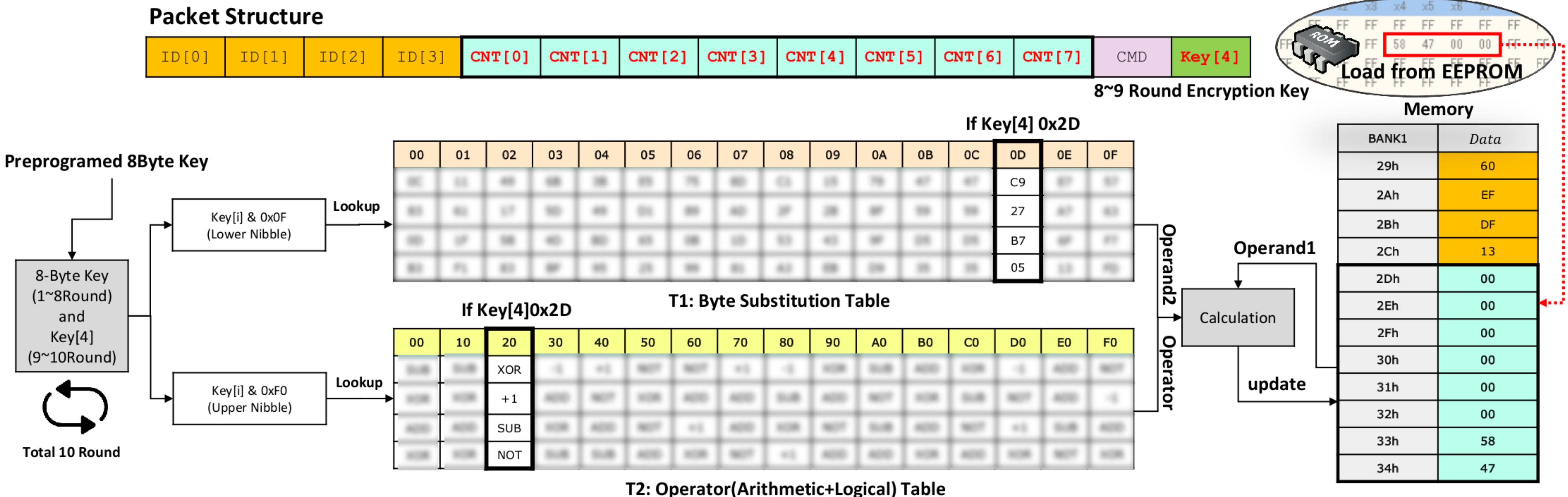
# Counter as Key Generation Seed

- The Encryption key seed is also used to sync counter
- The key seed is exposed on is being transmitted in plaintext
  - An attacker could decrypt ciphertext at any time
- It rely on the secrecy of the encryption scheme and key generation mechanism
  - It may potentially allow an attacker to break other devices that use the similar implementation



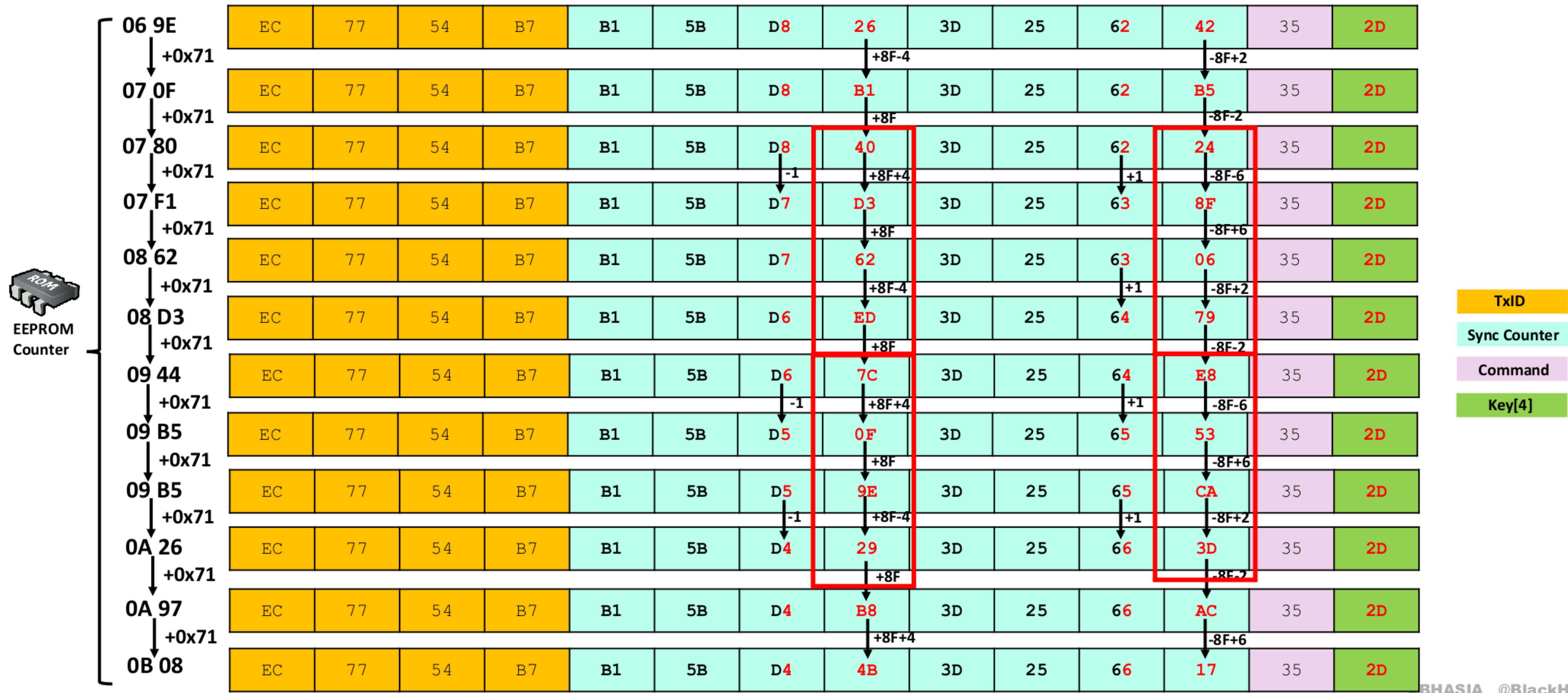
# Proprietary Cipher on Door Lock

- Transmit cipher key at learning time
  - Preprogrammed 8-byte key for 8 rounds, with the 5th key being exposed in the packet for 2 rounds
  - This cipher is to use a combination of substitution and operator table to generate a rolling counter



# Monotonous Pattern of Code

- ❑ The consistent pattern code may be due to the absence of a permutation feature
- ❑ The generated rolling counter has a somewhat monotonous pattern, making it easy for attackers to predict



# Evaluation : Confidentiality

- Evaluation on a limited set of doorlocks
  - All models are vulnerable to picking the lock with sniffed RF packets

Vendors	Models	Algorithm	Confidentiality	Remarks
A	A-1	AES-128	✗	
B	B-1	AES-128	✗	
	B-2	AES-128	✗	
	B-3	AES-128	✗	
	B-4	AES-128	✗	
C	C-1	AES-128	✗	
	C-2	AES-128	✗	
	C-3	AES-128	✗	
D	D-1	AES-128	✗	
	D-2	AES-128	✗	
E	E-1	None	✗	X Compromised
F	F-1			
G	G-1			
H	H-1			
I	I-1	XTEA	✗	1) Pre-programmed Fixed Key(Hardcoded)
J	J-1	Proprietary Encryption	✗	1)Pre-programmed random key is transmitted at learning time 2) It is feasible to deduce the next code from the packet 3) Serial Number(=TxID) is leaked to plaintext
	J-2			



# **Authentication in RF-based Door Locks**

---

**The crucial of ID in RF system**

To ensure Secure Rolling Code transmission (The three critical properties)

## 1. No transmission is ever repeated

- Each transmitted message should have **different contents**
- Receiver should **ignore messages** that have already been sent
- Keep track of the last used code
  - But, re-synchronization should be considered

## 2. The packet contents are virtually impossible to predict, even if previous messages are known

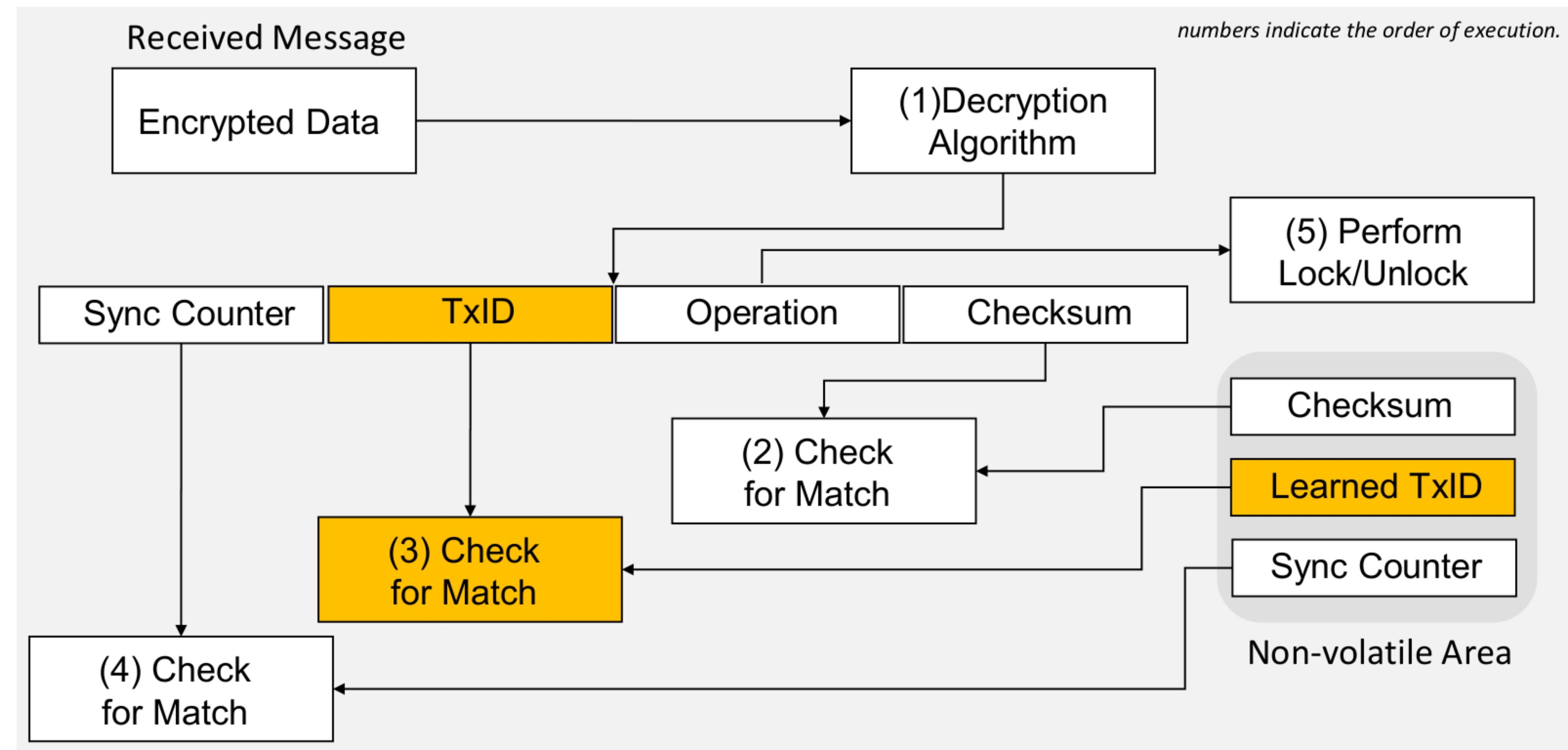
- Protect the confidentiality of the rolling code (Encryption Algorithm)
  - "TxID" and "rolling counter" are the information that needs to be kept confidential
  - It can only be read by the intended recipient

## 3. Prevent unauthorized access

- Serial Number is learning information in most of door lock, ID verification is a common method for authorizing
- Serial Number (=ID) should not be guessable and should not appear in a sequential format

# Authentication check

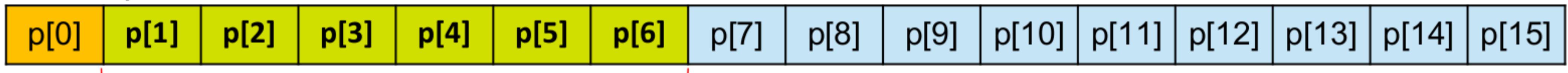
- ❑ Filtering mechanism, a unique serial number(TxID) is used to achieve
- ❑ When a valid message is received, the message is decrypted, and the serial number is used to determine if it is from a learned transmitter. If it is from a learned transmitter, the synchronization counter is verified



# Packet Confusing before Encryption

- It may be difficult to keep track of the original order of the elements

Encryption Key Seed



Confusing Area

S1 (Original order of the elements)

p[1]	p[2]	p[3]	p[4]	p[5]	p[6]
------	------	------	------	------	------

S2

p[3]	p[6]	p[1]	p[4]	p[2]	p[5]
------	------	------	------	------	------

S3

p[4]	p[2]	p[5]	p[1]	p[6]	p[3]
------	------	------	------	------	------

3-States (vendor-specific)

```
function confusing(encryption_key_seed, s1, s2, s3, range1, range2)
    L_Nibble = encryption_key_seed AND 0x0F
    if L_Nibble < range1 then
        circular_shift(s1, L_Nibble)
        return s1
    else if L_Nibble >= range1 AND L_Nibble < range2 then
        circular_shift(s2, L_Nibble)
        return s2
    else
        circular_shift(s3, L_Nibble)
        return s3
    end if
end function
```

Vendor-specific

Circular Shift based confusing

# Sequential ID values



- ❑ TxID(=Serial Number) should not be guessable and should not appear in a sequential format
  - ❑ If the TxID values are sequential or predictable, the attacker can easily predict the next door's value
- ❑ Actual TxID changed by only 2~3Bytes, the number of possible values is significantly reduced
  - ❑ It is important to use unique and non-sequential TxID values to prevent potential attacks.

$TxID_1 = \{0xBA, 0xA4, 0x0A, 0xA6\}$

$TxID_2 = \{0xBA, 0xA3, 0x0A, 0x43\}$

$TxID_3 = \{0xBA, 0xA6, 0x08, 0x6C\}$

$TxID_4 = \{0xBA, 0xA2, 0x09, 0x5E\}$

$TxID_5 = \{0xBA, 0xA4, 0x09, 0xC4\}$

$TxID_6 = \{0xBA, 0xA4, 0x06, 0xC4\}$

E.g., Vendor B

$TxID_1 = \{0x95, 0xA5, 0x28, 0xAE\}$

$TxID_2 = \{0x95, 0xA4, 0x26, 0xFC\}$

$TxID_3 = \{0x95, 0xA4, 0x1B, 0xDB\}$

$TxID_4 = \{0x95, 0xA5, 0x26, 0xAE\}$

E.g., Vendor A

Vendors	Models	Serial Number	Remarks
B	A-1	✗	$\approx 2.5 \sim 3$ Bytes
	B-1	✗	
	B-2	✗	
	B-3	✗	
C	B-4	✗	
	C-1	✗	$\approx 2$ Bytes
	C-2	✗	
D	C-3	✗	
	D-1	✗	$\approx 3$ Bytes
E	D-2	✗	
	E-1	✗	$\approx 2.5 \sim 3$ Bytes
F	F-1	✗	
G	G-1	✗	
H	H-1	✗	
I	I-1	✗	
J	J-1	✓	$\approx 4$ Bytes

✗ Very Weak

✗ Weak

✓ Moderate

✓ Strong



## **Technical Details of RF Lockpicking Tool**

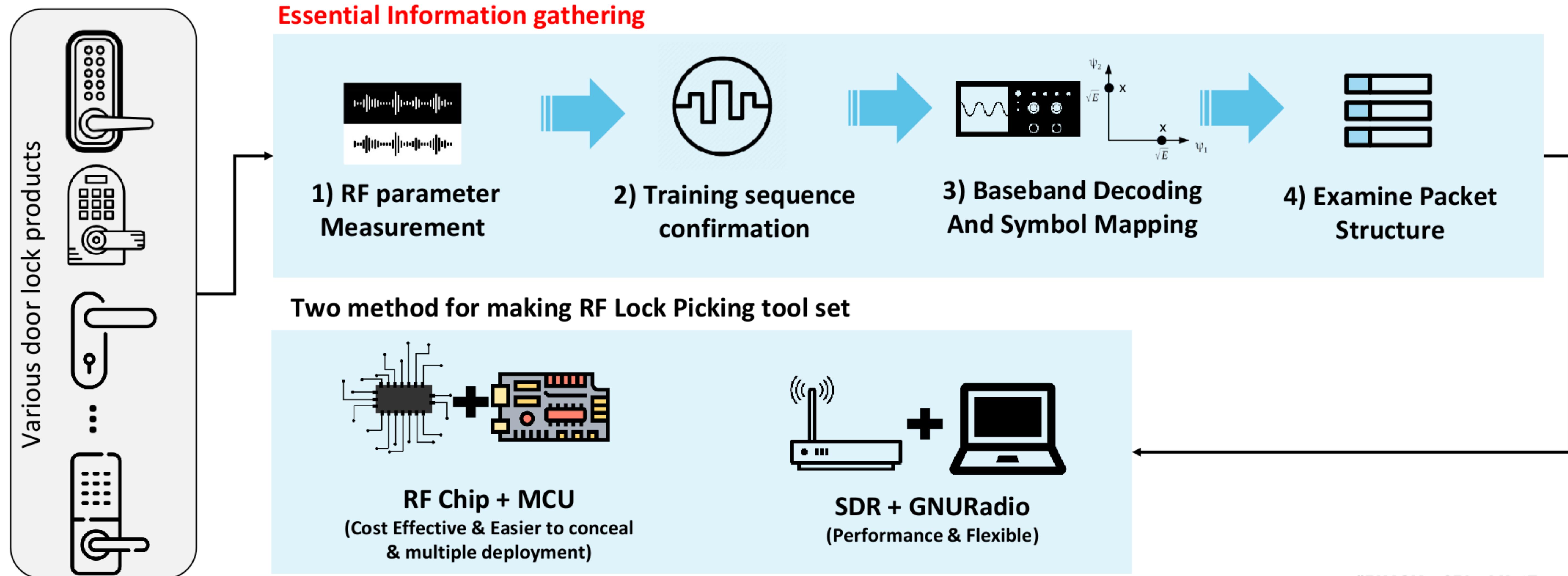
---

**Tools for RF Capture, Decoding, and Transmission**

# An overview of making a RF lock picking tool set

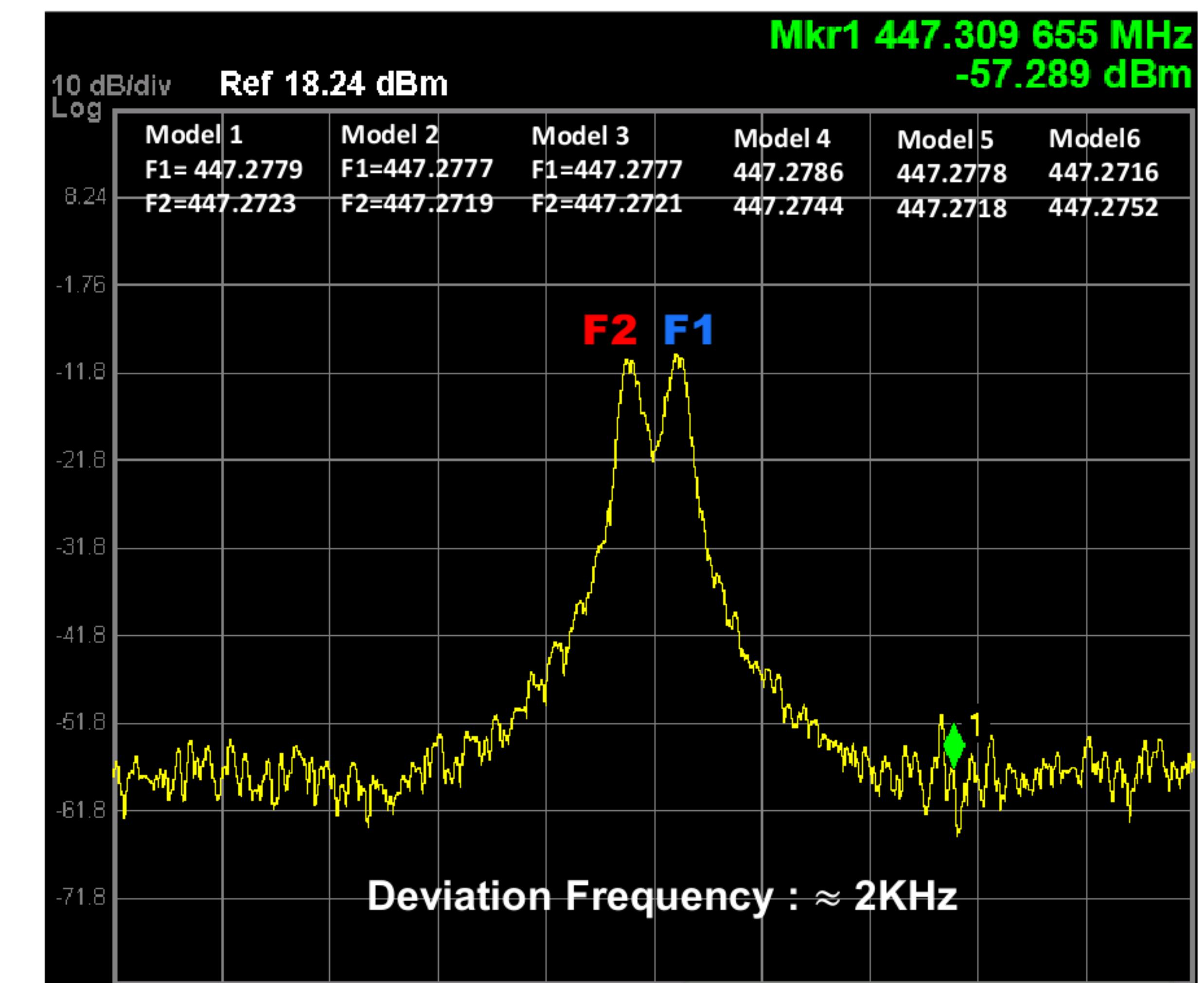
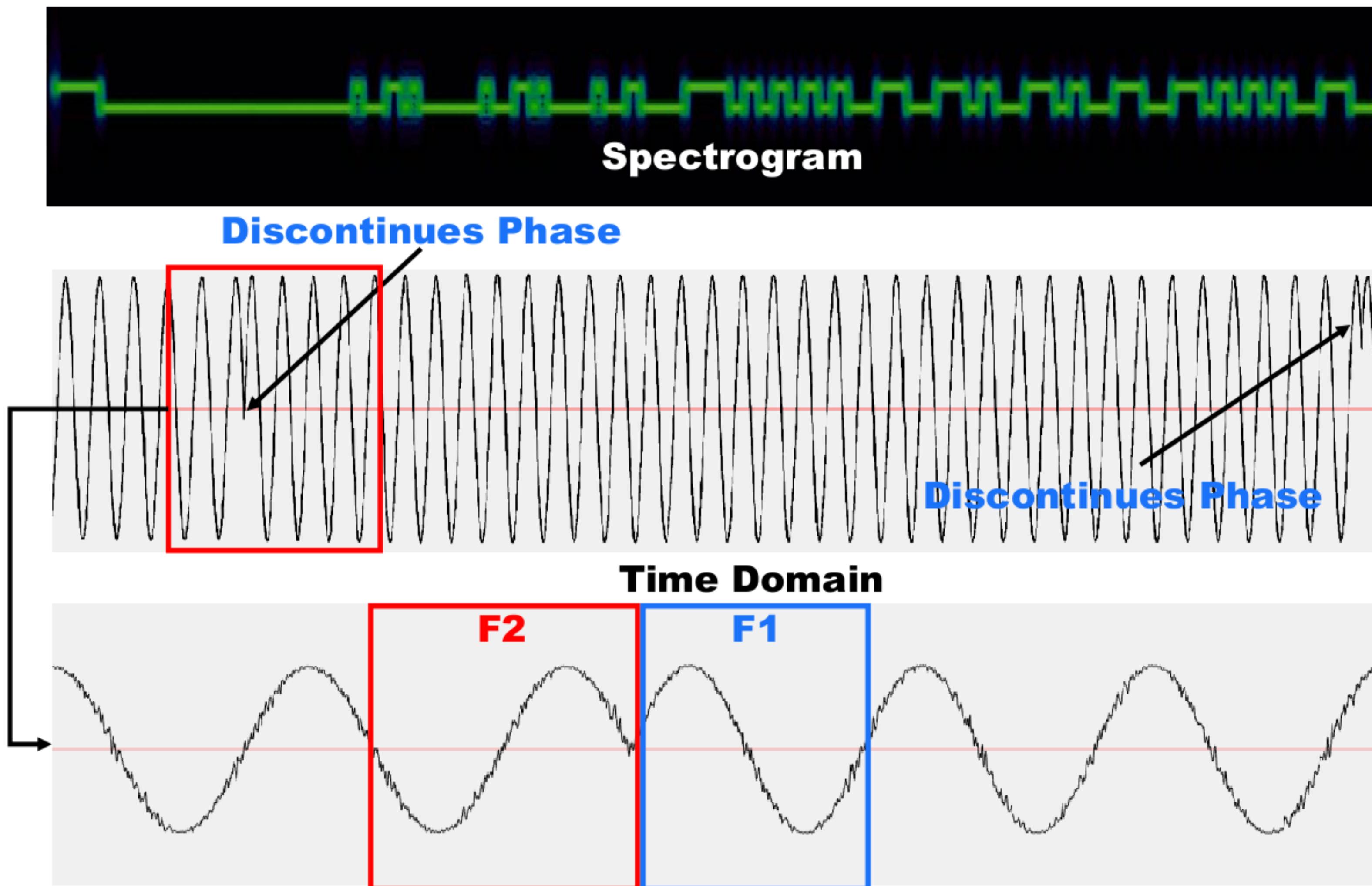
## □ The RF Lock Picking tool set : **CodeCatcher + CodeCrusher**

- **CodeCatcher** : The sniffer could include for demodulation, decoding, descrambling, decrypt, digital data recording
- **CodeCrusher** : The transmitter includes the reverse of the above functions, replaying signals or sending custom signals



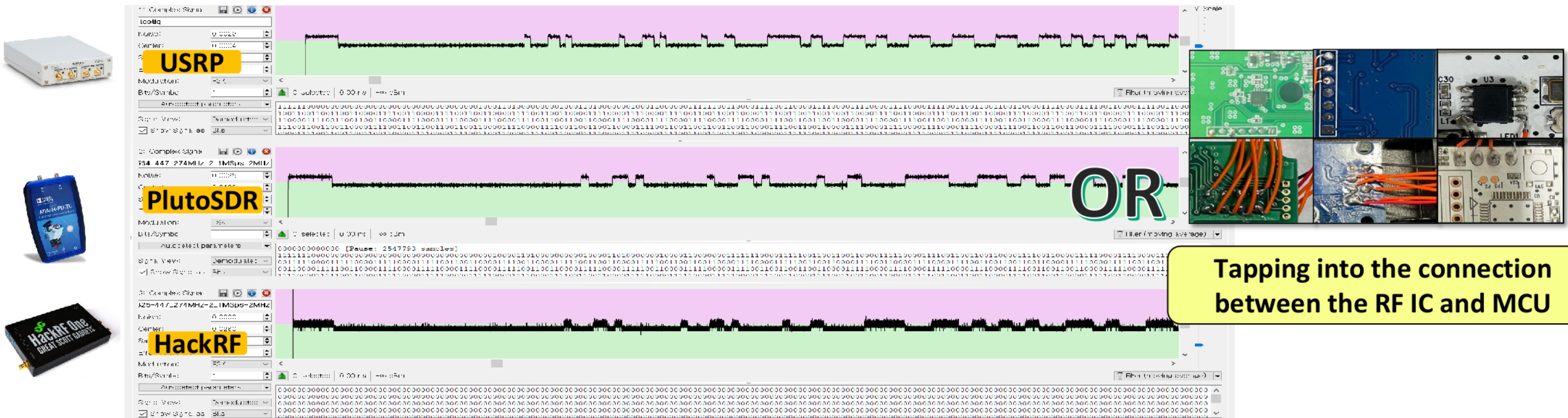
# Viewing Door Lock RF Signal

- Center frequency : 447.274 MHz or 447.261 MHz
- Modulation and Deviation : 2KHz Fix (Regulation)



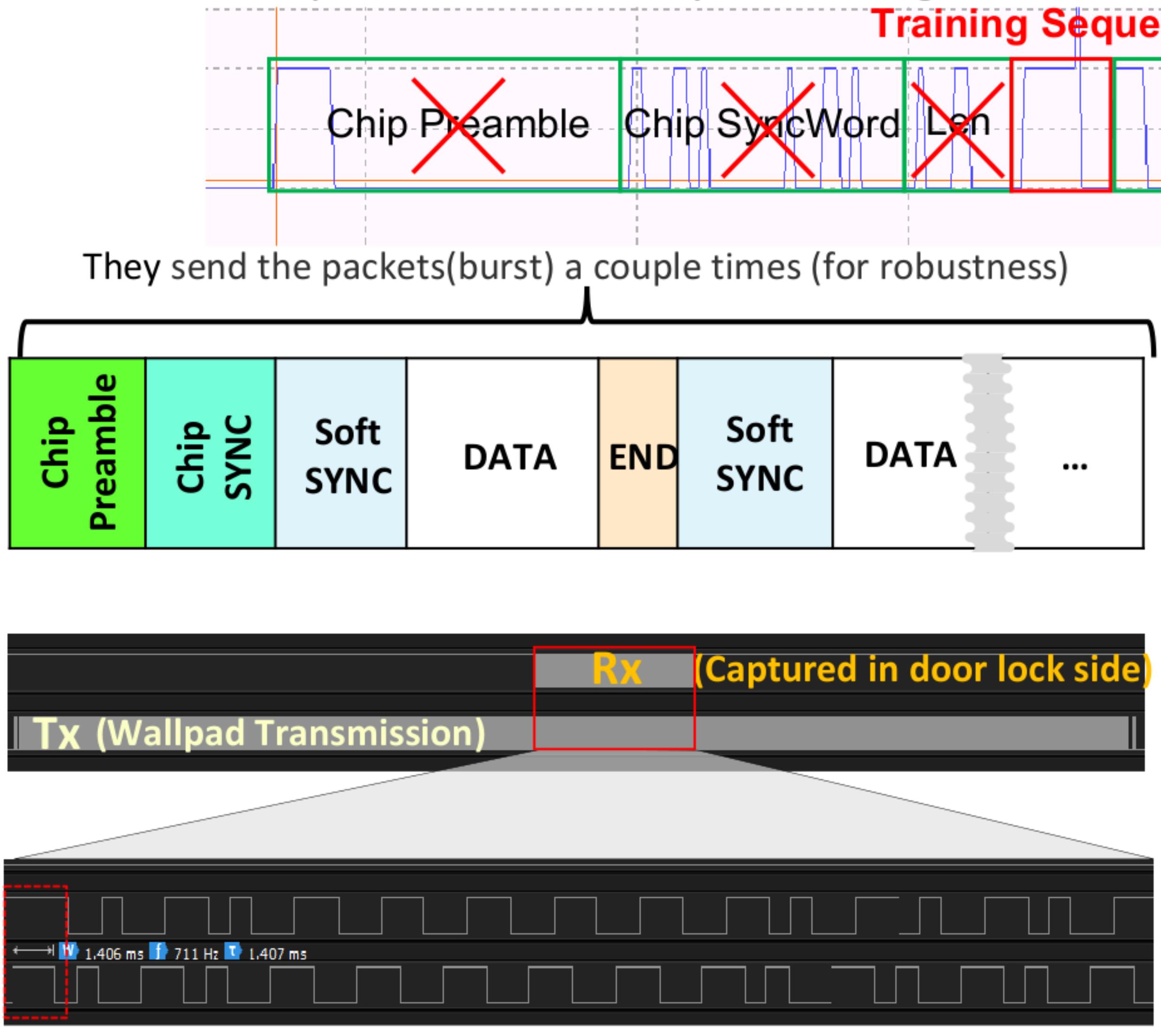
# Viewing RF Baseband

- URH can provide insights for base band analysis, without any RF knowledge
  - But, It may be impractical to continuously monitor and collect signals for our real-world attack
- Tapping into the connection between the RF IC and MCU is also best option
- PS. Small deviation frequency might cause interference in the IF signal
  - Note : Super heterodyne receiver, HackRF one, the optimal deviation frequency was found to be 100Khz or higher

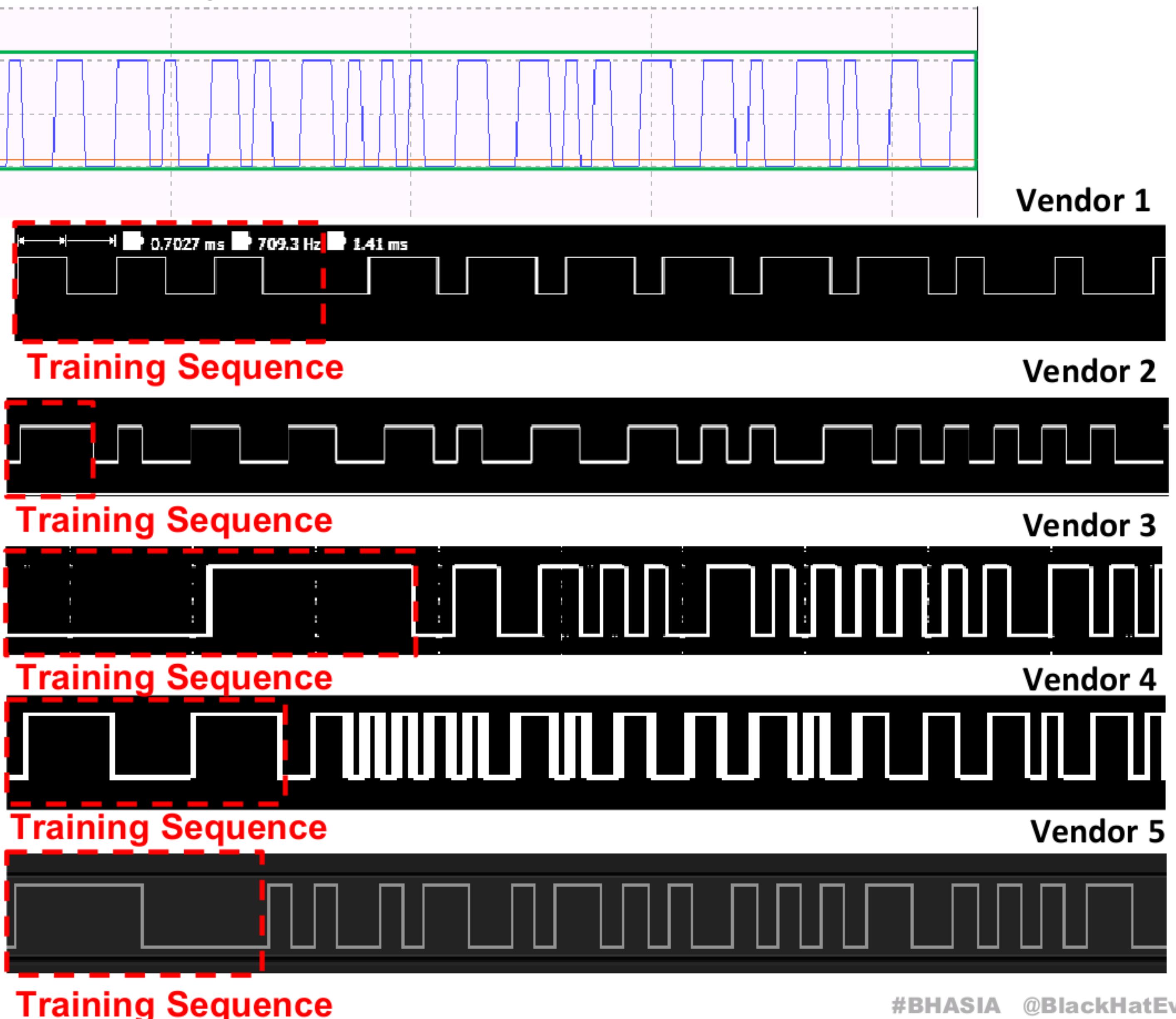


# Training Sequence for custom packet

- ❑ Vendor-specific training sequence for timing synchronization
  - The preamble and syncword generated by the RF chip handler are not used in door locks

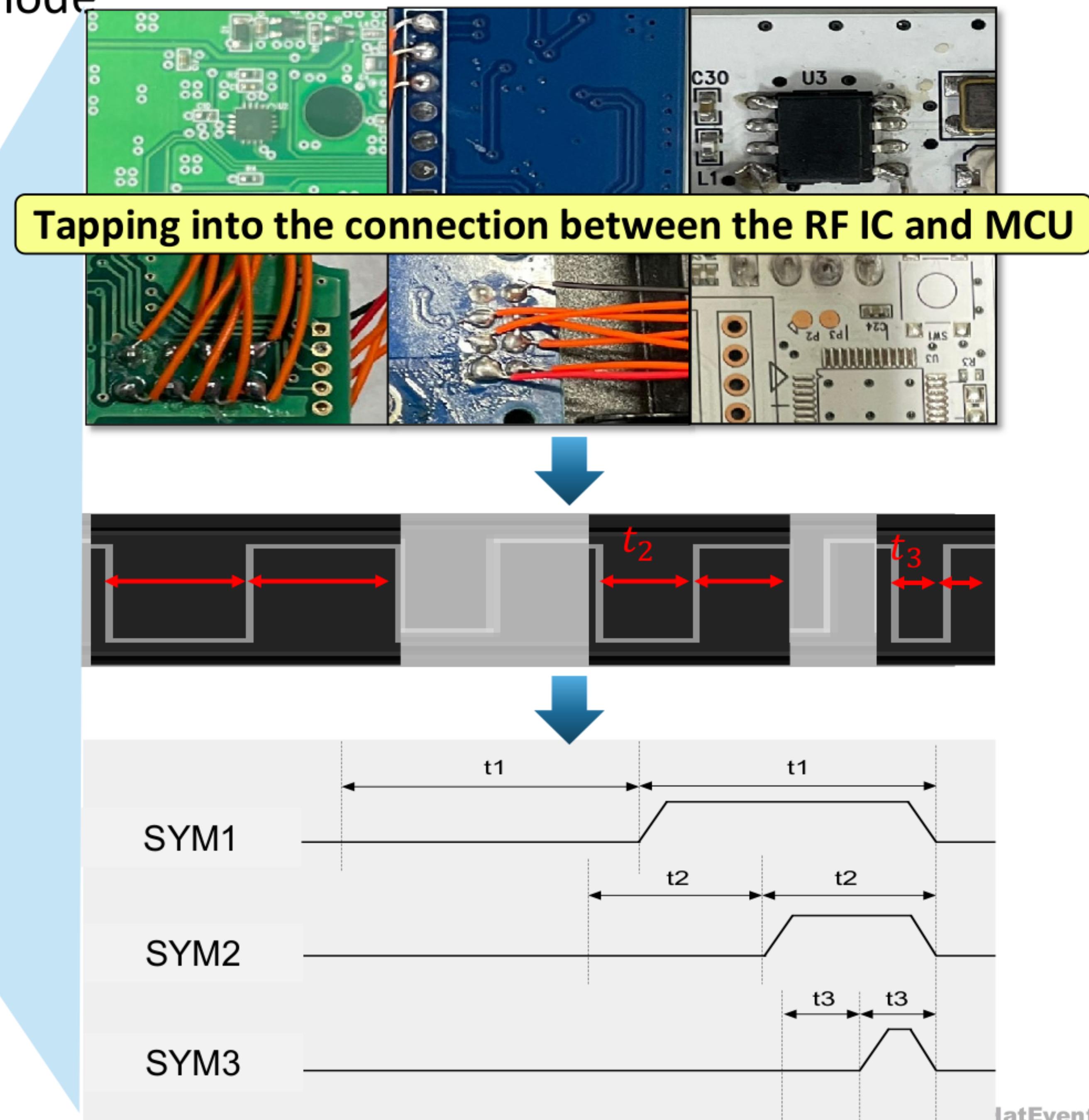
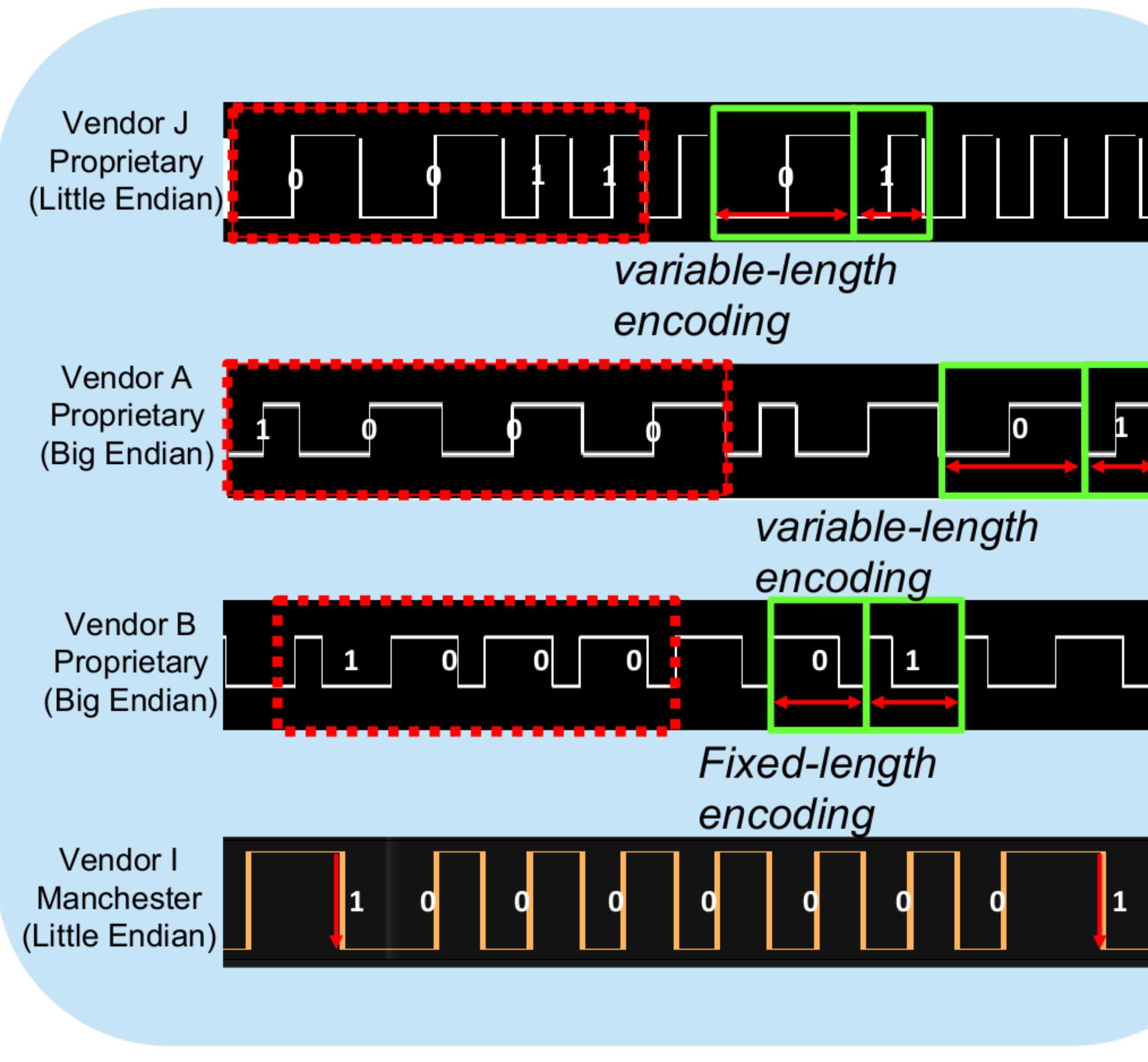


If there is a match, the data is accepted



# Proprietary Baseband Encoding

- Vendor-specific digital encodings in RAW transmission mode



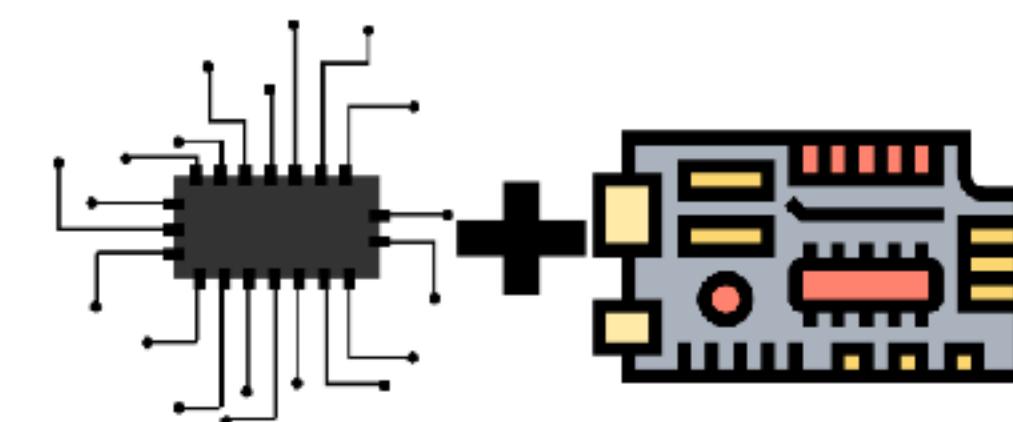
# Let's make it



imgflip.com

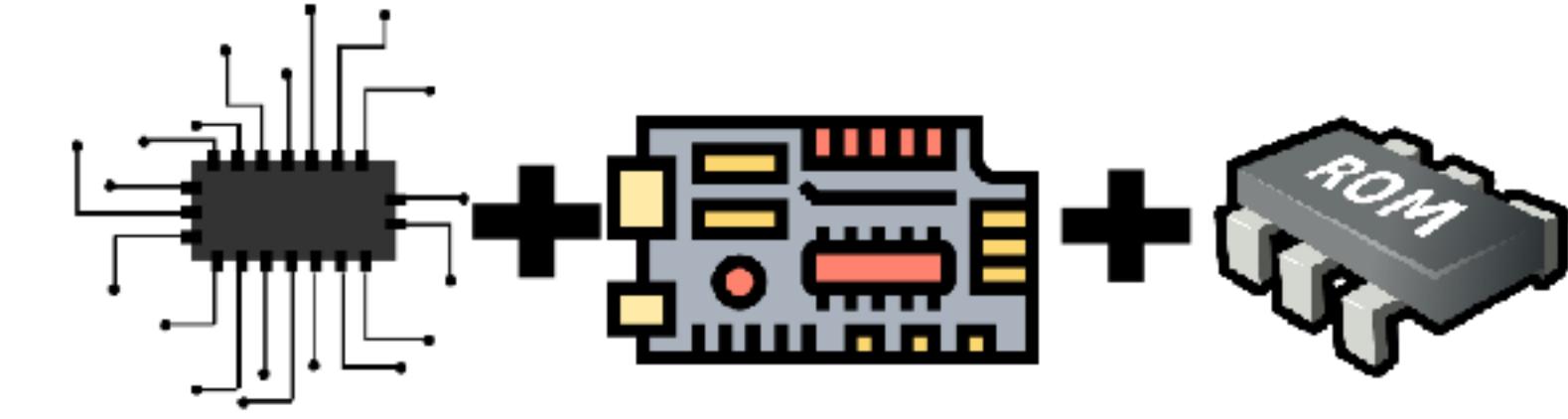
## Code Crusher Mini

\$10~(excluding battery)



## Code Catcher Mini

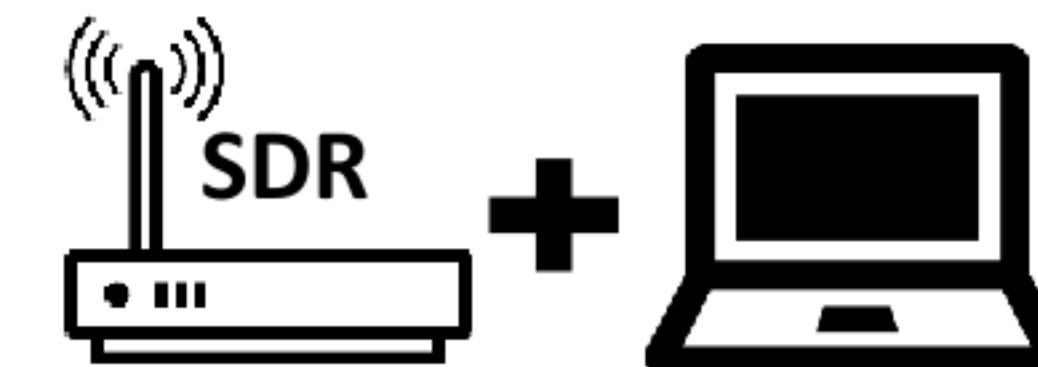
\$18~(excluding battery)



(Cost Effective & Easier to conceal & multiple deployment)

## SDR based Code Catcher and Crusher

\$200~(excluding Laptop)



(Performance & Flexible)

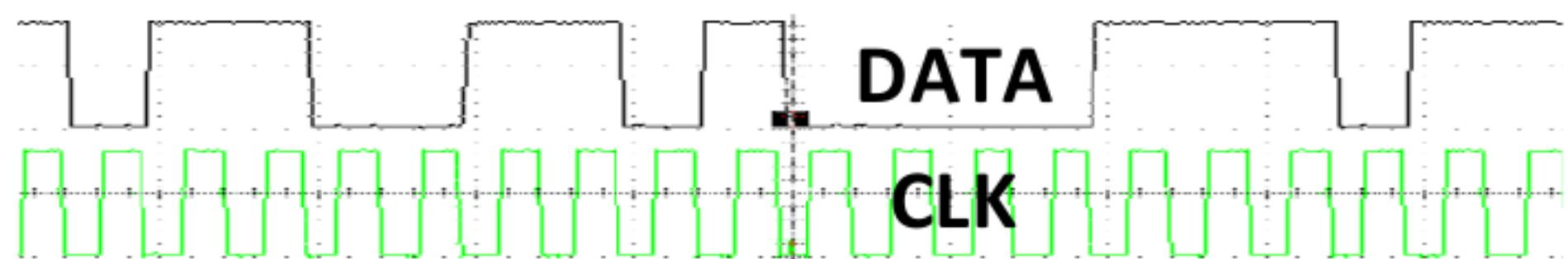
# Code Catcher Mini Configuring RF Chip for Direct Mode

- Validate through signal debugging for RF parameter decision

- According to Transmission Mode..

- Direct Mode(Aka. RAW transmission mode)

- Received RXDATA is output on a physical output pin in real-time



- Packet Mode

- The data is packaged into a specific format

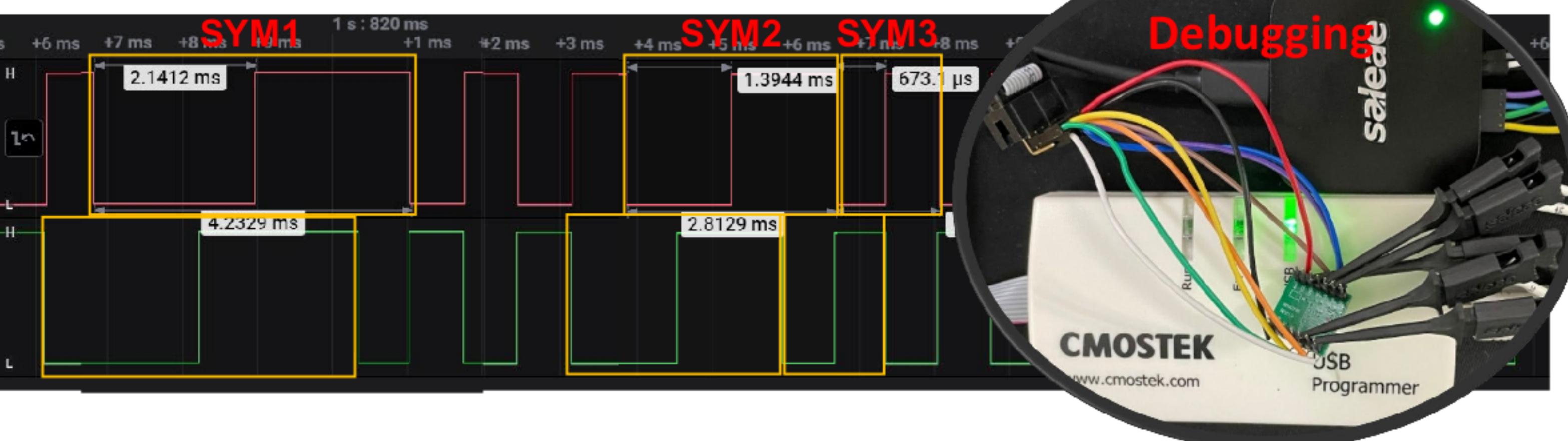


Choosing Chips that Support Direct Mode		
Direct / Packet (o)	Packet Mode Only (x)	Direct Mode Only (o)
CMOSTEK CMT2219B/CMT2300	Analog Device MAX4147	Melexis TH71101/71120
Silicon LABS Si4455x/443x	CMOSTEK CMT2217	Analog Device MAX7042
TI CC1000/1101	TI CC1125	
TDA 5150		

Index	Frequency	Xtal frequency	Demodulation	AGC	Data Rate	Deviation	Rx Xtal Int	Bandwidth	COH type	COH IR Range	AHC	AHC Method
0	433.920 MHz	26.0000 MHz	FSK	On	2.4 kbps	20.0 kHz	40 ppm	Auto-Select kHz	Counting	NA	On	Auto-Select
1	447.270 MHz	26.0000 MHz	FSK	On	2.4 kbps	2.0 kHz	40 ppm	Auto-Select kHz	Counting	NA	On	Auto-Select
2	447.242 MHz	26.0000 MHz	FSK	On	2.6 kbps	2.0 kHz	40 ppm	Auto-Select kHz	Counting	NA	On	Auto-Select
3	447.242 MHz	26.0000 MHz	FSK	On	2.6 kbps	2.0 kHz	40 ppm	Auto-Select kHz	Counting	NA	On	Auto-Select
4	447.242 MHz	26.0000 MHz	FSK	On	2.6 kbps	2.0 kHz	40 ppm	Auto-Select kHz	Counting	NA	On	Auto-Select
5	447.274 MHz	26.0000 MHz	FSK	On	2.6 kbps	2.0 kHz	40 ppm	100 kHz	Counting	NA	On	Auto-Select

Chip Demod.  
Signal

Transmitter  
B.B. Signal



```
The following are the Register contents:
Addr Value
0x00 0x40
0x01 0x22
0x02 0xL+
0x03 0x05
0x04 0x06
0x05 0x00
0x06 0x00
0x07 0x11
0x08 0x80
0x09 0x39
0x0A 0x01
0x0B 0x00
0x0C 0x62
0x0D 0x0F
```

RF Config. Value

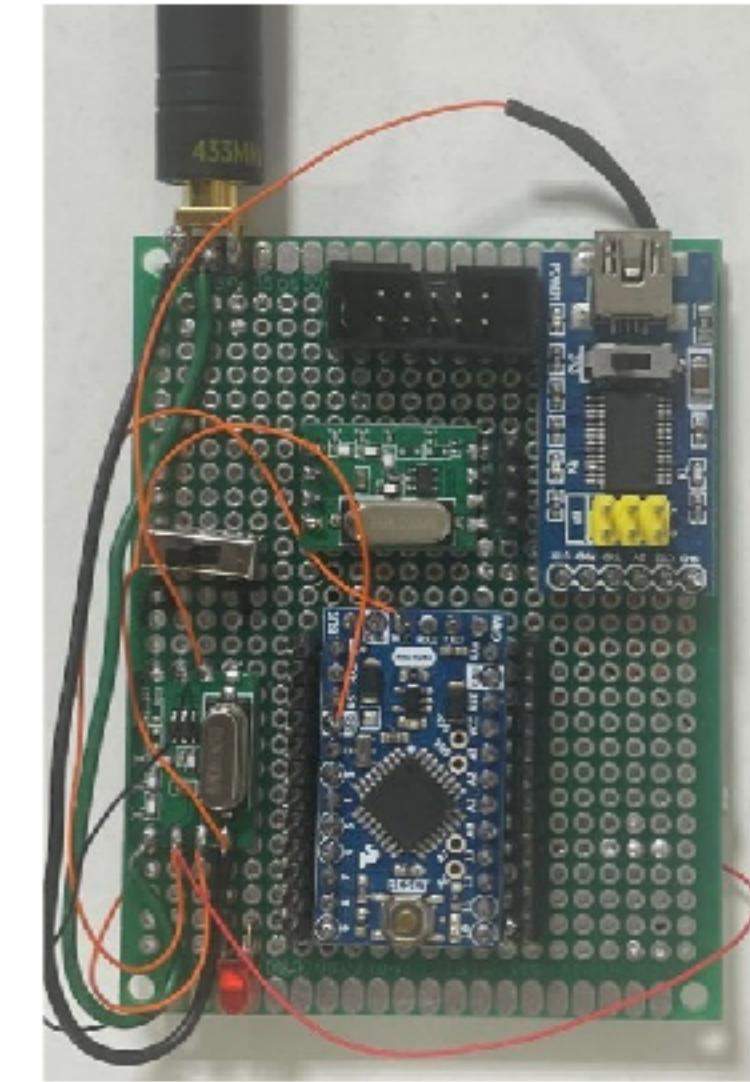
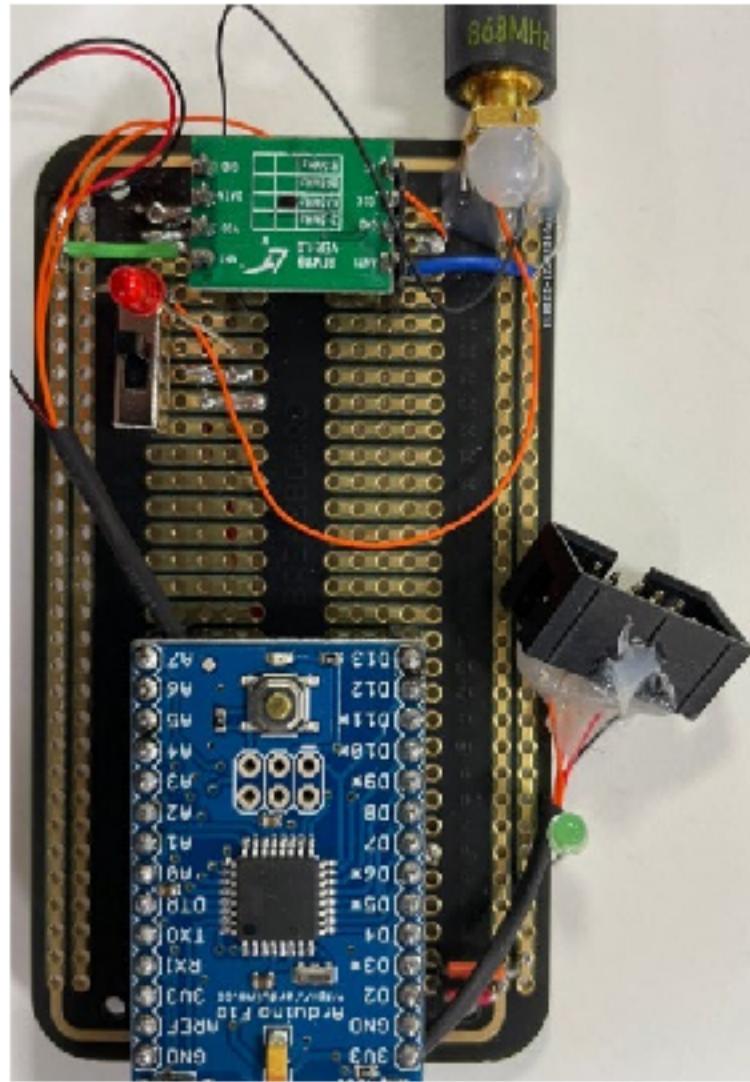
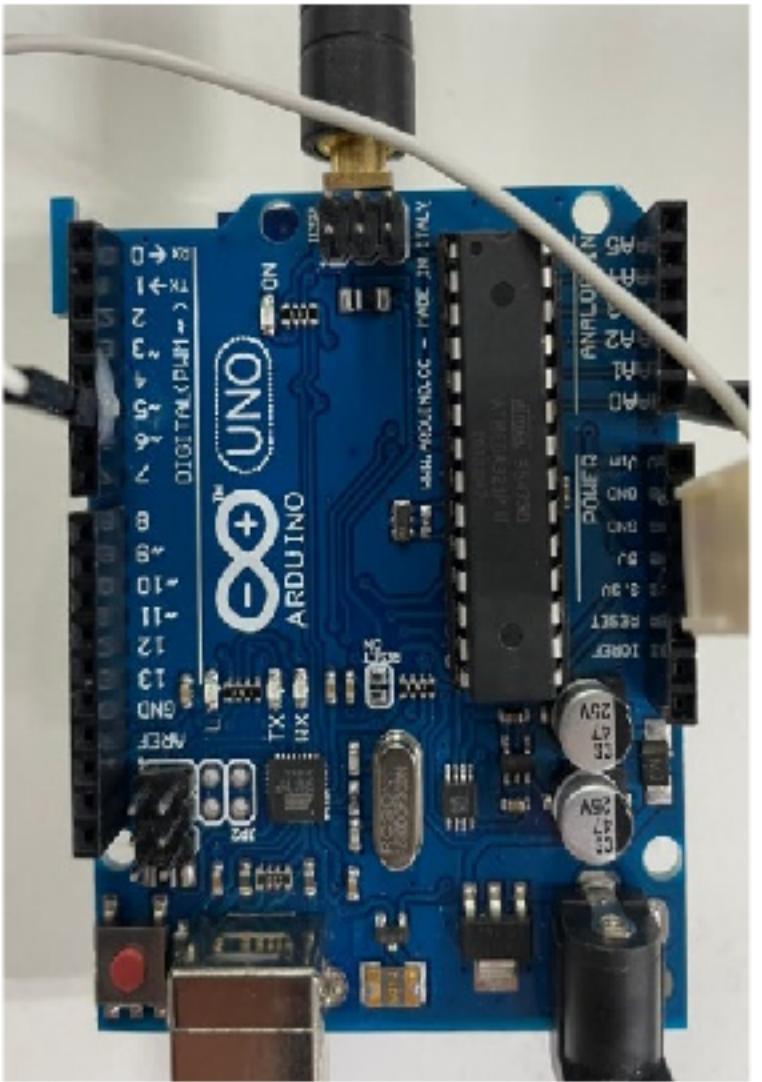


When analog debugging is complete,  
export the configuration file

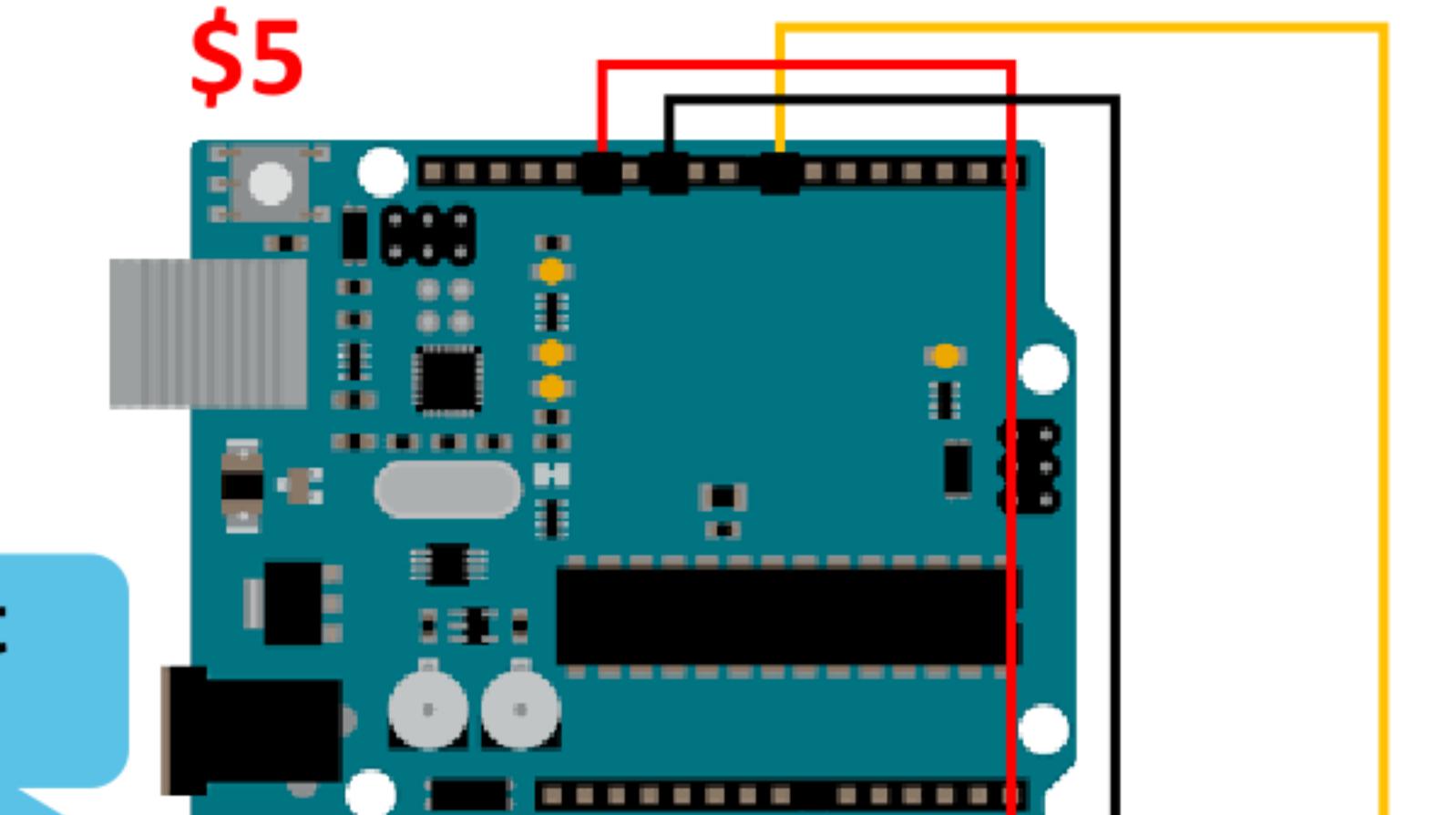




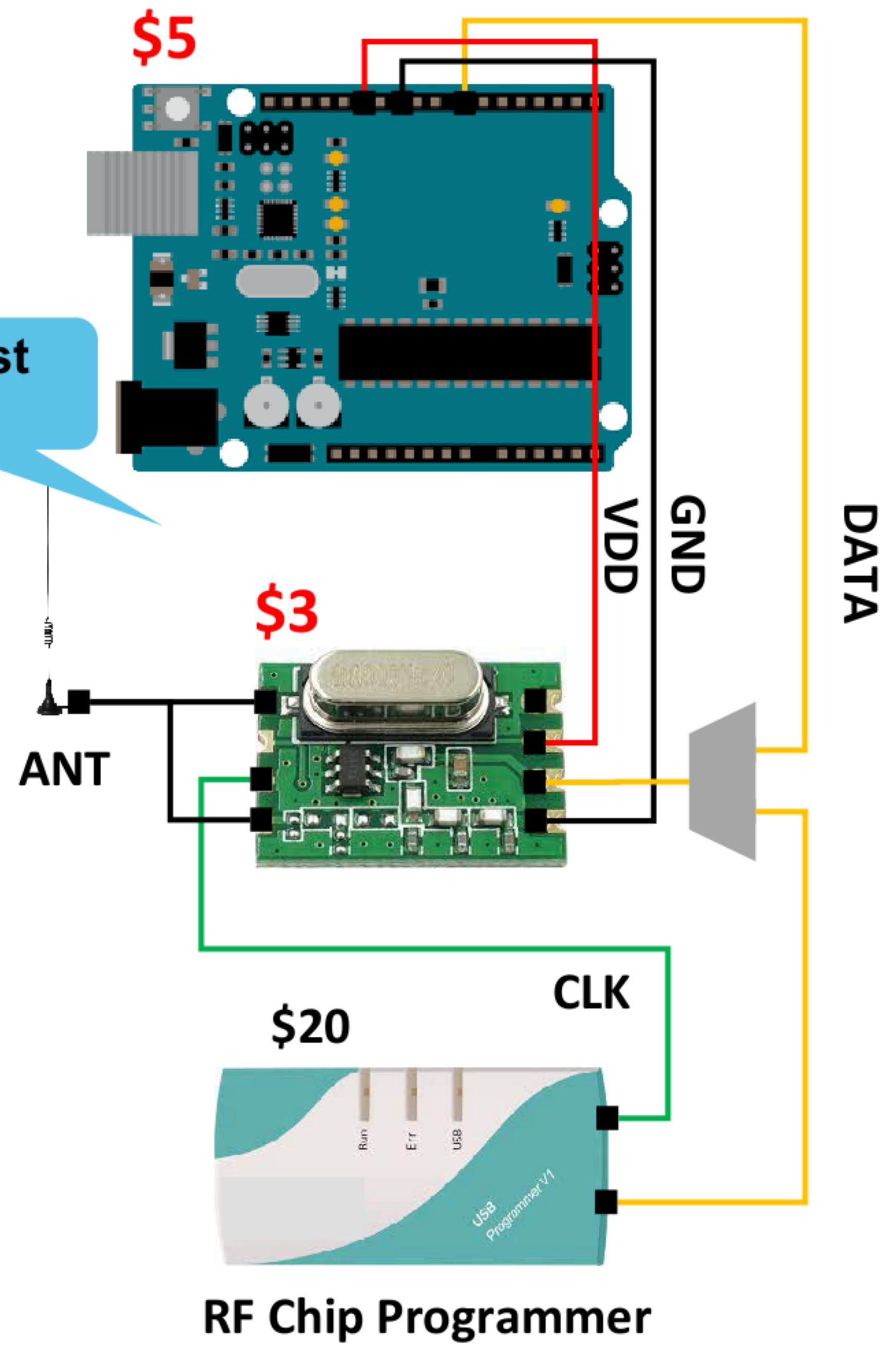
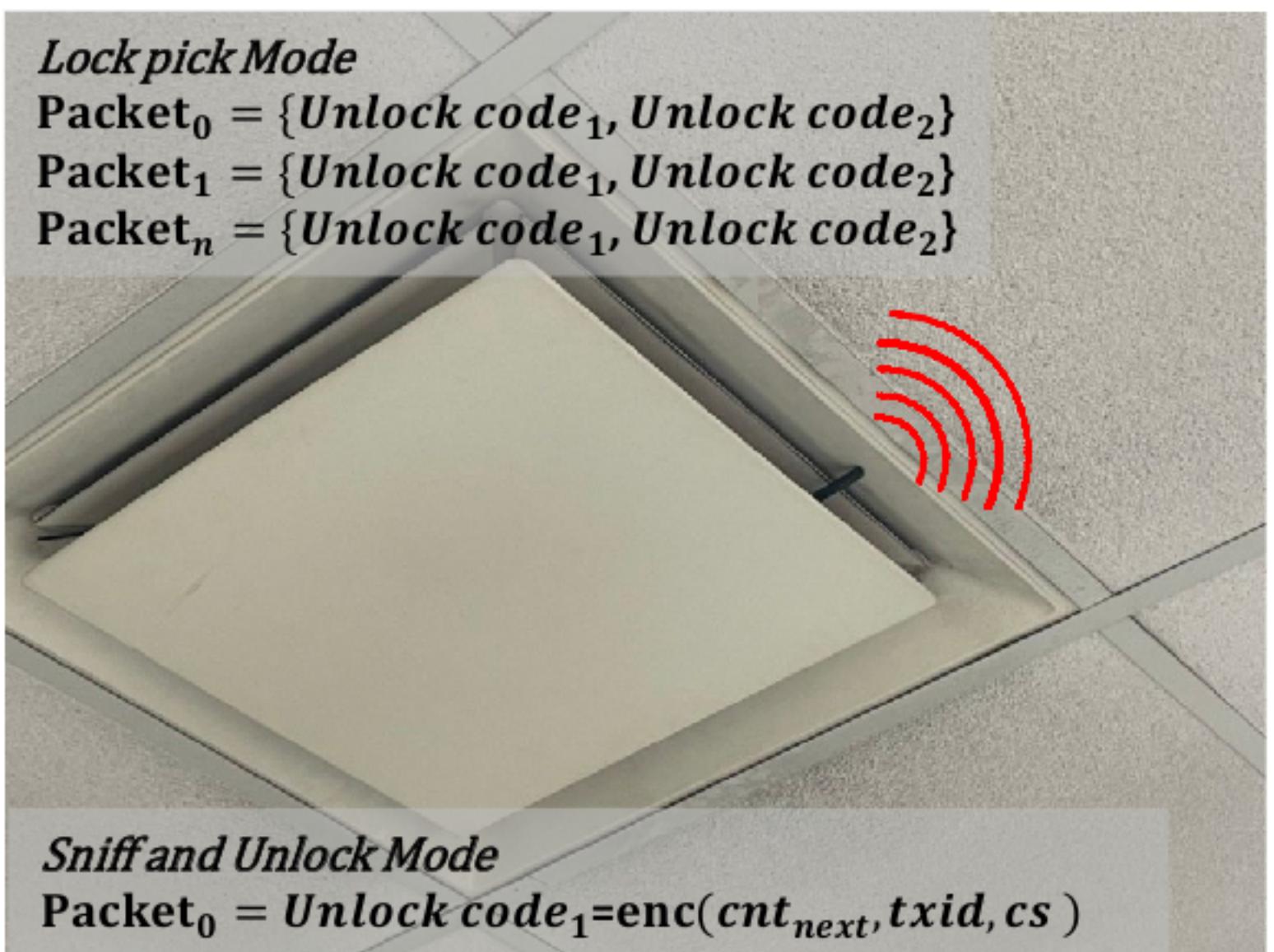
# Code Crusher Mini : RF IC Based Tx



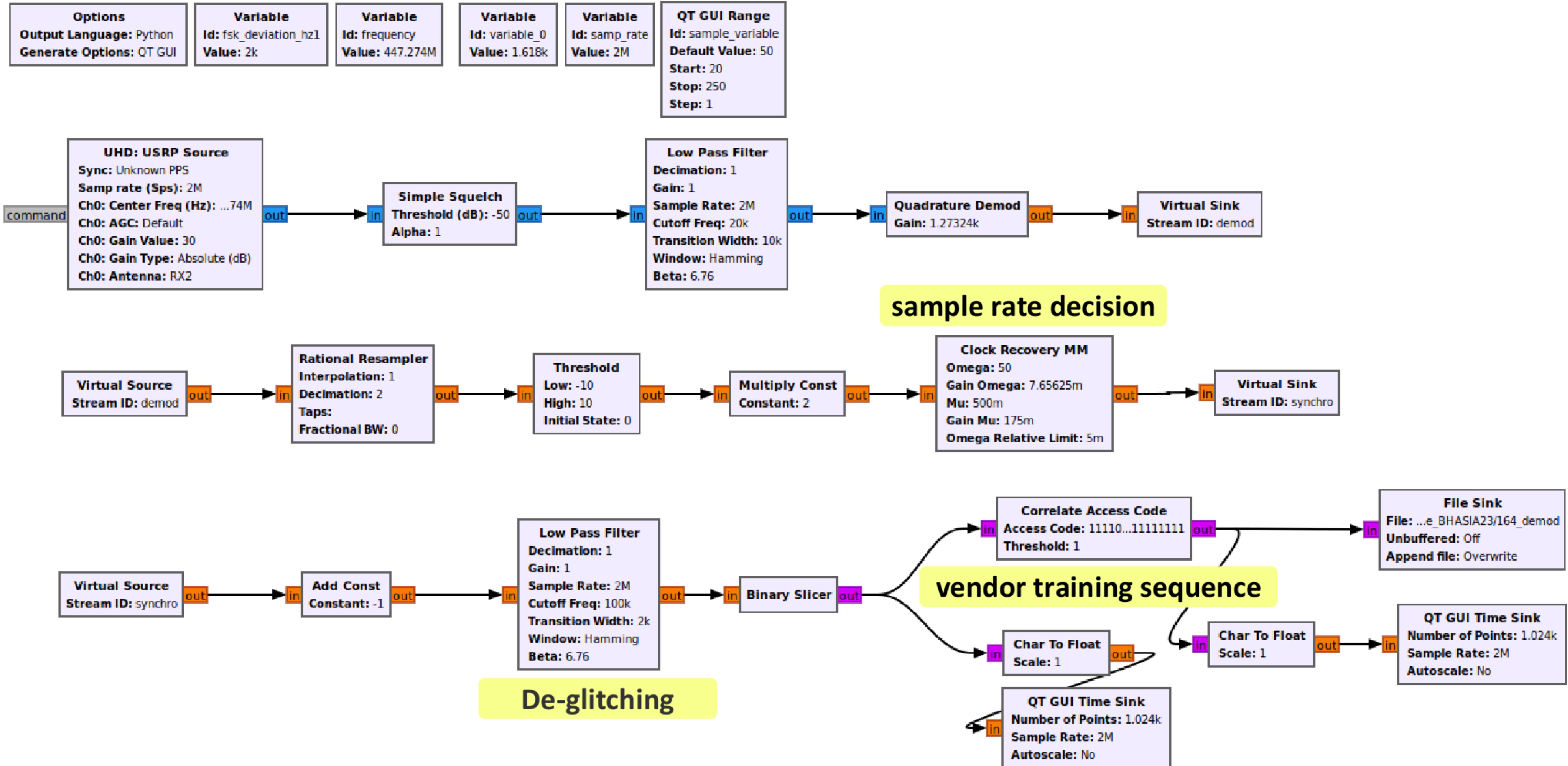
Total Cost  
< \$10



"It can be attached near by target, and the operating time can be extended as much as desired depending on the Lithium Polymer Battery that is installed."

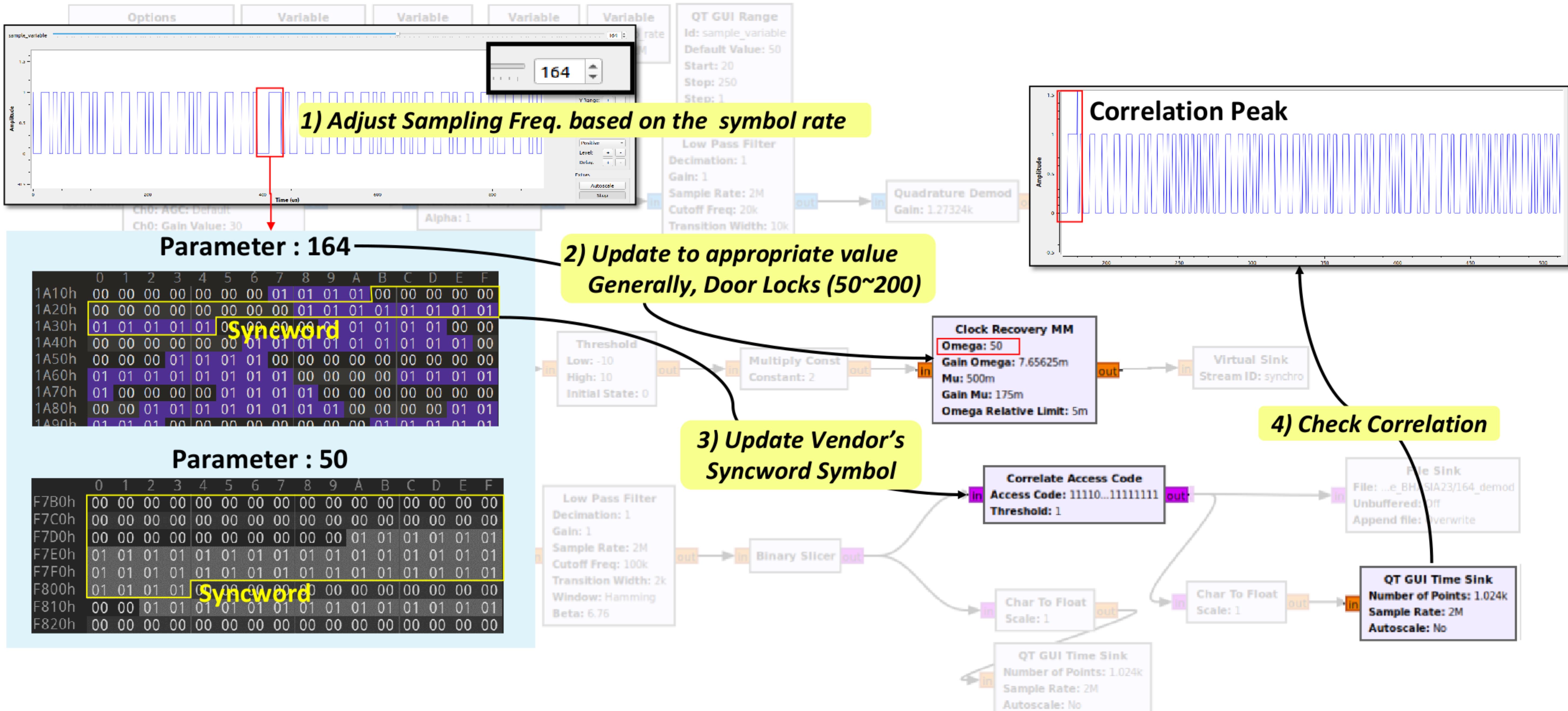


# SDR based Code Catcher gr-block for Door Lock RF Sniffing

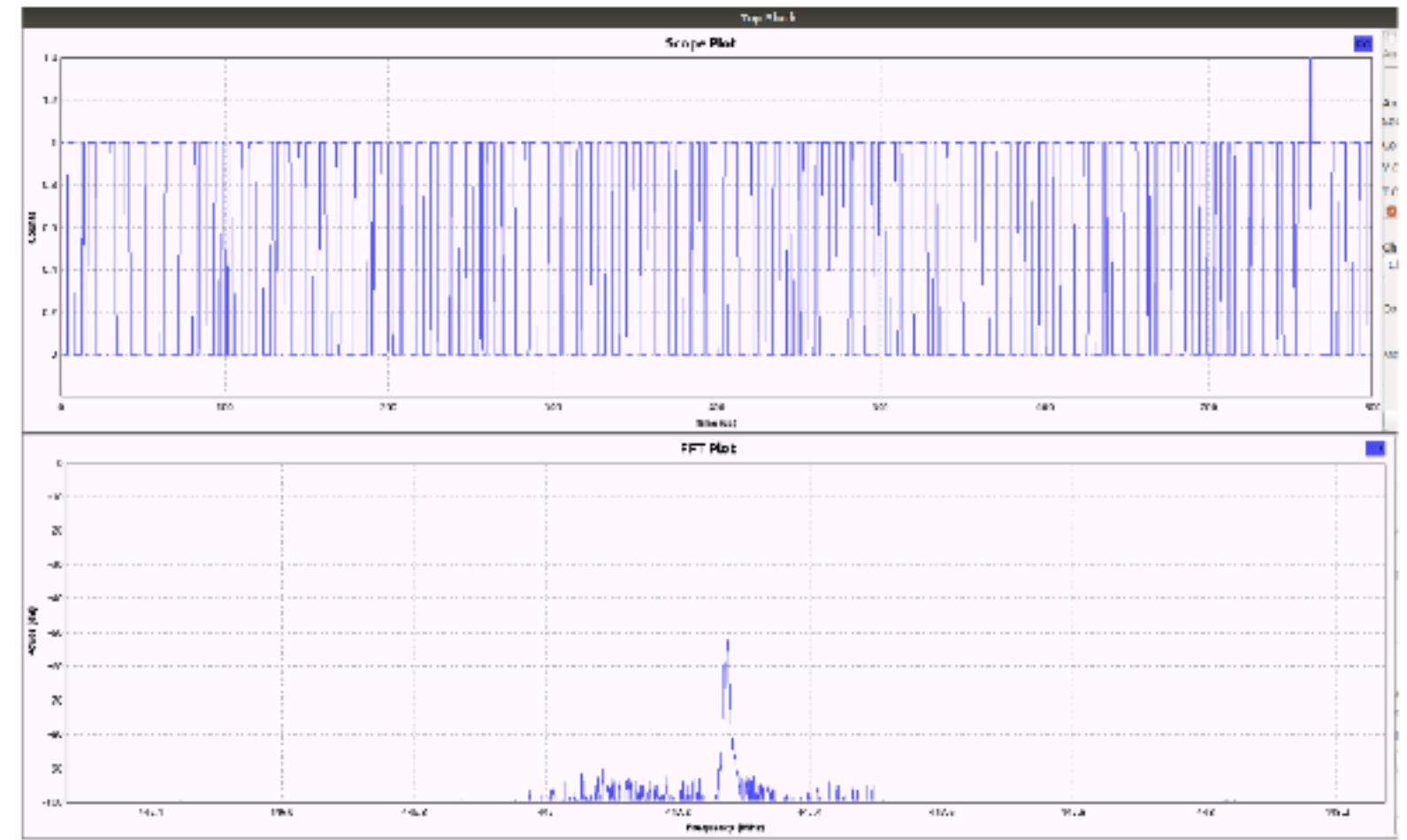


# SDR based Code Catcher configuration for Sniffing

numbers indicate the order of step.



# SDR based Code Catcher Decode and Decrypt



Capture and Demodulation

**BIN**

```

1 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F
00 00 00 00 01 01 01 01 01 01 01 00 00 00 00 00
00 00 00 00 00 01 01 01 01 01 01 01 01 01 00 00
00 00 00 00 00 01 01 01 01 01 01 01 01 01 01 00
00 00 00 01 01 01 01 00 00 00 00 00 00 00 00 00
00 00 00 00 01 01 01 01 01 01 01 01 01 01 01 01
01 02 00 00 00 00 00 00 00 00 01 01 01 01 01 01
01 01 00 00 00 00 00 00 00 00 01 01 01 01 01 01
01 01 01 00 00 00 00 00 00 00 01 01 01 01 01 01
01 01 01 01 00 00 00 00 00 00 01 01 01 01 01 00
00 00 00 01 01 01 00 00 00 00 01 01 01 01 01 01
00 00 00 00 00 00 00 00 00 01 01 01 01 01 01 01
01 00 00 00 00 00 00 00 00 01 01 01 01 01 01 01
01 01 00 00 00 00 00 00 00 00 01 01 01 01 01 01

```

Demodulated Signal

shaftmon@shaftmon-system-Product-Name:~/desktop/demod\_decode\_BHASIA23\$  
./doorloc sniff and decrypt.exe

**SNIFF&DEC**

```

preamble detected 1
Received Message :E 18 E0 87 1D E3 36 B5 1C D4 13 B5 BA BE F2 96 7B
First Block Key :F1 F2 F3 F4 F5 F6 F7 F8 F9 FA FB FC FD FE FF 00
Second Block Key :D2 18 49 3F 28 87 D3 52 87 2C 21 65 BC AA 37 49

Encrypted Message :18 E0 87 1D E3 36 B5 1C D4 13 B5 BA BE F2 96 7B
3::f195a52389010000000000000000d8
Decrypted Message :F1 95 A5 23 89 01 00 00 00 00 00 00 00 00 DB

preamble detected 2
Received Message :E 18 E0 87 1D E3 36 B5 1C D4 13 B5 BA BE F2 96 7B
First Block Key :F1 F2 F3 F4 F5 F6 F7 F8 F9 FA FB FC FD FE FF 00
Second Block Key :D2 18 49 3F 28 87 D3 52 87 2C 21 65 BC AA 37 49

```

Decode and Decrypt

Transmitter A

```

preamble detected 7
Received Message :11 DE 9F E7 33 D6 D8 B2 83 A9 3C 90 B5 7B 71 A1 61
First Block Key :11 12 13 14 15 16 17 18 19 1A 1B 1C 1D 1E 1F 20
Second Block Key :8A E9 E1 36 4D 45 3E 51 CF 27 ED 82 FB EC 5F AE

Encrypted Message :DE 9F E7 33 D6 D8 B2 83 A9 3C 90 B5 7B 71 A1 61
Decrypted Message :11 95 A4 1B DB 01 00 00 00 00 00 00 41

```

Sync Counter TXID: 95 A4 1B DB

Checksum

Transmitter B

```

preamble detected 16
Received Message :44 1 1F 73 92 85 42 79 2 3C EB 13 7 79 C4 86 D8
First Block Key :44 45 46 47 48 49 4A 4B 4C 4D 4E 4F 50 51 52 53
Second Block Key :21 D8 FF 33 63 30 71 ED E3 6B 4E D3 29 67 0F 0D

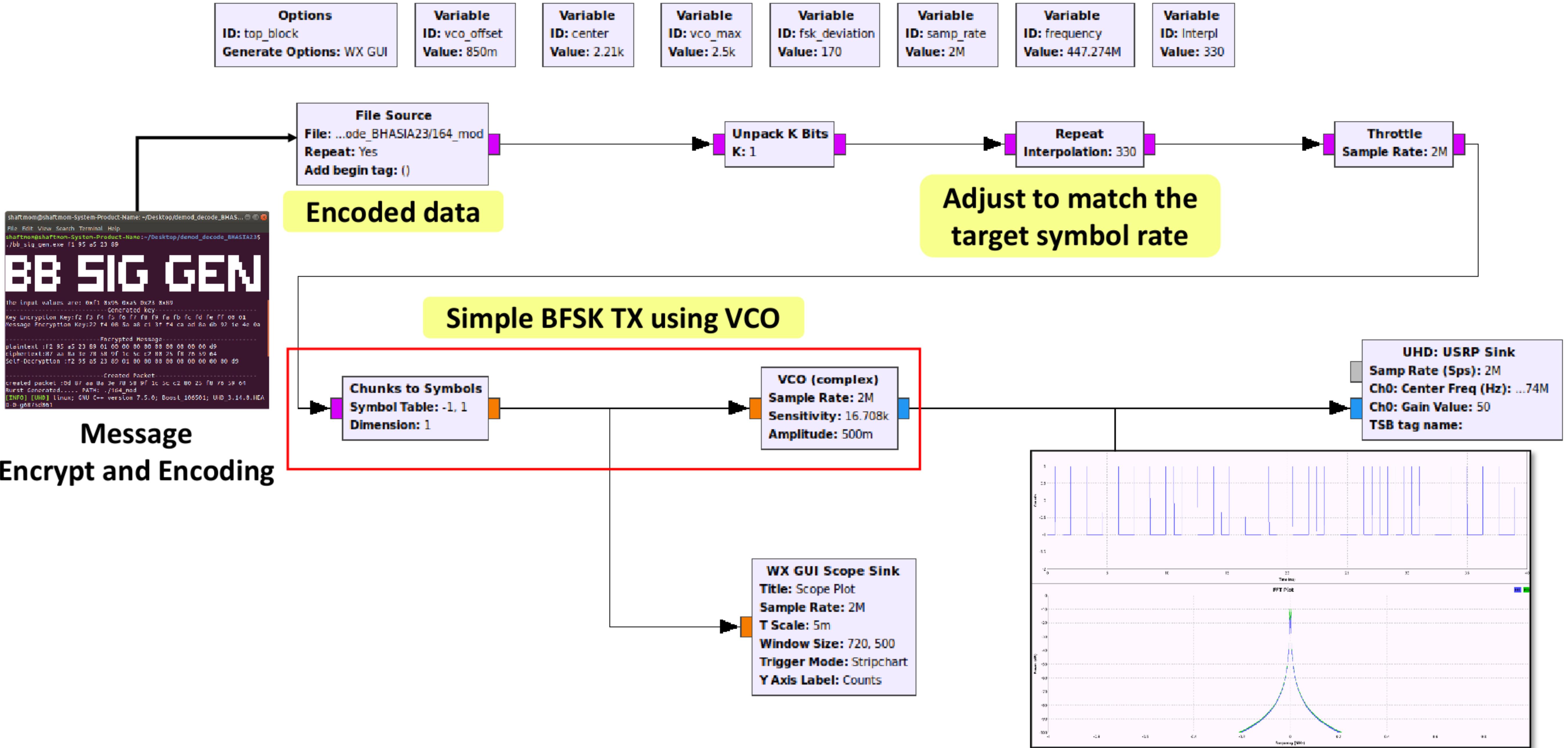
Encrypted Message :01 1F 73 92 85 42 79 02 3C EB 13 07 79 C4 86 D8
Decrypted Message :44 95 A5 28 AE 01 00 00 00 00 00 00 55

```

Sync Counter TXID: 95 A5 28 AE

Checksum

# gr block based Tx : Encode and Transmit





## Practical Attack

---

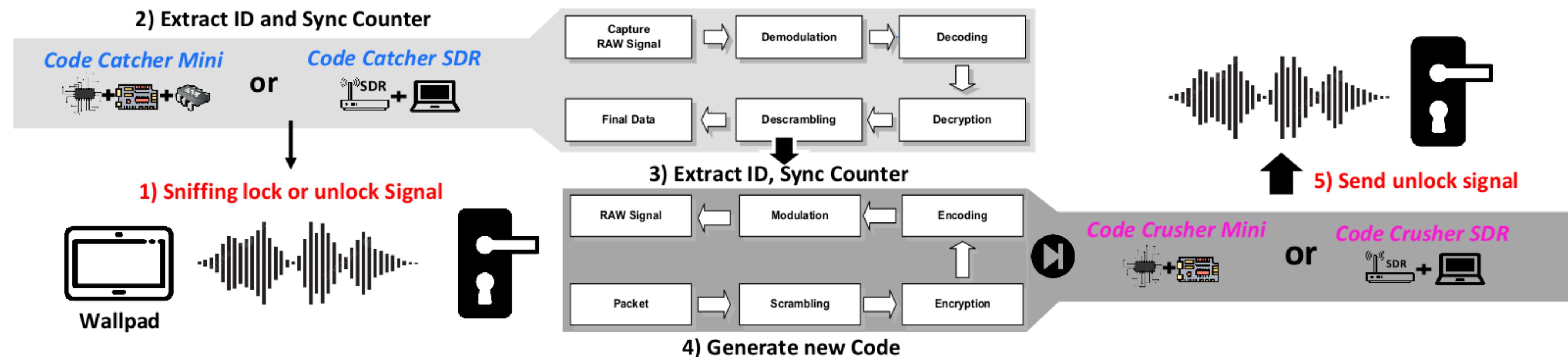
### The Art of RF Lock Picking

# Two Types of Lock Picking

## □ Type1: Sniff and unlock

- The attacker extracts the “ID” and “synchronous counter” and generates a new code

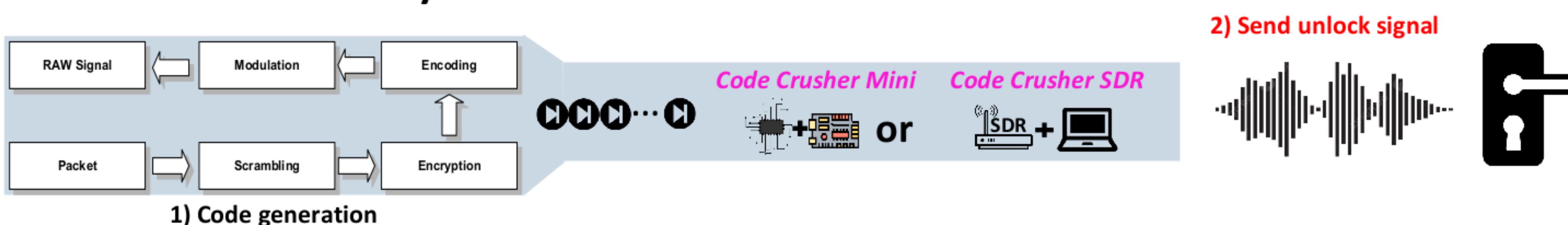
※ Attacker know “current synchronous counter” value, and is expected to know the next value



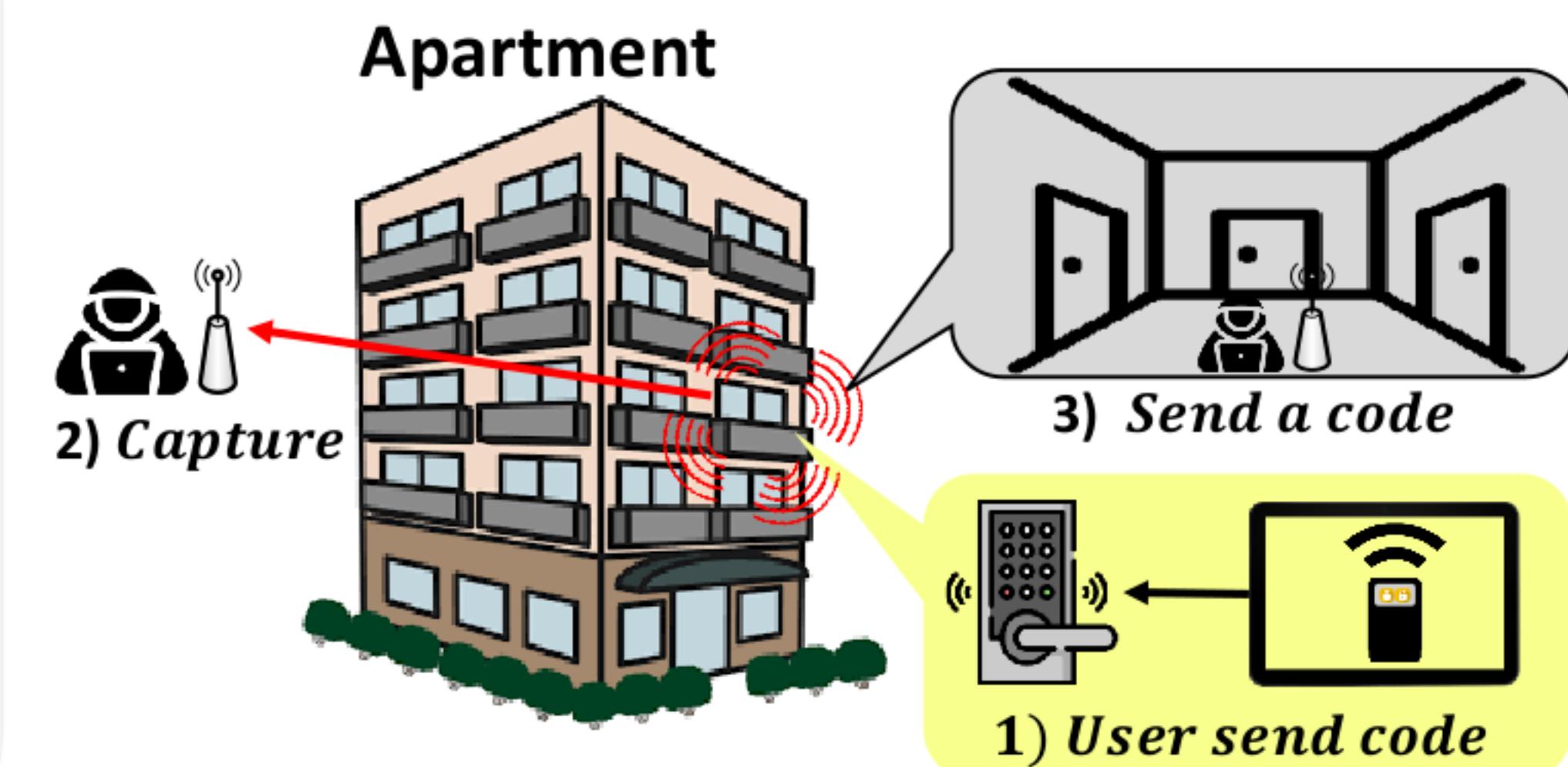
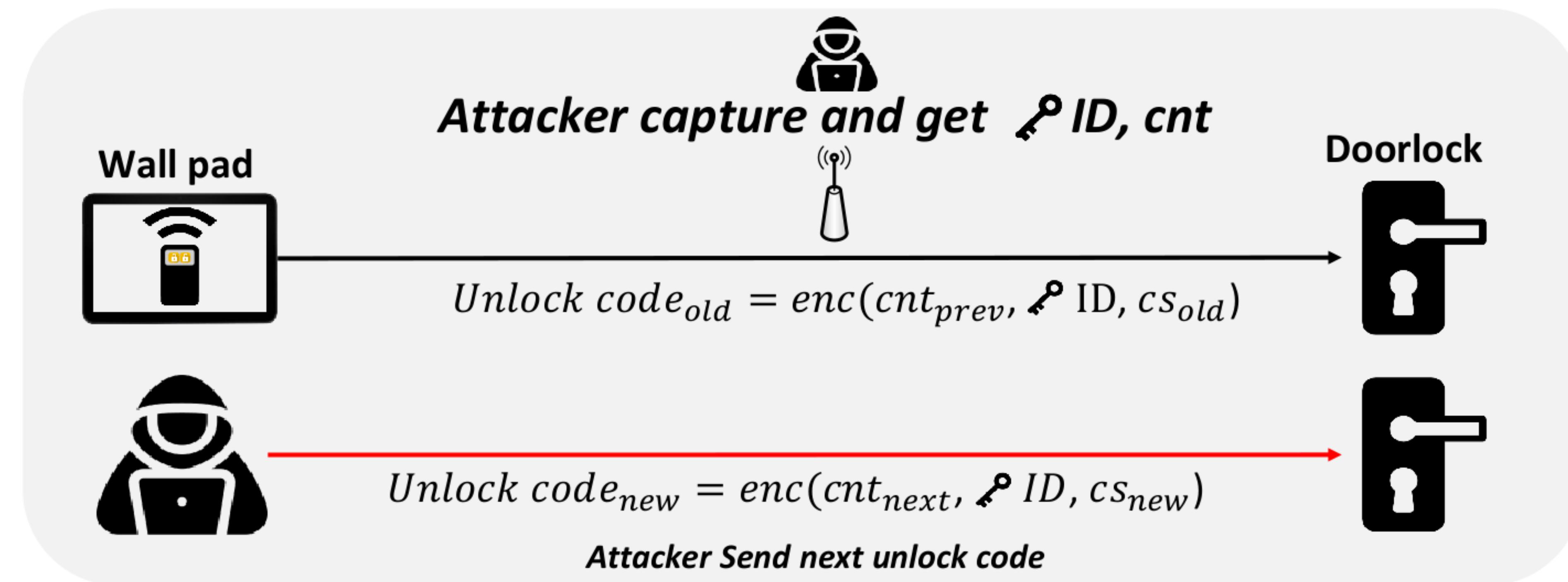
## □ Type2: Lock picking - Without Sniffing (=Brute Force Attack)

- The attacker generates a new code by only changing the ID(=Serial Number) value

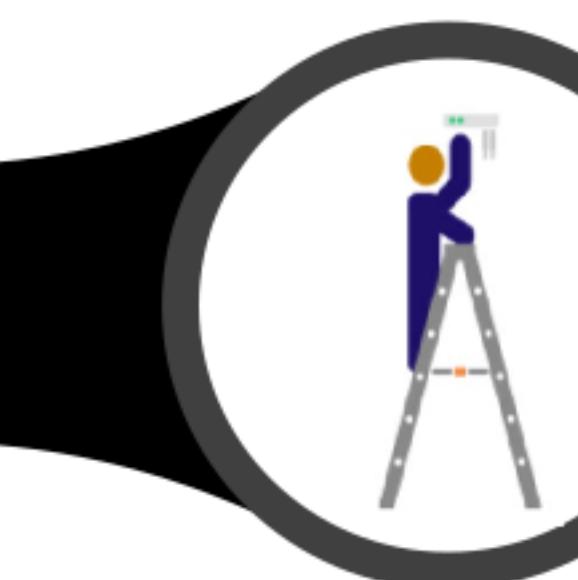
※ Attacker don't know “current synchronous counter” value



# Lock Picking : Sniff and Unlock



*Set the reception range*



*Deploying the sniffer*



*Capture Signal  
And Extract Data*



*Send Signal*



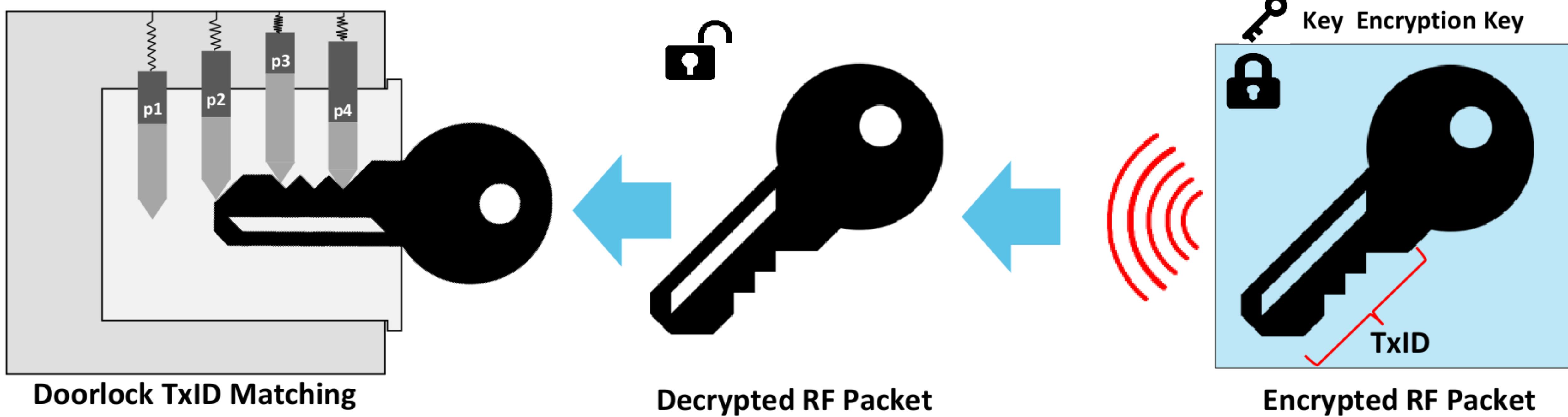
*Unlock the door*



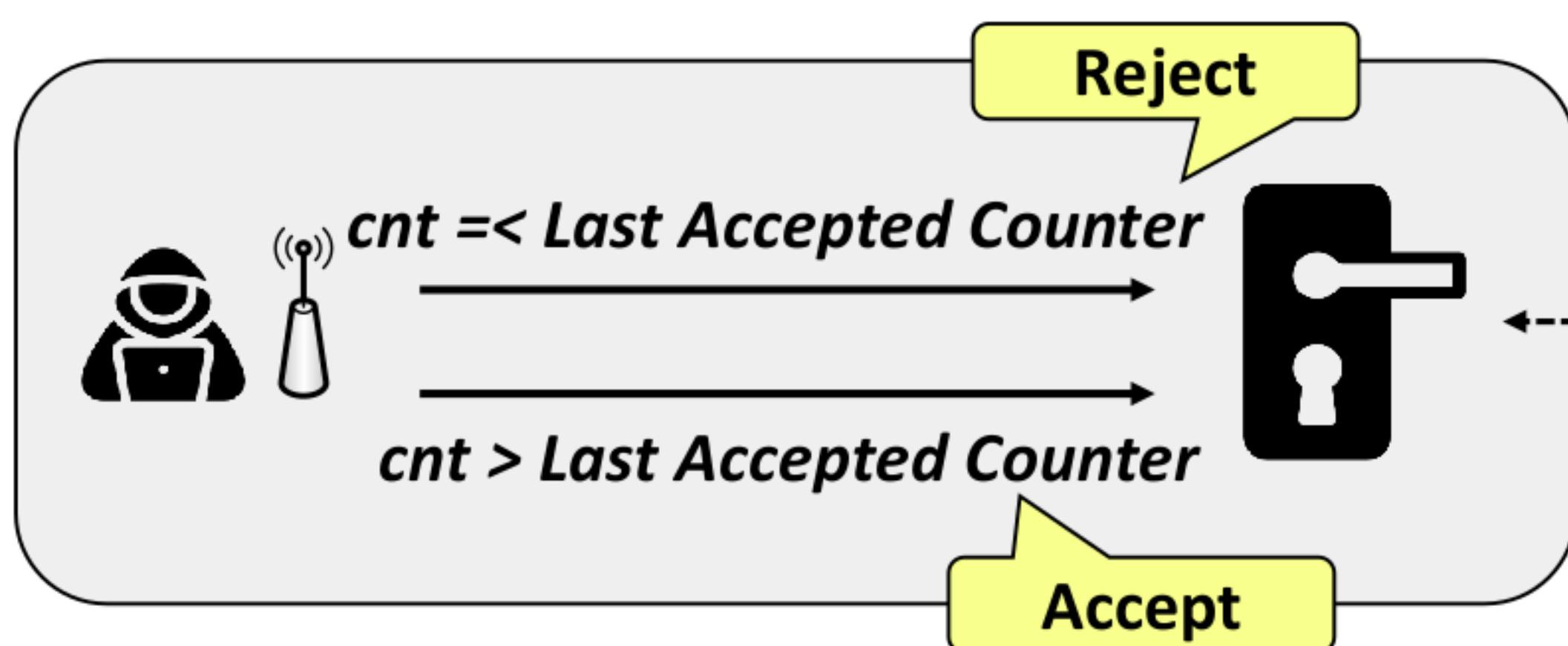
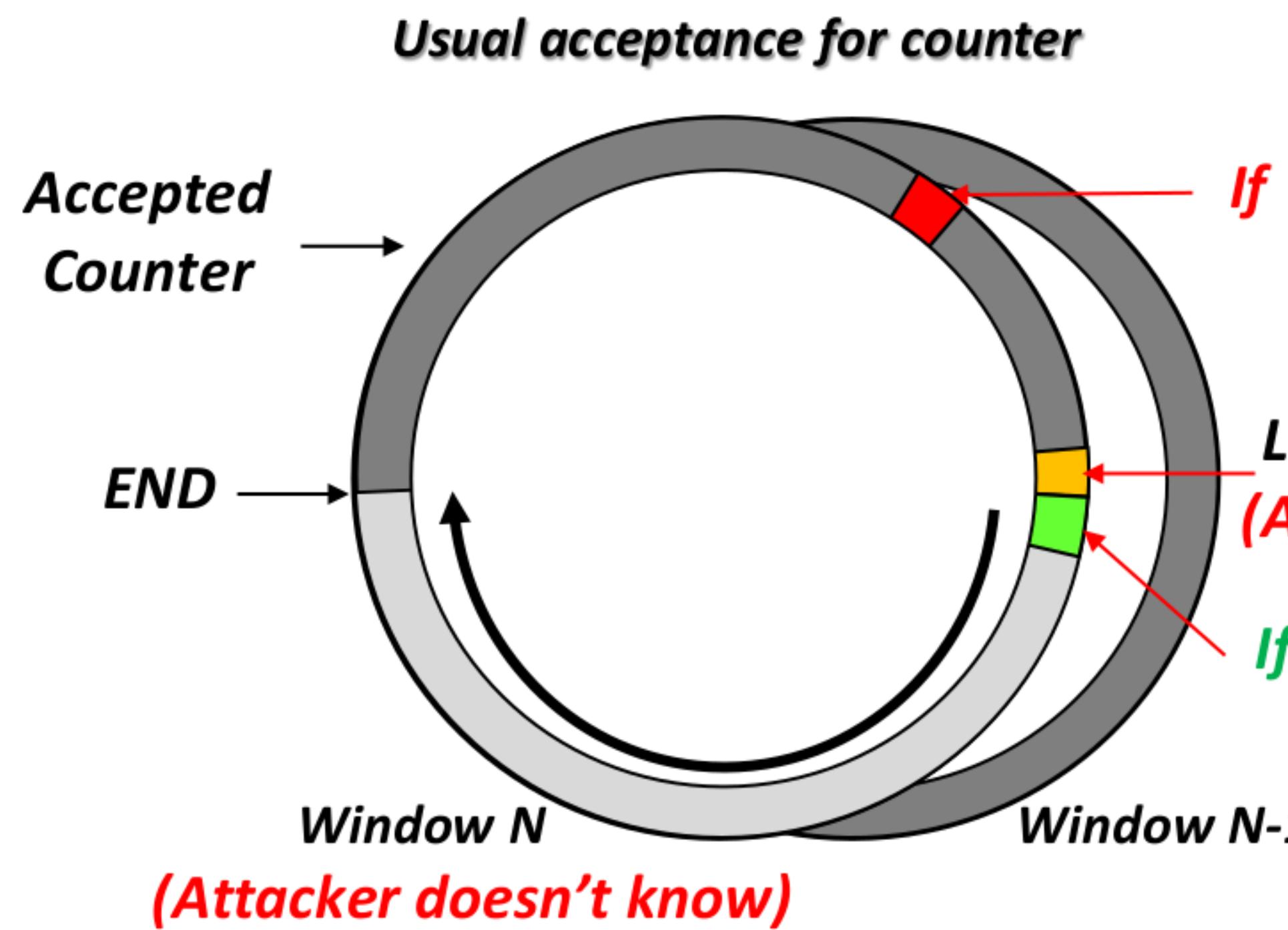
# Lock Picking w/ Sniffing Demo Video

# Lock Picking : without sniffing

- ❑ This attack involves unlocking the victim's door without the need for any RF sniffing
- ❑ The attack is to find a Serial Number(=TxID) that matches the one used by the door lock.
- ❑ The attacker's message must always be accepted, even if the current counter value is unknown



# Considering Attacker's Counter Value Selection



If this counter chosen by the attacker (Reject)

Last Accepted counter  
(Attacker doesn't know)

If this counter chosen by the attacker (Accept)

$\exists i (i \leq |LUT| \wedge LUT[i] = \text{count\_recv})$

$i_j = i, j = j + 1$

No

$j > 3$

yes

$(i_0 == i_0 + 1) \wedge (i_2 == i_1 + 1)$

If True: Accept

Receive Message

$txid_{prev} == txid_{recv}$

yes

$P[1]_{prev} == P[1]_{recv}$

yes

$\exists i (i \leq |LUT| \wedge LUT[i] = \text{count\_recv})$

$i_{prev} < i$

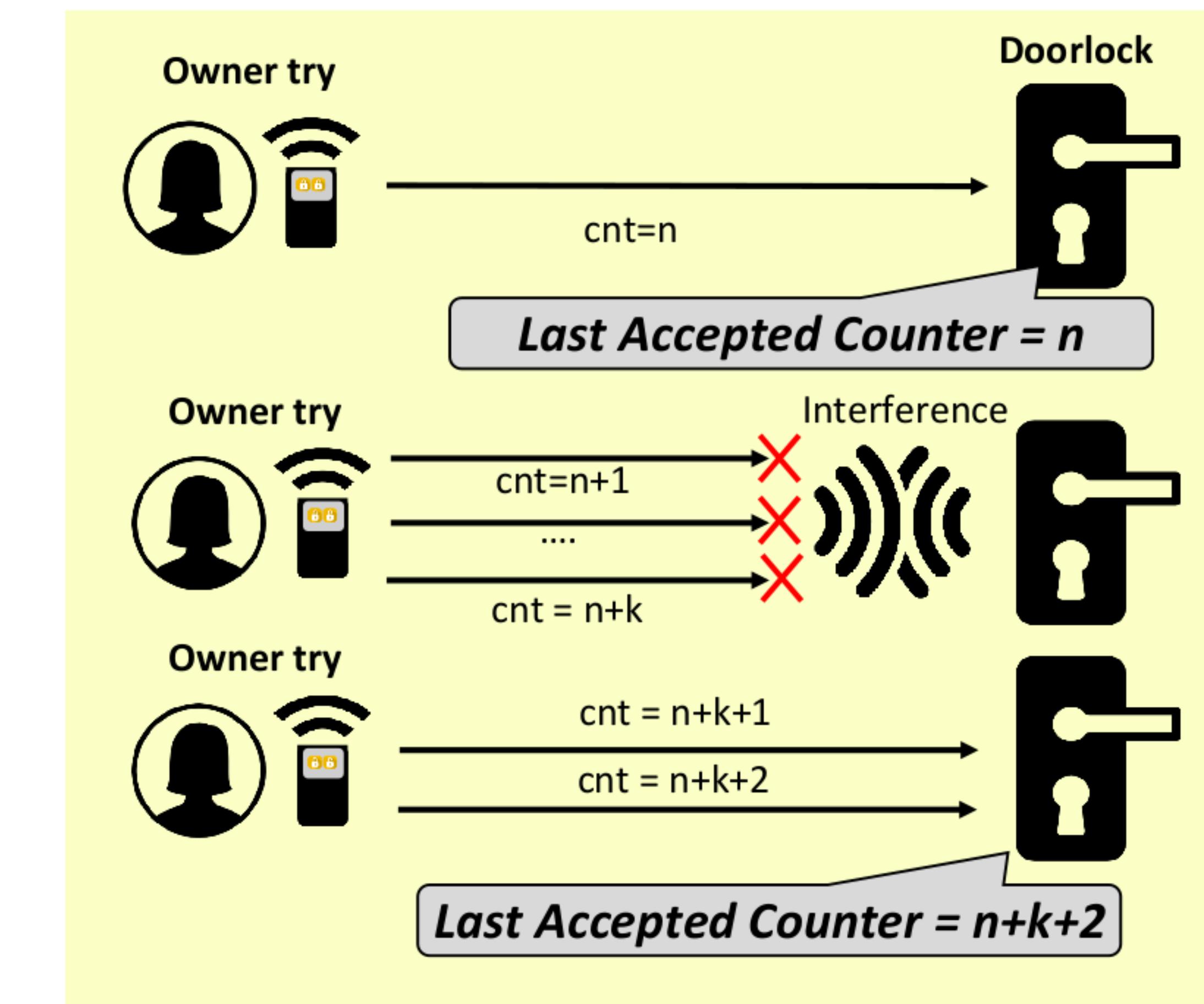
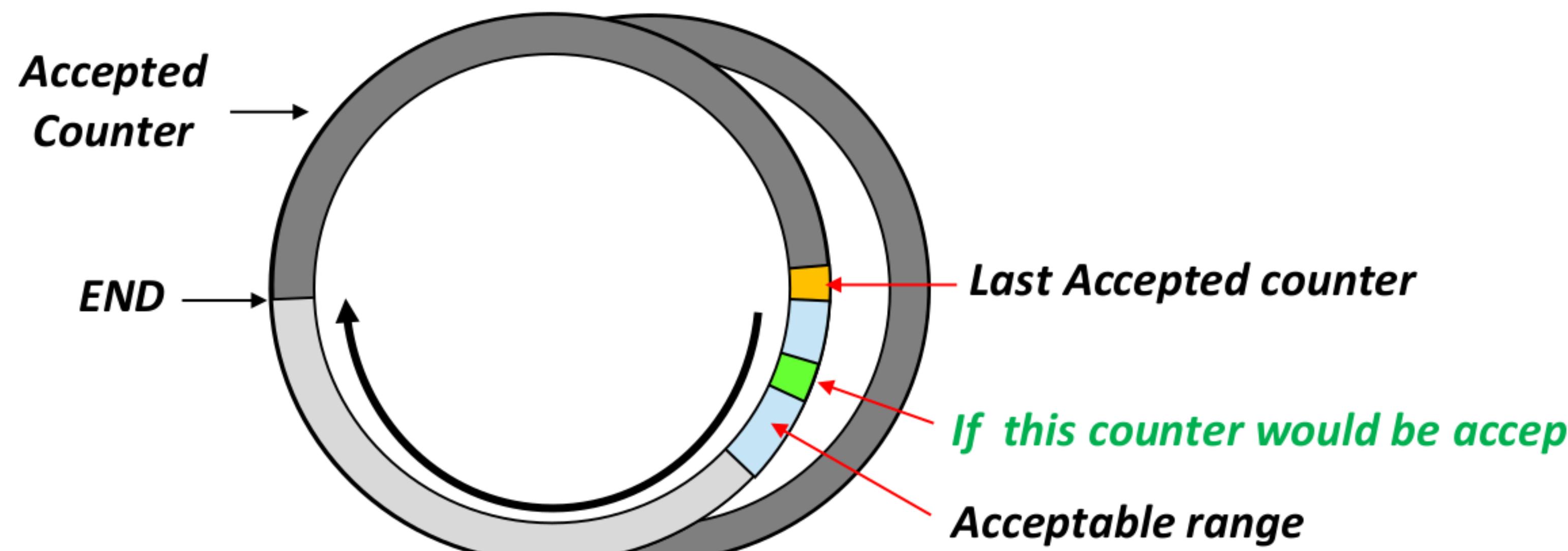
Accept

Reject

Reject

# Re-Synchronization and Acceptable Counter Range

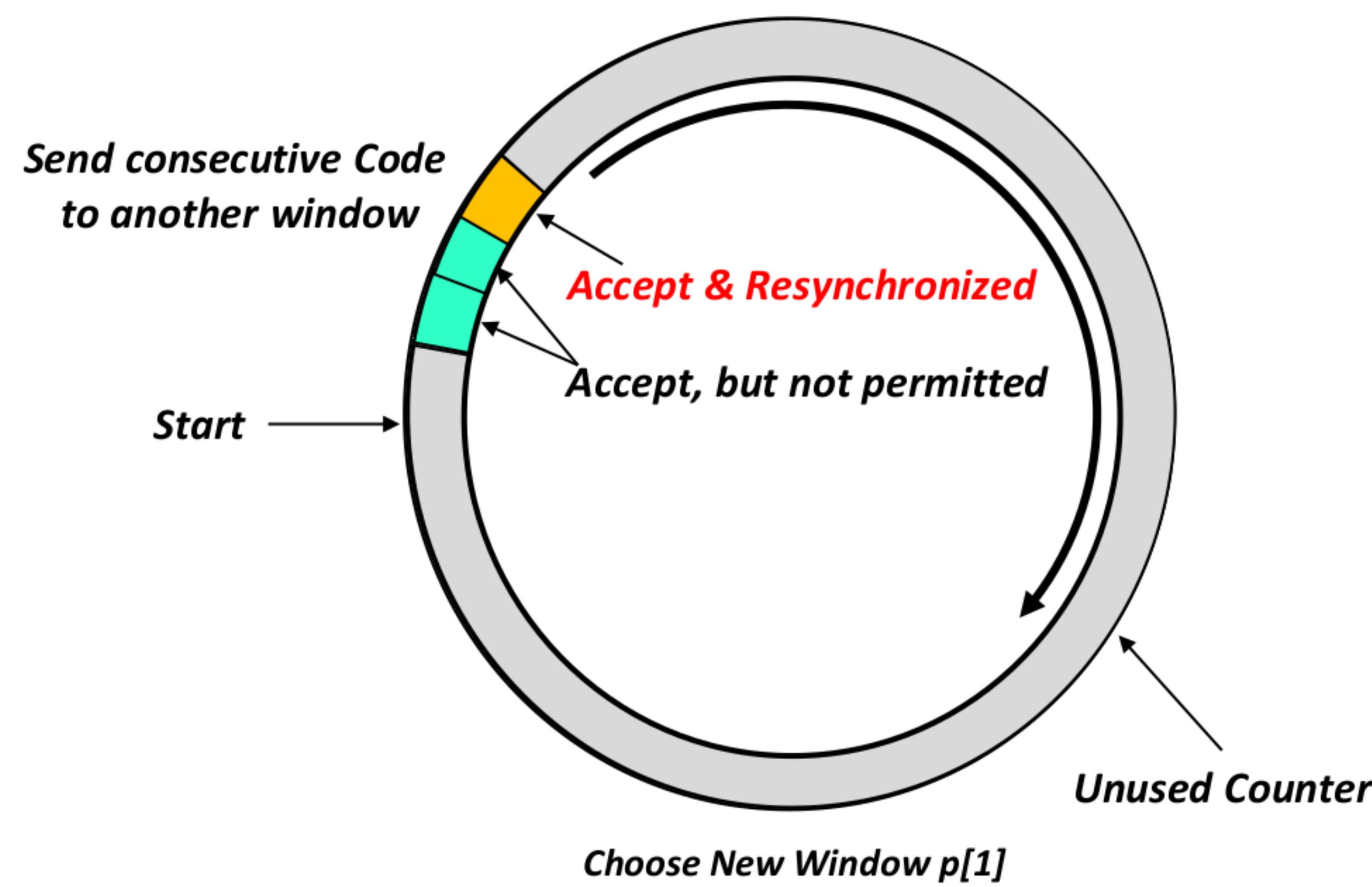
- ❑ There is always the possibility that the transmitter has been activated several times outside the receiver's range, the receiver must accept values
- ❑ To address this issue, many door locks have a synchronization function that allows them to accept in a specific range of counter values
  - ❑ Door locks will not accept a large counter value exceed to specific range



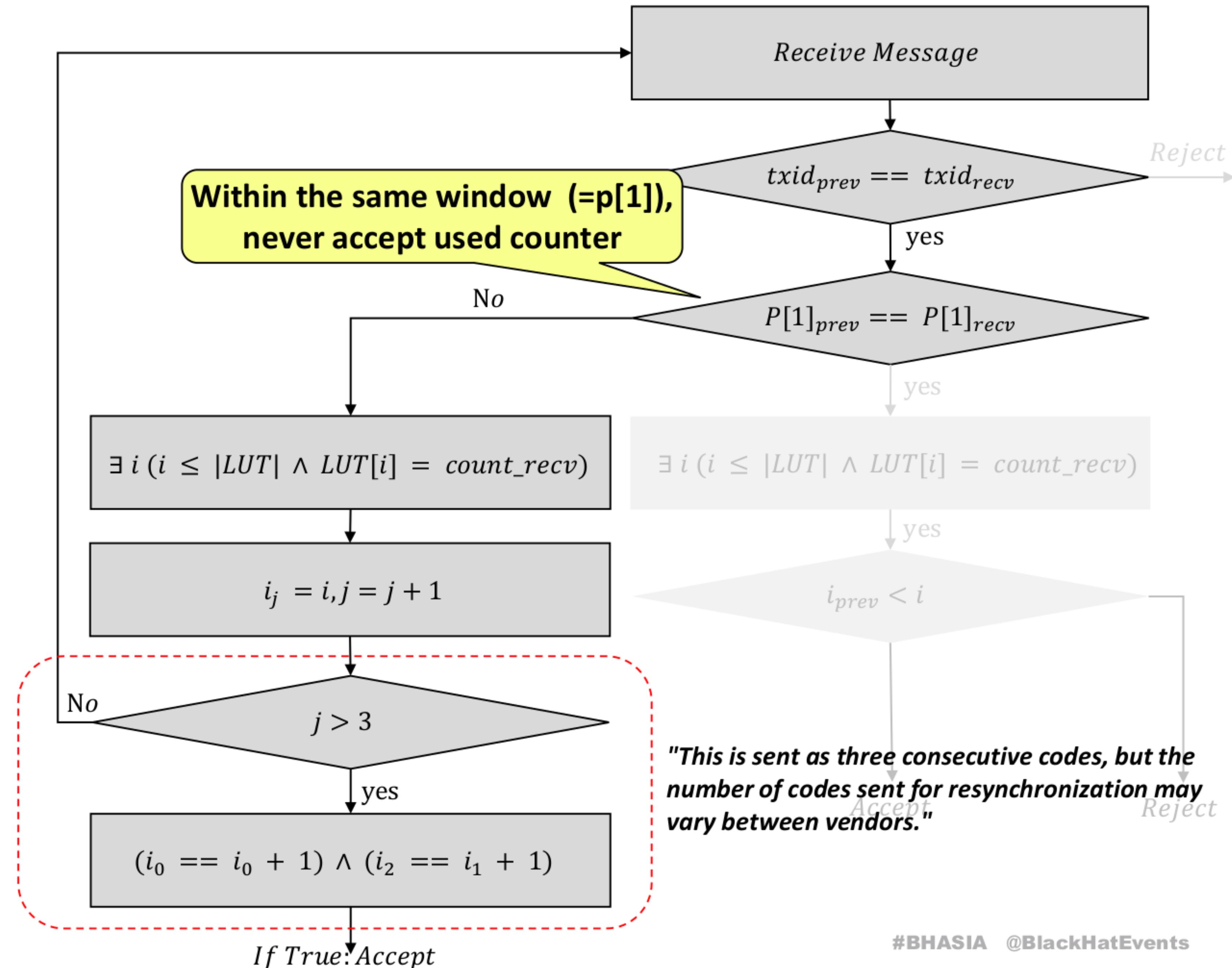
# Force Synchronization



## Resynchronization for Attack

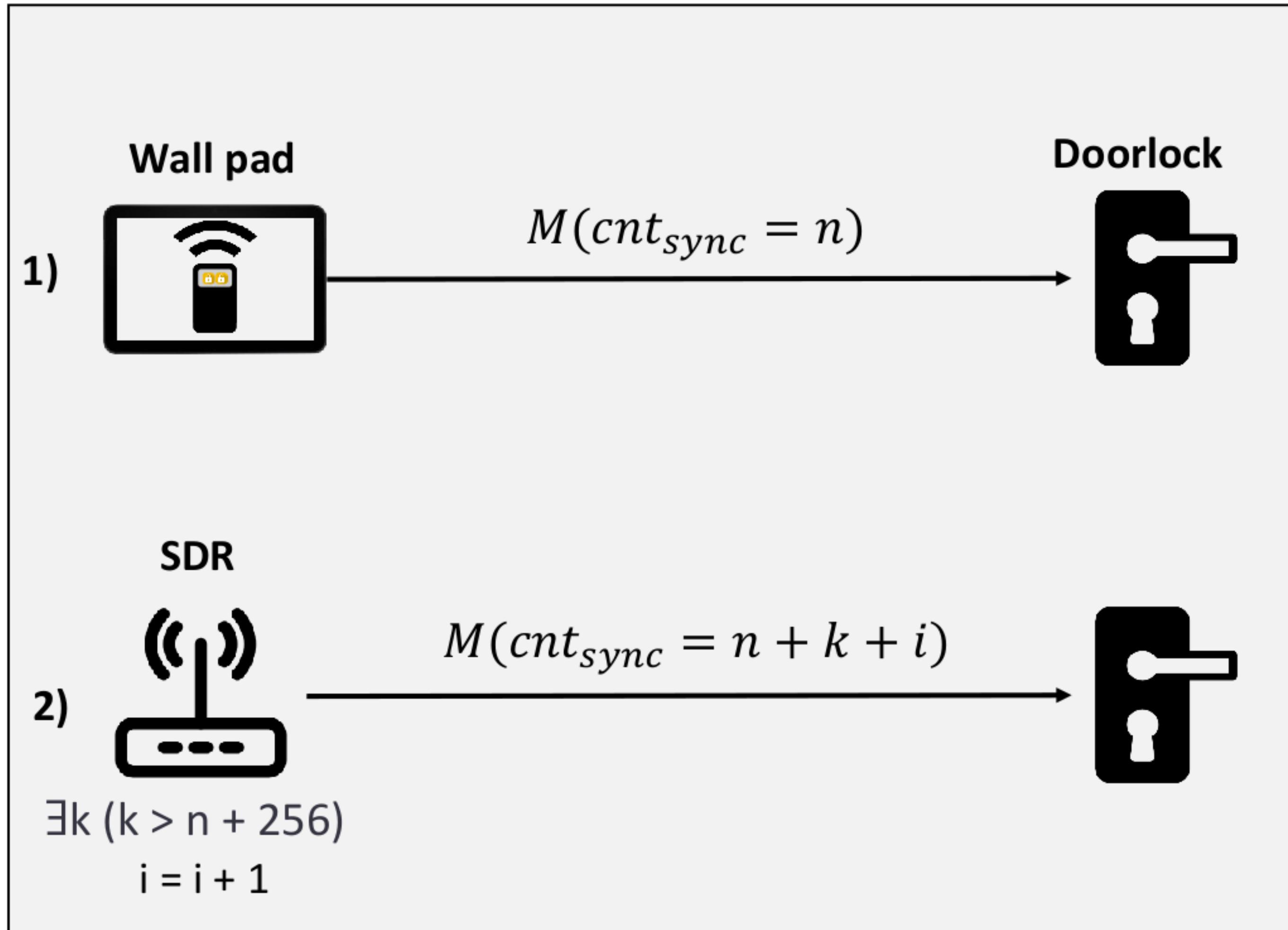


Minimizing the transmission packets necessary for resynchronization is crucial factor



- The number of consecutive code transmissions required for force synchronization
  - It is an important factor for increasing overall attack time

## *Test Method*



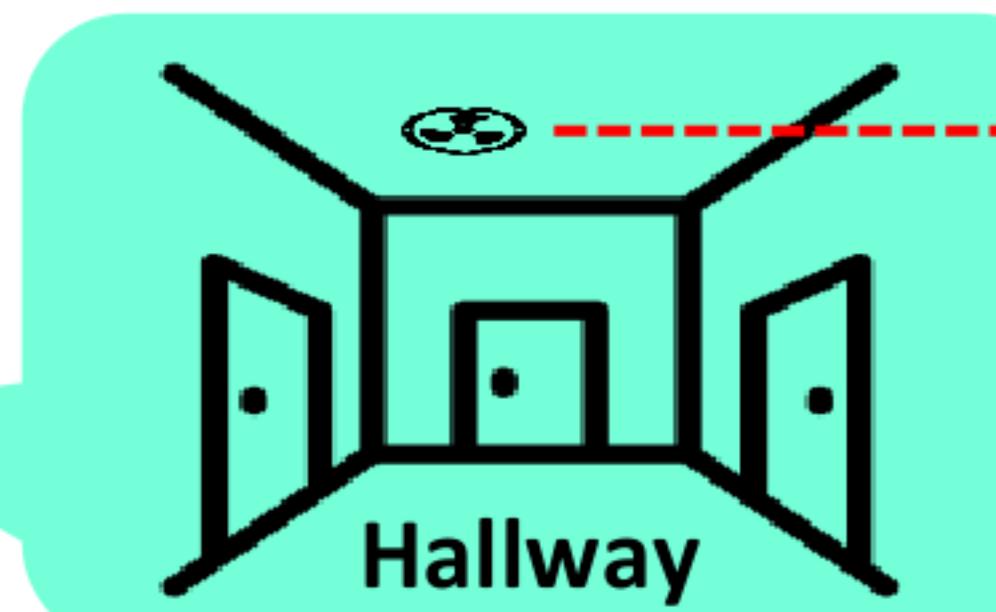
## *Code Counts for force resynchronization*

Vendors	Models	code count for force resync. max(i)
A	A-1	1 or 2
	B-1	3
	B-2	3
	B-3	3
B	B-4	3
	C-1	2
	C-2	2
	C-3	2
C	D-1	2
	D-2	2
I	I-1	1
J	J-1	1

# Lock Picking Attack Scenario

**Case : Vendor B** (*The products Send Three Consecutive Code for Re-synchronization*)

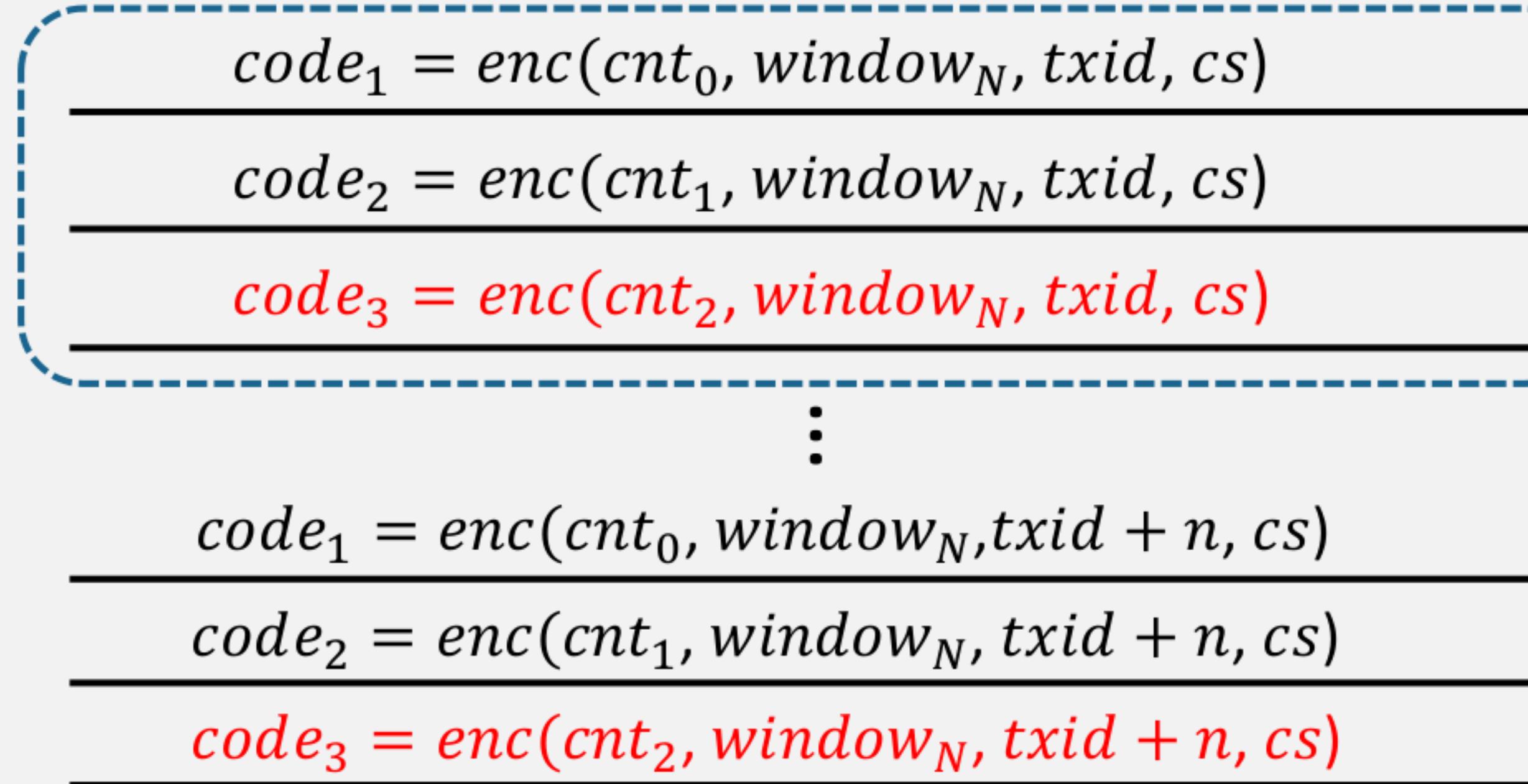
**Attack Scenario**



Suffixed at a hidden spot



*Message*



**Attacker**



**Vendor X**

Vendor B serial number Range is  
{0xBA, 0xA0, 0x00, 0x00}~{0xBA, 0xAF, 0x0F, 0xFF}  
Number of Codes: 2 Bytes =  $2^{16}$  = 65536

Total Tx time in the worst case  
=(Number of Codes × Packet count for Force Sync)  
/ Transmission Rate (MPS)  
= 65536 \* 3 / 5  
= approximately 39321 sec = 10.9 Hours



# Lock Picking w/o Sniffing Demo Video



## Discussion and Conclusion

---

# Takeaways



## ❑ Variant Replay Attack

- “**RollJam**” is inevitable without timestamps and “**RollBack**” is also feasible in door lock systems
- Our new variant attack called "**Loop Play Back**" has been confirmed as feasible in door lock systems
- At least in door lock system, the root cause of these attacks is confirmed

## ❑ Lock Picking Attack

- Easily exploitable by picking the lock w/ sniffing one signal
  - If signal archetype is known, it would be possible to recover the next code with one time
- It may still be vulnerable to open any door lock that's the same model w/o the use of sniffing
  - Depending on the properties of the TxID(=Serial#), it can be more practical to carry out this attack
  - Re-synchronization process is also key factor for brute force attack

## ❑ Easy-to-make and affordable tool

- We provided a diverse set of options to make tools using various methods (from SDR to DIY electronic parts)
- Detailed guide using affordable and easily accessible parts

# Lessons Learned



- RF security testing is essential
  - It is an important component of a comprehensive security strategy, especially for systems that rely on wireless
- Security through obscurity is not an answer
  - It's important to design systems that are resilient to attacks even if an attacker knows how they work
- The implementation should be based on elaborate principles and best practices
  - Mutually complementary and interdependent
- Assigning a unique key to each product is a better secure approach in one-way RF
  - If an attacker steals a key from one product, it will not affect communication with other products
  - Of course, using secure encryption algorithms is essential for security.



**Thank you!**

**If you have any question, please send me email**



**Kwonyoup Kim**  
**CEO/founder**  
**kkyoup@sntworks.kr**



**Seungjoon Lee**  
**Senior Researcher**  
**sj.lee@sntworks.kr**  
**shaftmom@gmail.com**