usenix THE ADVANCED COMPUTING SYSTEMS ASSOCIATION 2015

24th Security Symposium

GSMem

Data Exfiltration from Air-Gapped Computers over GSM Frequencies

Mordechai Guri, Assaf Kachlon, Ofer Hasson, Gabi Kedma, Yisroel Mirsky, Yuval Elovici

Ben-Gurion University of the Negev, Israel

Background

BRIDGING THE AIR GAP

Background

Air Gapped Networks

Definition: A cyber security measure that secures computer network by **physically isolating** it from unsecured networks, such as the public Internet or another unsecured local area network.

Examples of air gapped networks:

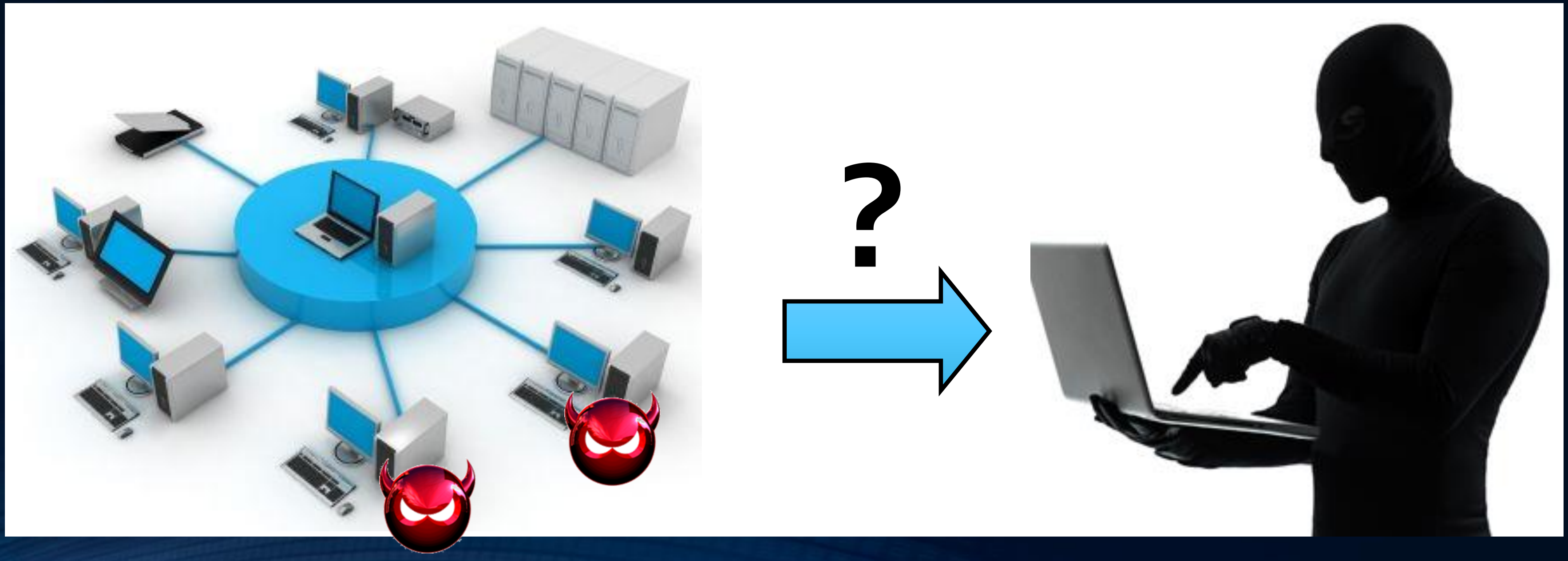
- Military defense system
- Critical infrastructure command and control centers
- Computerized medical equipment
- Finance
- And more…

Background

Air Gapped Networks

The Scenario:

- An attacker has succeeded in infecting the network
  - USB, insider, etc…
- The Attacker now wants to retrieve data from that network (over the air gap).



Background

Previous Work

Method	Transmitter	Receiver	Distance (m)	Rate (bit/s)
AirHopper [23] (78MHz -	Display cable	Cellular FM receiver	7	104-480
Ultrasonic [21] [24]	Speaker	Microphone	19.7	20
SAVAT [22] (~80KHz)	CPU/memory (laptops)	Dedicated equipment	1.0	N/A
BitWhisper [25]	Computer CPU/GPU	Computer Heat Sensors	0.4	8 bit/hour

Background

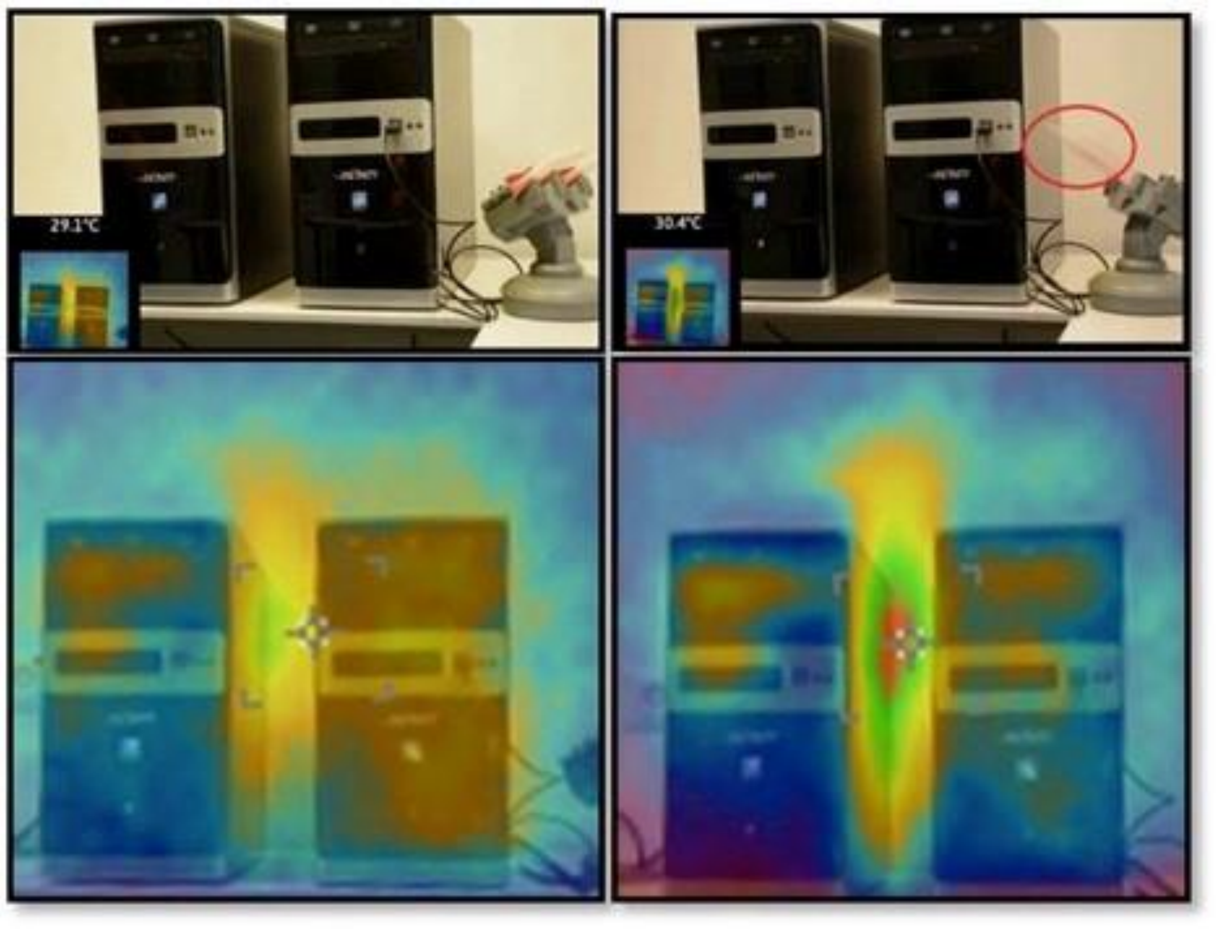
Previous Work

Method	Transmitter	Receiver	Distance (m)	Rate (bit/s)
AirHopper [23] (78MHz - 108MHz)	Display cable	Cellular FM receiver	7	104-480
Ultrasonic [21] [24]	Speaker	Microphone	19.7	20
SAVAT [22] (~80KHz)	CPU/memory (laptops)	Dedicated equipment	1.0	N/A
BitWhisper [25]	Computer CPU/GPU	Computer Heat Sensors	0.4	8 bit/hour

Background

Previous Work

Method	Transmitter	Receiver	Distance (m)	Rate (bit/s)
AirHopper [23] (78MHz - 108MHz)	Display cable	Cellular FM receiver	7	104-480
Ultrasonic [21] [24]	Speaker	Microphone	19.7	20
SAVAT [22] (~80KHz)	CPU/memory (laptops)	Dedicated equipment	1.0	N/A
BitWhisper [25]	Computer CPU/GPU	Computer Heat Sensors	0.4	8 bit/hou



Background
GSMem Overview

An ordinary desktop PC is converted into a small transmitting cellular antenna!

Transmitter

Receiver

Background

GSMem Overview

Demonstration Video

https://www.youtube.com/watch?v=RChj7Mg3rC4

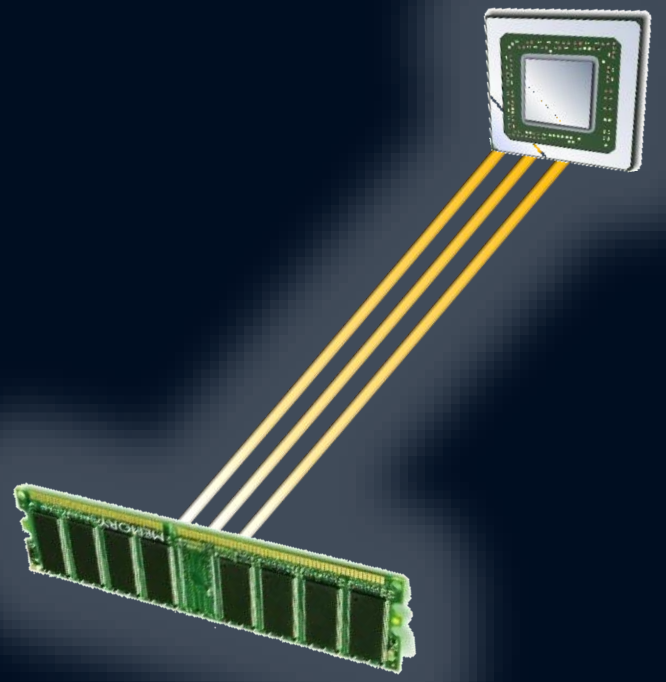
Transmitter

GSMEM

Transmitter

CPU-Memory BUS Emissions

How do we convert a computer's CPU-RAM configuration into a **radio antenna**?



How do antennas work?

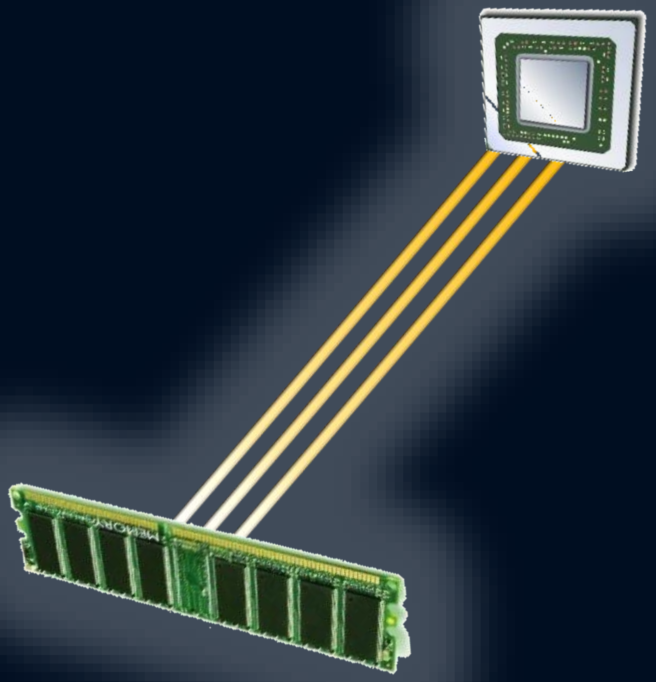
- Antennas emit radio waves (EMR) by oscillating current through their terminals
- Radio waves are characterized by their frequency (oscillation in Hz) and amplitude (strength in dBm).

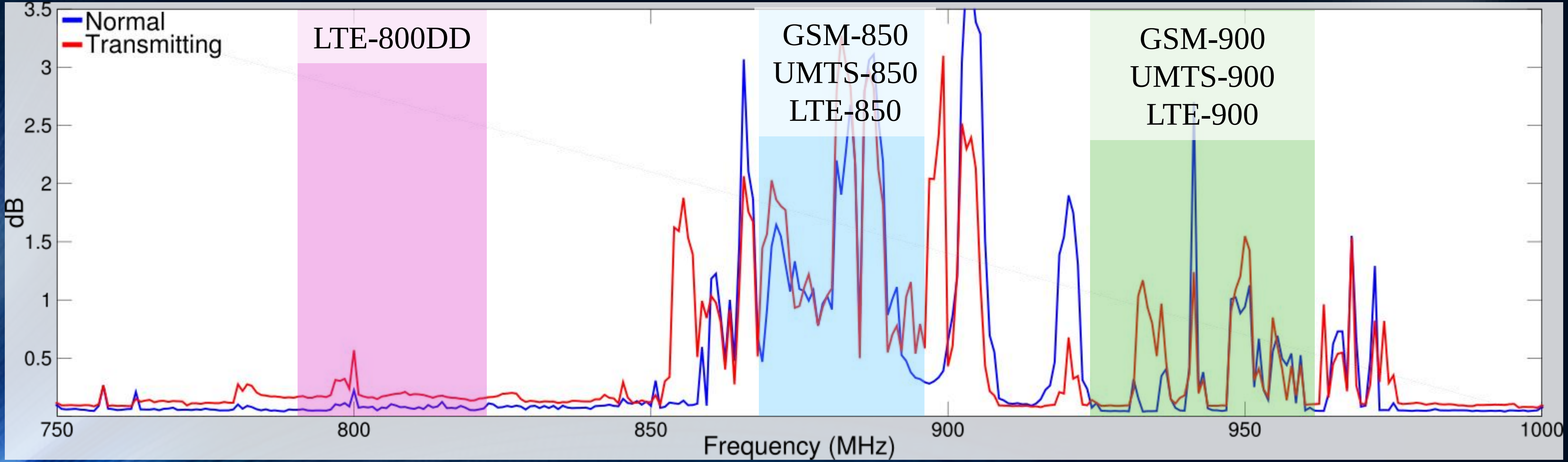
Transmitter

CPU-Memory BUS Emissions

How do we get this “antenna” to emit EMR on a cellular band (range of frequencies)?

- **Observation 1**: A **large** CPU-RAM transfer builds up oscillating current in the configuration.
- **Observation 2**: The BUS transfers bits at the FSB speed, emitting the energy around that frequency (e.g. 800 MHz)



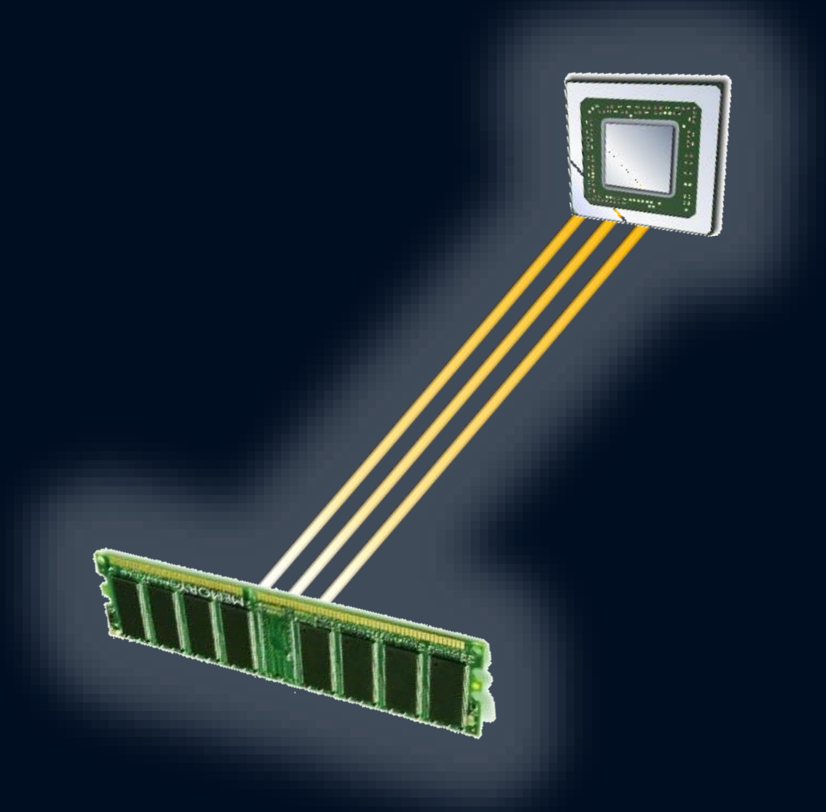


Transmitter

CPU-Memory BUS Emissions

Algorithm 1 transmit32 (data)

```
buffer ← ALIGNED_ALLOCATE(16,4096)
tx_time ← 500000
for bit_index ← 0 to 32 do
    if (data[bit_index] = 1) then
        start_time ← CURRENT_TIME()
        while (tx_time > CURRENT_TIME() - start_time) do
            buffer_ptr ← buffer
            for i ← 0 to buffer_size do
                SimdNtMov(buffer_ptr, 128bit_register)
                buffer_ptr ← buffer_ptr + 16
            end for
        end while
    else
        SLEEP(tx_time)
    end if
end for
```

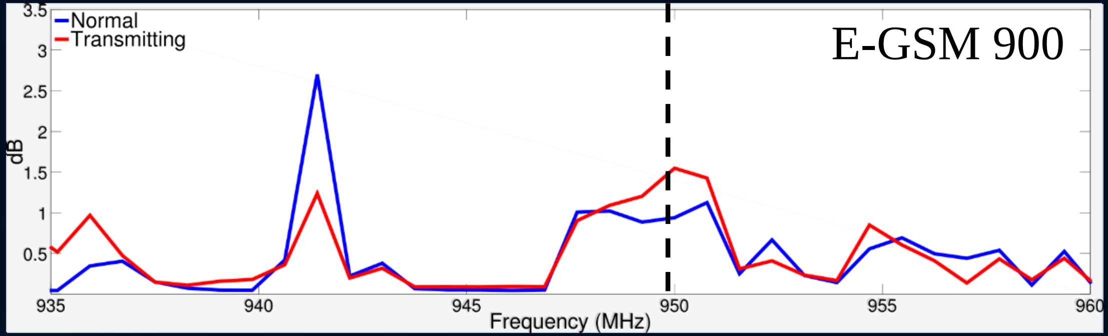


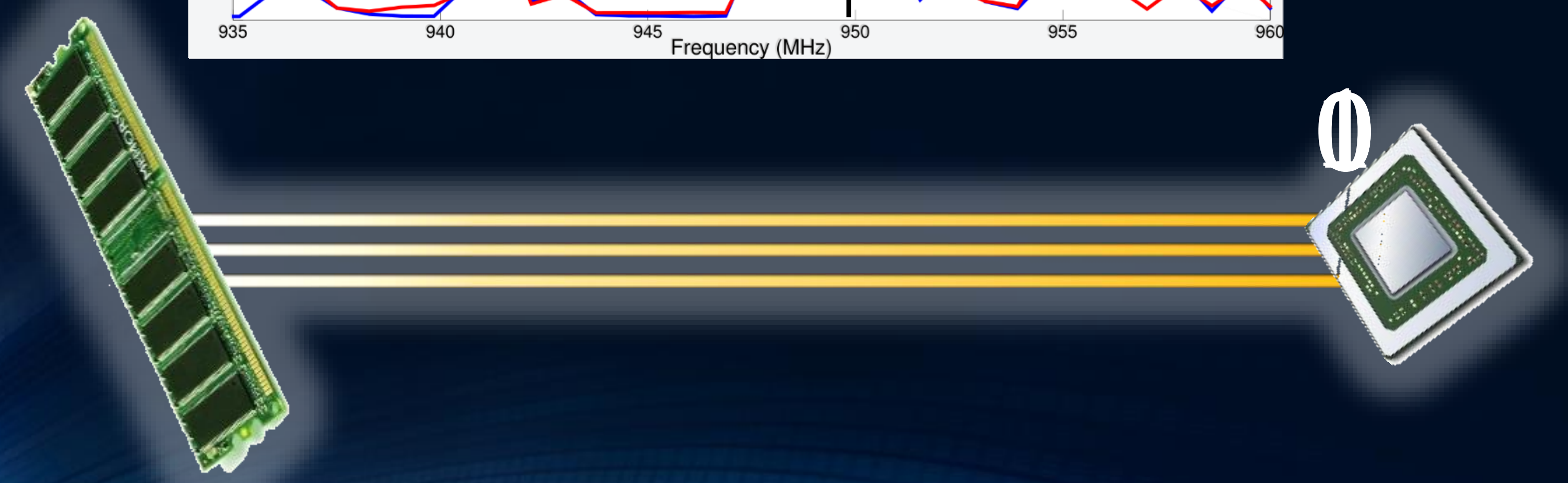
Transmitter

Sending a Bit (Modulation)

To send a bit, we use a variant of **B-ASK**:

Send(“0”): Do nothing for T seconds
Send(“1”): Raise amplitude for T seconds





Transmitter

Sending Lots of Bits (Framing)

To send a sequence of bits (some data payload) we perform framing.

This is for the benefit of the receiver to perform:

1. Transmission detection
2. Synchronization
3. B-ASK threshold selection (what amplitude is "0"?)
   - Dynamically updated (change in distance...)

Preamble	*Payload*	*Preamble*	*Payload*
1010	12 bits	1010	12 bits

Transmitter

Properties & Characteristics of the Transmitter

- Only has a 4KB memory footprint
- No root/admin required
- No APIs are used

- Affects Intel and AMD architectures…
- Works on Windows/Linux…

Receivers

GSMEM

Receiver

About Modifying Phones…

Baseband processor:

- The connection with the cellular network is managed by a dedicated chip, called the "baseband".
- Completely separated from the main OS (e.g., Android).
- Firmware of all common brands is **closed-source**

This will not deter highly motivated, and resourceful threats

…as we've seen in the past.

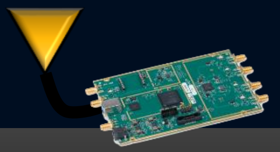
Receiver

About Modifying Phones…

Then how did we modify the firmware?

OsmocomBB: An open source GSM baseband software implementation (2010)

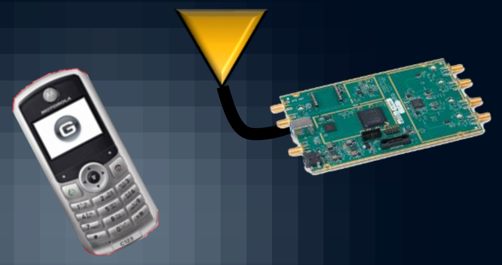
- For our experiments, used the **OsmocomBB** compatible Motorola C123 GSM phone.



We note that GSMem can even work on a nine-year old, low-end mobile phone ...modern technology can go even further.

Receivers

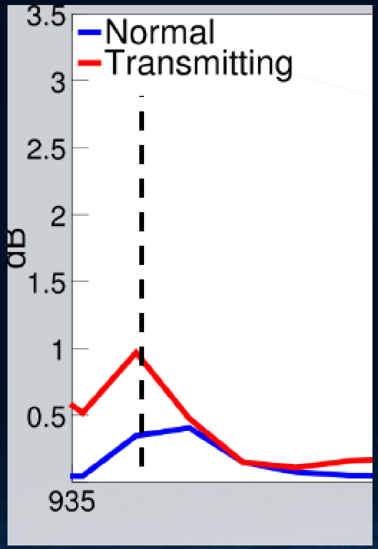
Getting the bits



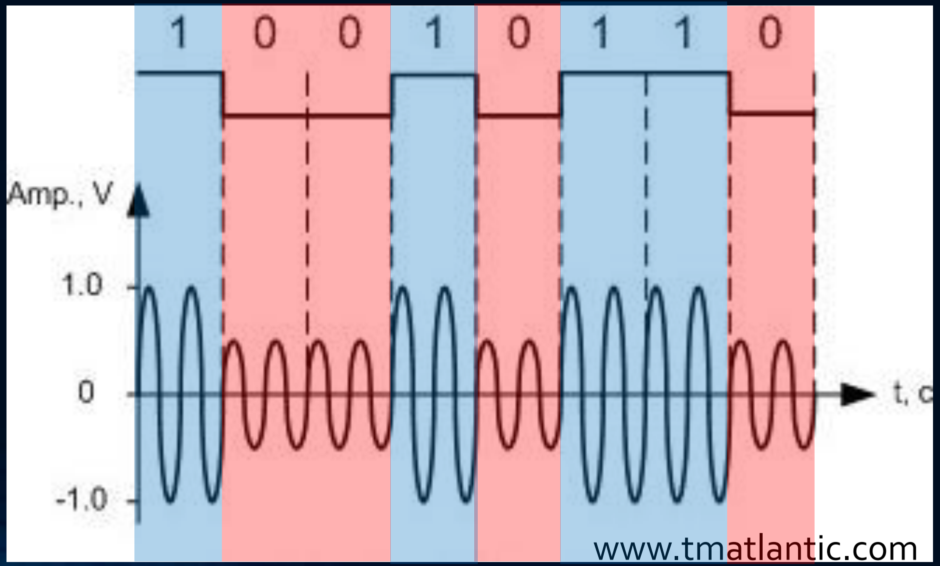
A Very Simplistic Approach:

1. Listen on "best" **frequency**
2. Search for the '1010' **preamble** (each bit T seconds long)
   - Threshold based (dynamically changed)
3. Extract 12 bit **payload** if **preamble** is found

Frequency Domain

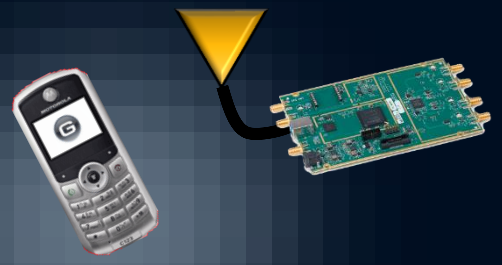


Time Domain



Receivers

Getting the bits



Algorithm 2 ReceiverHandler

```
dBm ← MEASURE(f_c)
filtered_signal ← UPDATEMOVINGAVERAGE(dBm)
if (state = SCAN) then
    f_c ← SCANFREQ()
    SETSTATE(PREAMBLE)
end if
if (state = PREAMBLE) then
    if (IDENTIFYPREAMBLE(filtered_signal) = true) then
        SETSTATE(RECEIVE)
    end if
end if
if (state = RECEIVE) then
    b ← DEMODULATEBIT(filtered_signal)
    bitSequence.add(b)
    if (bitSequence.size%16 = 0 or SIGNALLOST(filtered_signal)) then
        SETSTATE(PREAMBLE)
    end if
end if
```

Evaluation

GSMEM

Evaluation

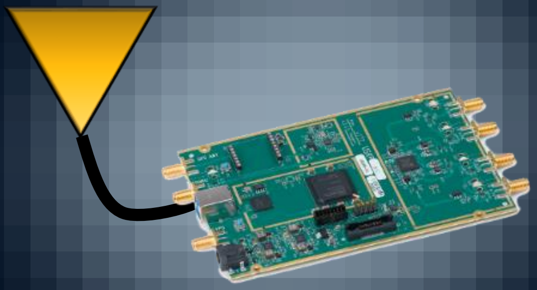
Experiment Setup

Transmitters

	WS1	WS2	WS3
OS	Linux Fedora 20		
Chassis (metal)	infinity chassis	GIGABYTE Setto 1020 GZ-AX2CBS	Silverstone RL04B
CPU	Intel i7-4790	Intel i7-3770	Intel i7-5820K
Motherboard	GIGABYTE GA-h87M-D3H	GIGABYTE H77-D3H	GIGABYTE GA-X99-UD4
RAM Type	2 x 4GB 1600MHz		4 x 4GB 2133MHz
RAM Frequencies Tested	1333/1600 MHz		1833/2133 MHz
RAM Operation Modes Tested	Single / Dual		Dual / Quad

Receivers

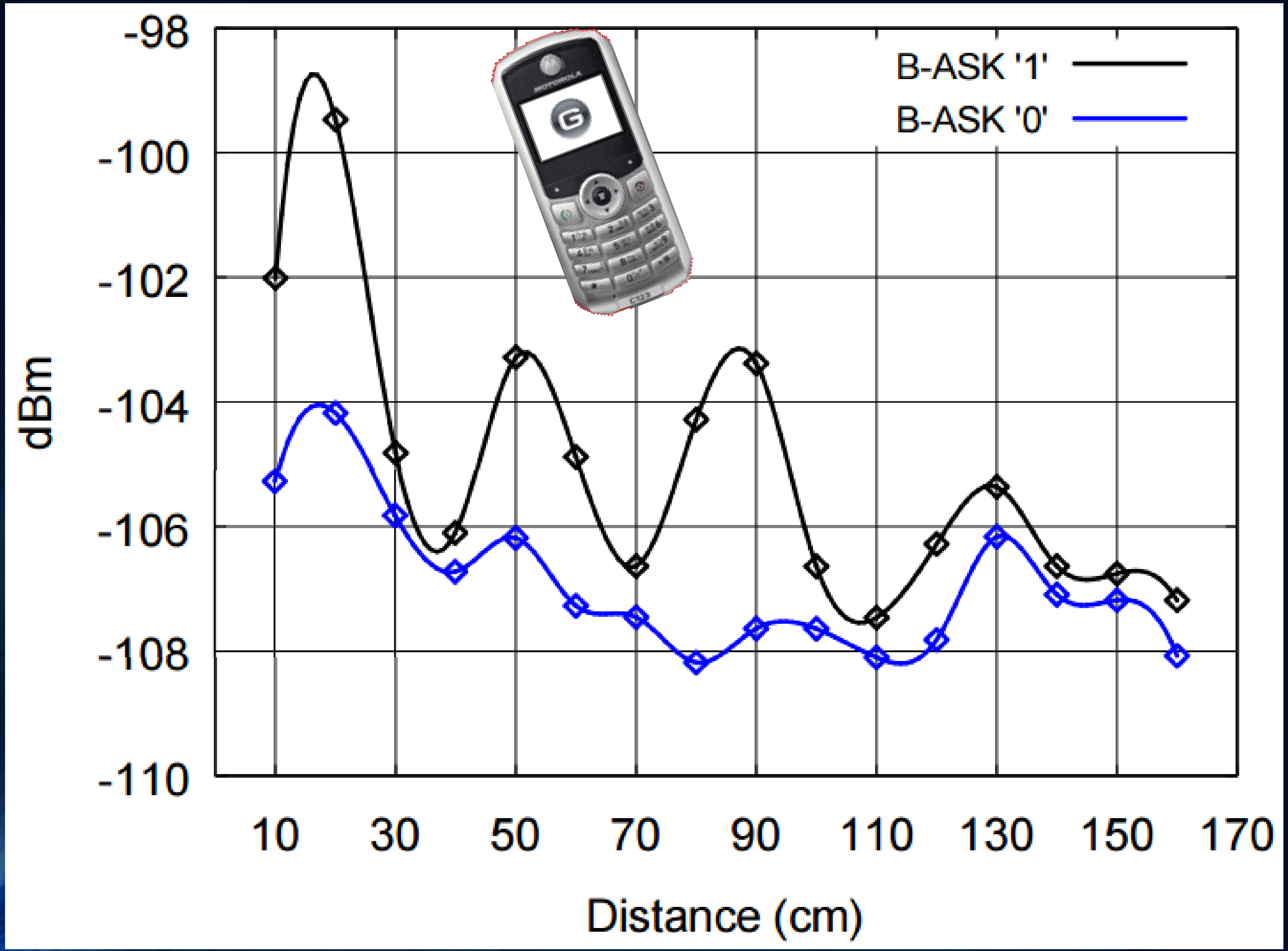
USRP B210



Motorola C123

Evaluation
Reception Distance

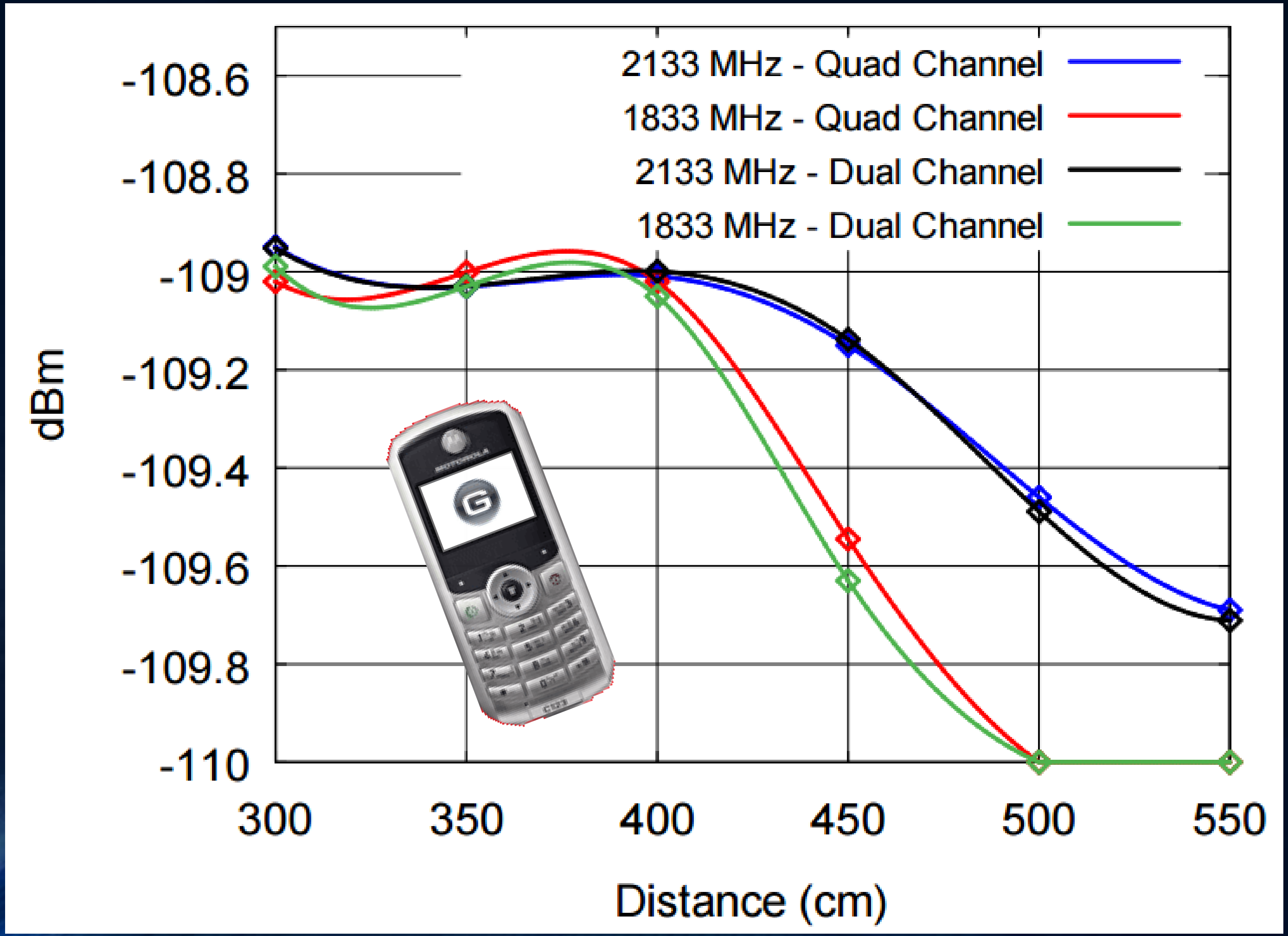
Amplitude ‘0’ vs ‘1’



XKCD

Evaluation
Reception Distance

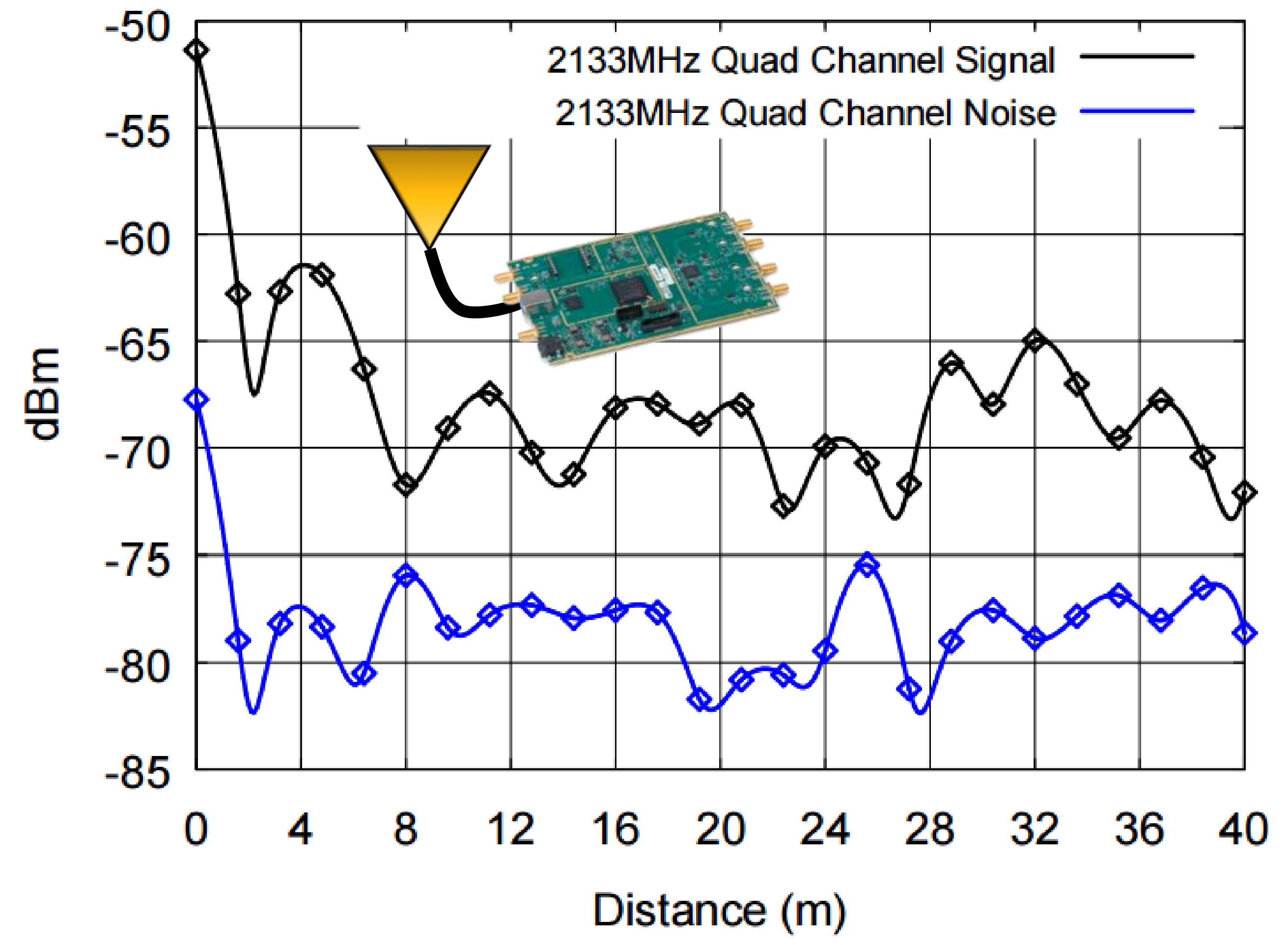
Delta between ‘0’ & ‘1’



XKCD

Evaluation
Reception Distance

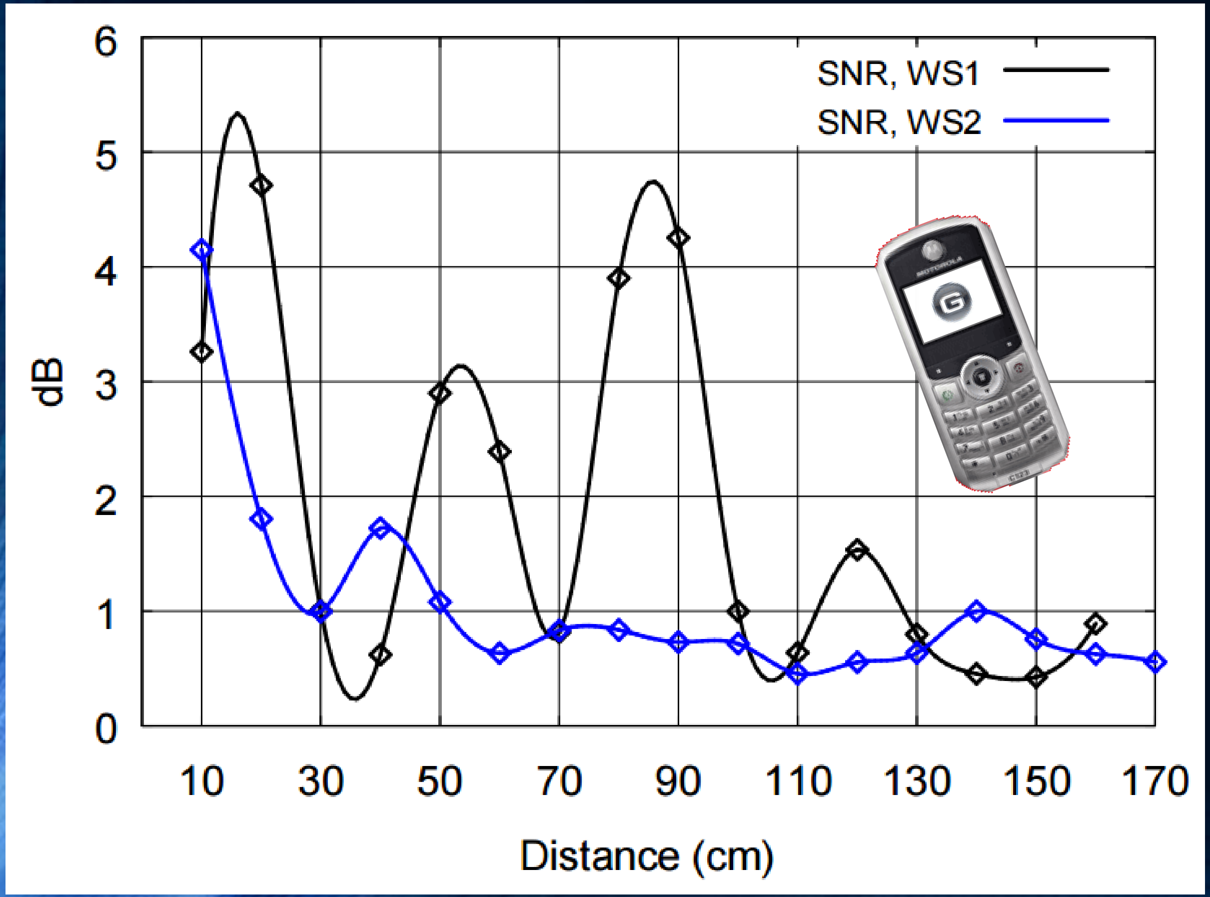
Amplitude ‘0’ vs ‘1’



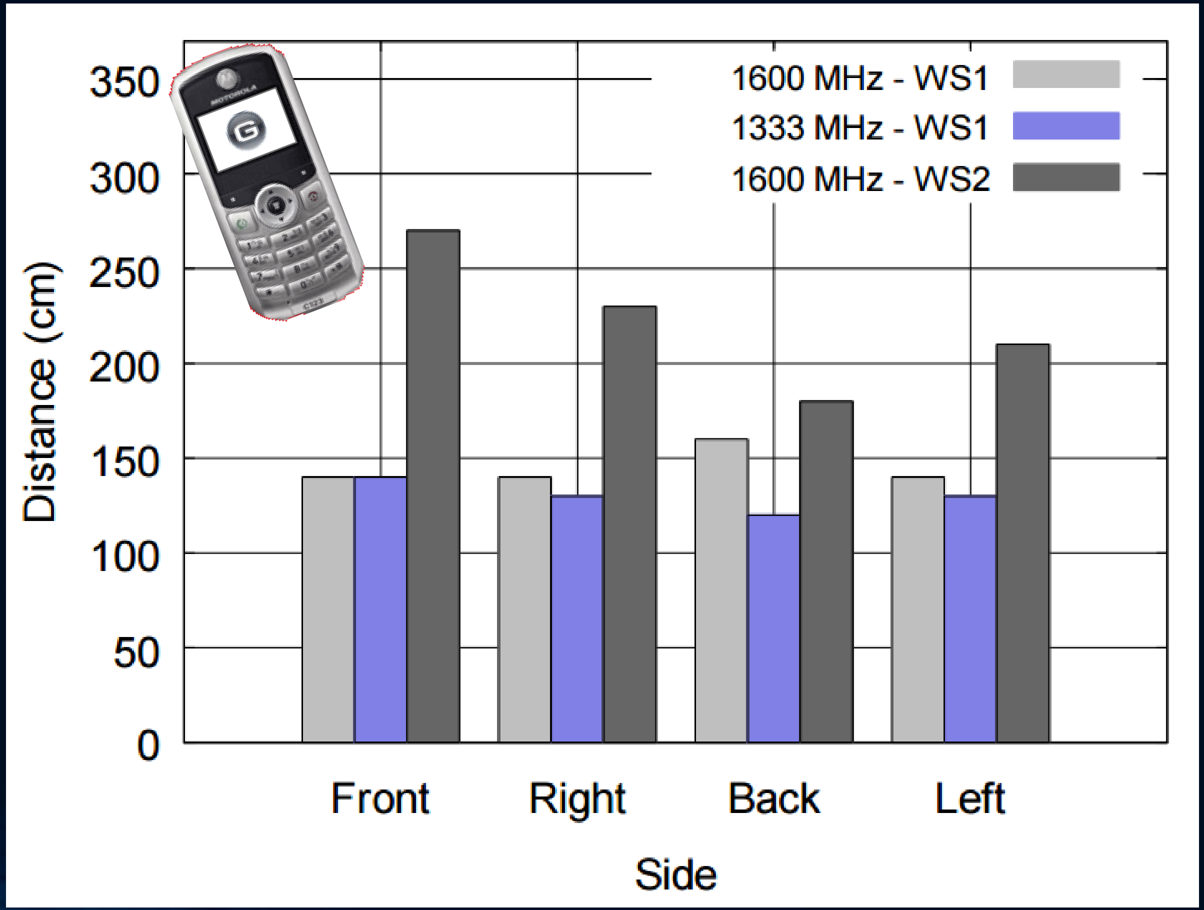
XKCD

Evaluation
Signal to Noise Ratio (SNR)

SNR from the back of WS1 &WS2

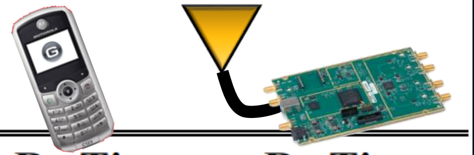


Distance at which SNR = 0.5dB



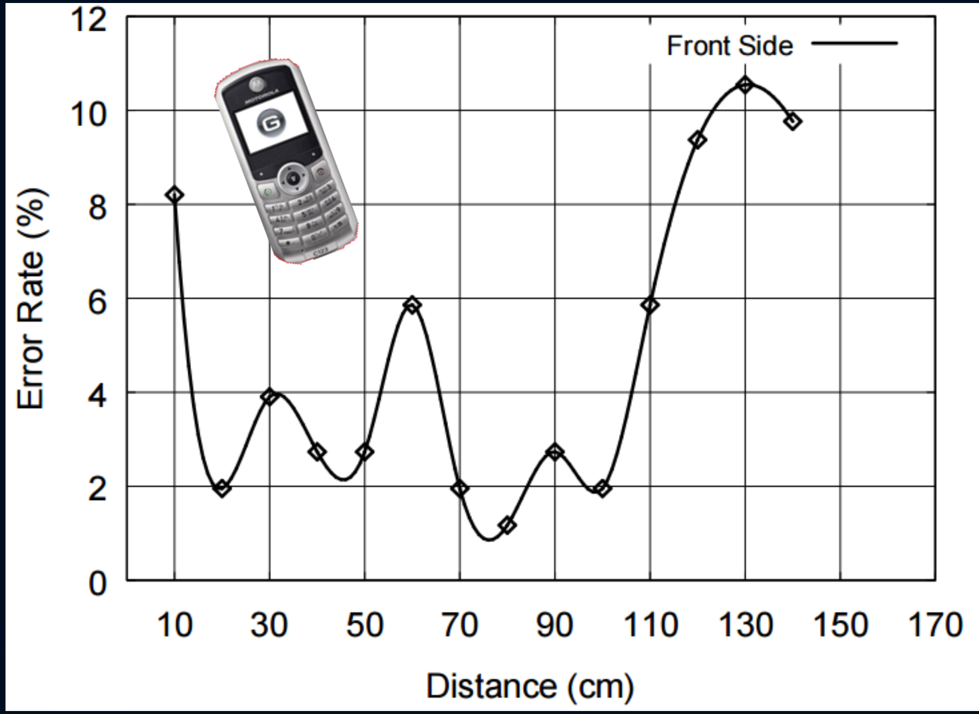
Evaluation

Bit Rates



Data	Length (bit)	Rx Time	Rx Time
MAC Address	48	30 sec	48 ms
Plain Password	64	40 sec	64 ms
MD5	128	1.3 sec	128 ms
GPS Coordinate	128	1.3 sec	128 ms
SHA1 Hash	160	1.6 min	160 ms
Disk Encryption Key	256	2.6 min	256 ms
RSA Private Key	2048	21.3 min	2.04 sec
Fingerprint Template	2800	29.1 min	2.8 sec

Bit Error Rate (BER)



Filters, FEC and other well known methodologies can improve the BER further!

Conclusion

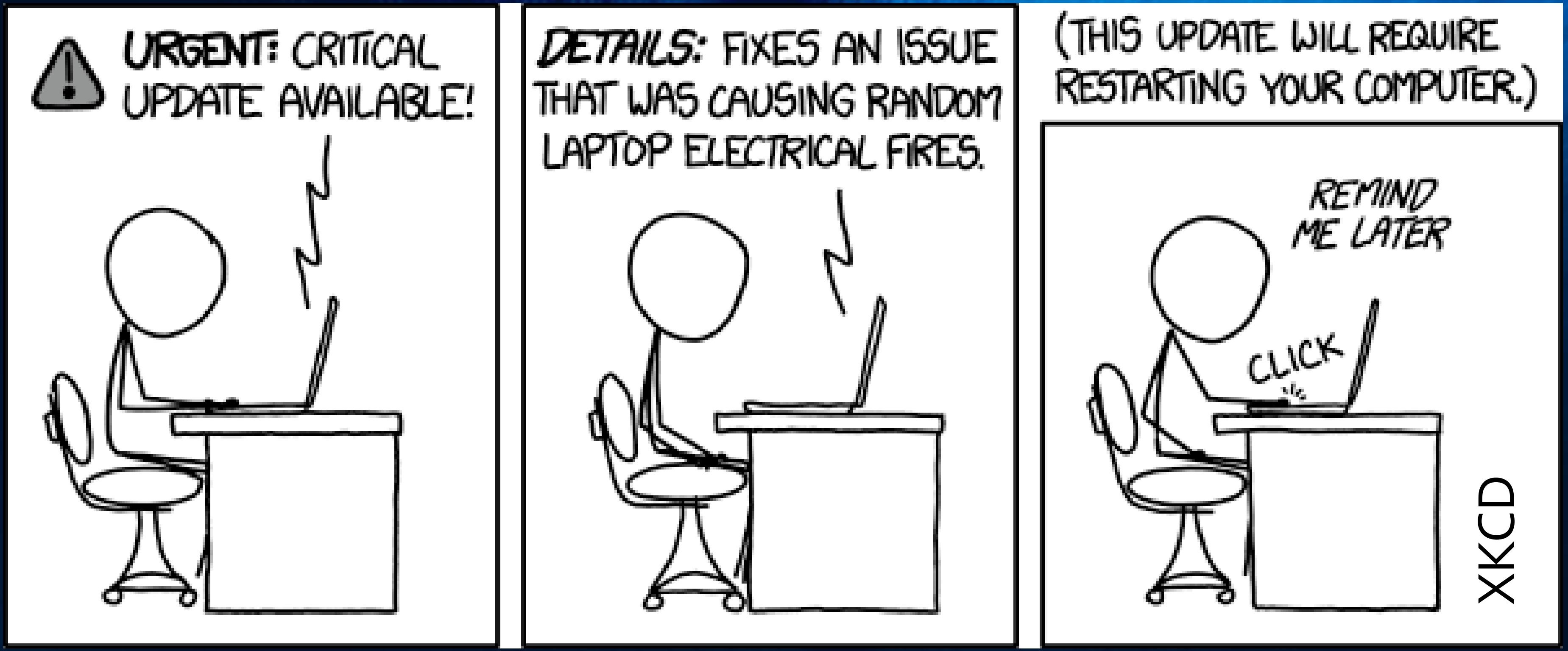
Summary

- It’s feasible to get data out of an “Air-Gapped” network
- EMR from memory-bus can be exploited to transmit information
- Mobile devices can receive this information

Note:

- Some corporations allow simple GSM phones into restricted areas…
- Issue applies to: GSM, LTE,… bands
- GSMem is relevant to other scenarios as well

Thank you for listening!



Questions?