Walnut

Hacking MEMS Accelerometers with Acoustic Injection Attacks

Outline

- Motivation
- Primer
- Signal Model
- Hardware Vulnerabilities
  - Filter
  - Amp
- Attacks
  - Output biasing
  - Output control
- Defends
  - Hardware
  - Software

Motivation

- Increasing number of motion-driven applications using MEMS such as medical implants, automobiles, avionics.
- Lots of efforts in software security, but not so much in hardware aspects.
- DoS attack to MEMS sensor is possible, but
  - Finer-grained control?
  - Will software be tricked?
  - How to protect?

System & model?

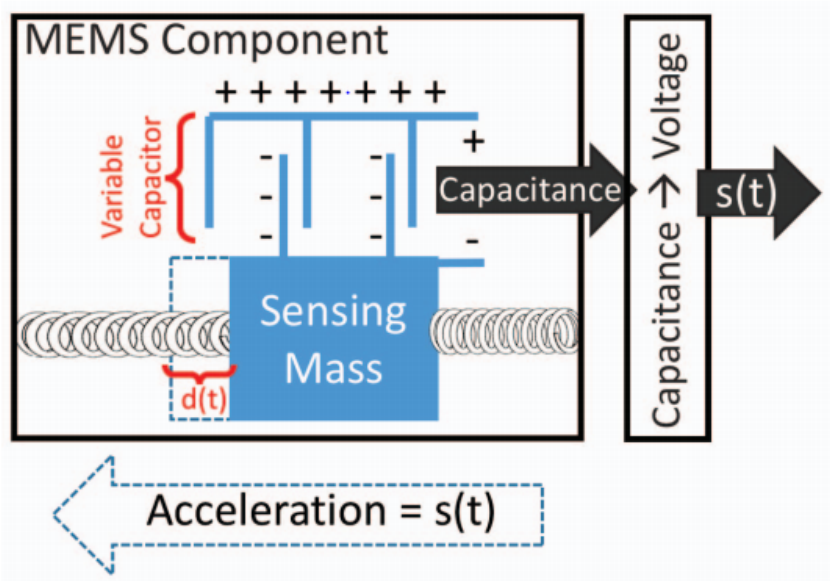
Primer (Accelerometer)

Newton's Law: $F = m \cdot a$

Hooke's Law: $F = -k_s \cdot d$

@ sensing mass: $a = \frac{-k_s d}{m}$

Acceleration $a(t)$ results in a displacement $d(t)$ which induces a time-varying capacitance and is converted into signal $s(t)$.

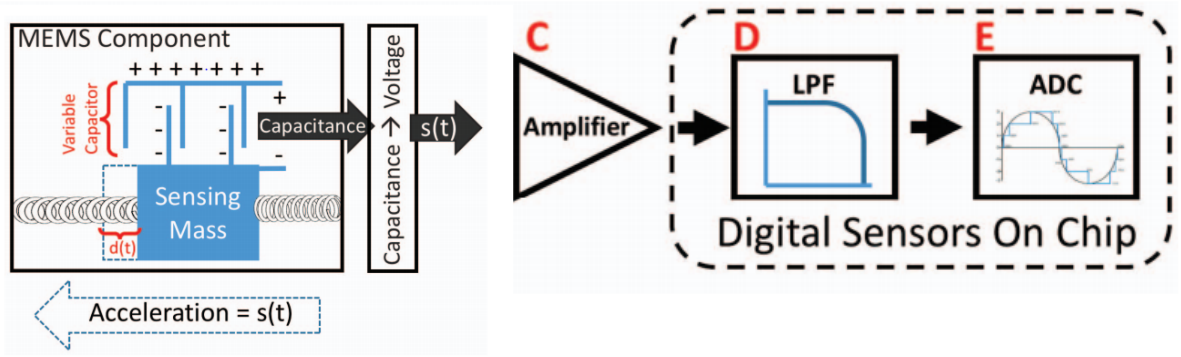


Primer (MEMS Accelerometer System)

Analog signal is sent into amplifier (C) and anti-aliasing LPF (D), and then sampled by ADC (E).

By Nyquist, $F_{\text{cutoff}} = \frac{1}{2}F_s$. But in reality there's transition band, some frequencies > F_{cutoff} get through.

Non-ideal amplifier has a dynamic range, above which signal clipping occurs.



Signal Model

- Key: Acoustic wave can move the sensing mass.
- Measured signal = (true signal) + (attenuated acoustic signal)

$\hat{s}(t)$ $s(t)$ $s_a(t)$

$$\hat{s}(t) = s(t) + A_1 \cdot s_a(t)$$

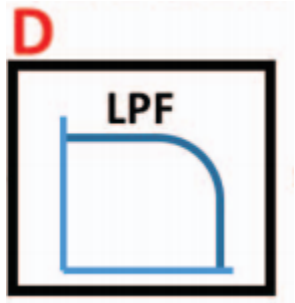
- For example, send sinusoid $s_a(t) = A_0 \cdot cos(2\pi F_a t + \phi)$. Measured signal is

$$\hat{s}(t) = s(t) + A_1 A_0 \cdot cos(2\pi F_a t + \phi) \quad (2)$$

- This is the signal model used throughout the paper.
- Actually, $A_1 = A_1(F)$ maximized at $F = F_{res}$, where measured signal significantly deviates from true signal.

Where to attack?

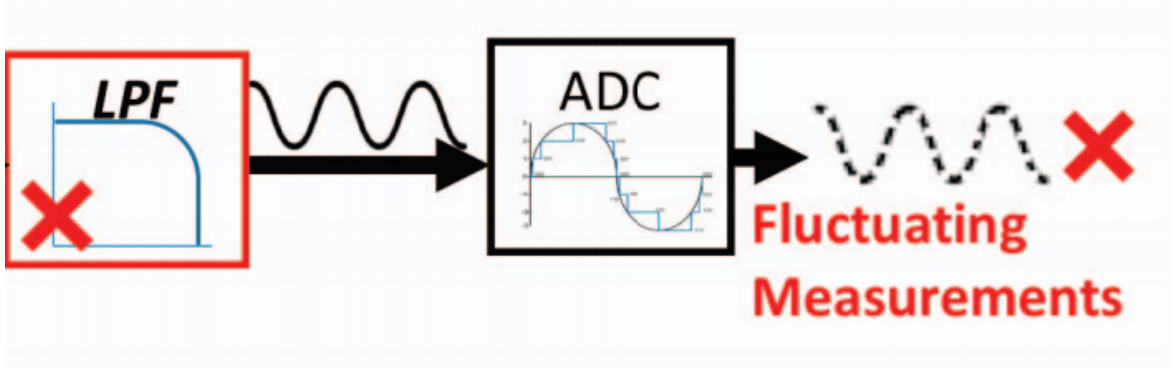
Hardware Vulnerabilities (LPF)



Recall that LPF are non-ideal (transition band).

Hence, by (2), the sinusoid doesn't get filtered out and will manifest itself as a sinusoid fluctuation in the **false** acceleration measurement.

A **secure LPF** has the acoustic frequency in its stop band. Additionally, resonant frequency should be in stop band.



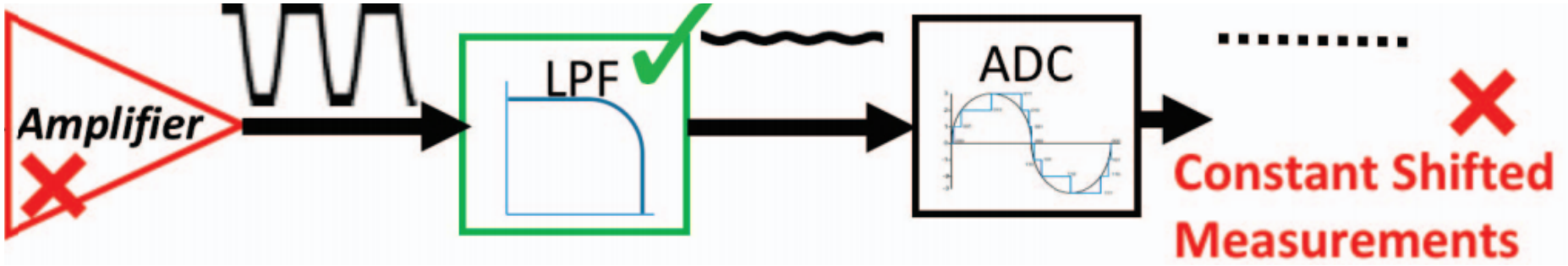
Hardware Vulnerabilities (Amplifier)

Recall that non-ideal amplifier has a dynamic range.

Send acoustic wave at resonant frequency → large displacement → exceed dynamic range → **clipping**

Clipping introduces a DC signal that passes through LPF, which is shown as a **bias** in the **false** acceleration measurement.

A **secure amplifier** has input to it within its dynamic range.



How to attack?

Define the attack

- There are types of attacks: DoS, etc…
- In this paper, attack is **using acoustic wave to generate desired sensor output signal.**
- Two ways:
  - Output biasing attack
  - Output control attack

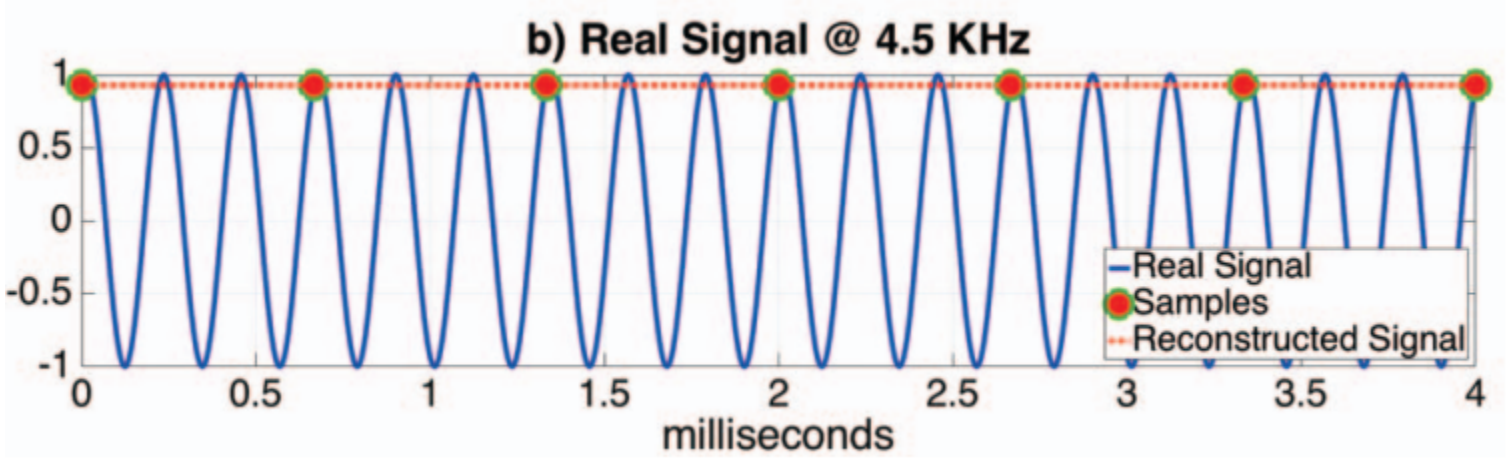
Output Biasing Attack

- Possible due to ADC sampling deficiencies and insecure LPF.
- 2 steps:
  - Generate DC alias
  - Modulate signal on resonant frequency

Output Biasing Attack

- DC alias when analog signal's frequency is an integer multiple of sampling frequency F_{samp}
- The ADC samples at time $t_k = k/F_{samp}$

Type equation here.



Output Biasing Attack

- We want:
  - Send acoustic wave at resonant frequency F_{res}
  - Also integer multiple of F_{samp}
- But this is rarely the case.
- However, resonance is a “zone”, so we can transmit at $F_a = F_{res} + f_\epsilon$ where $F_a = \mathrm{n}F_{samp}$

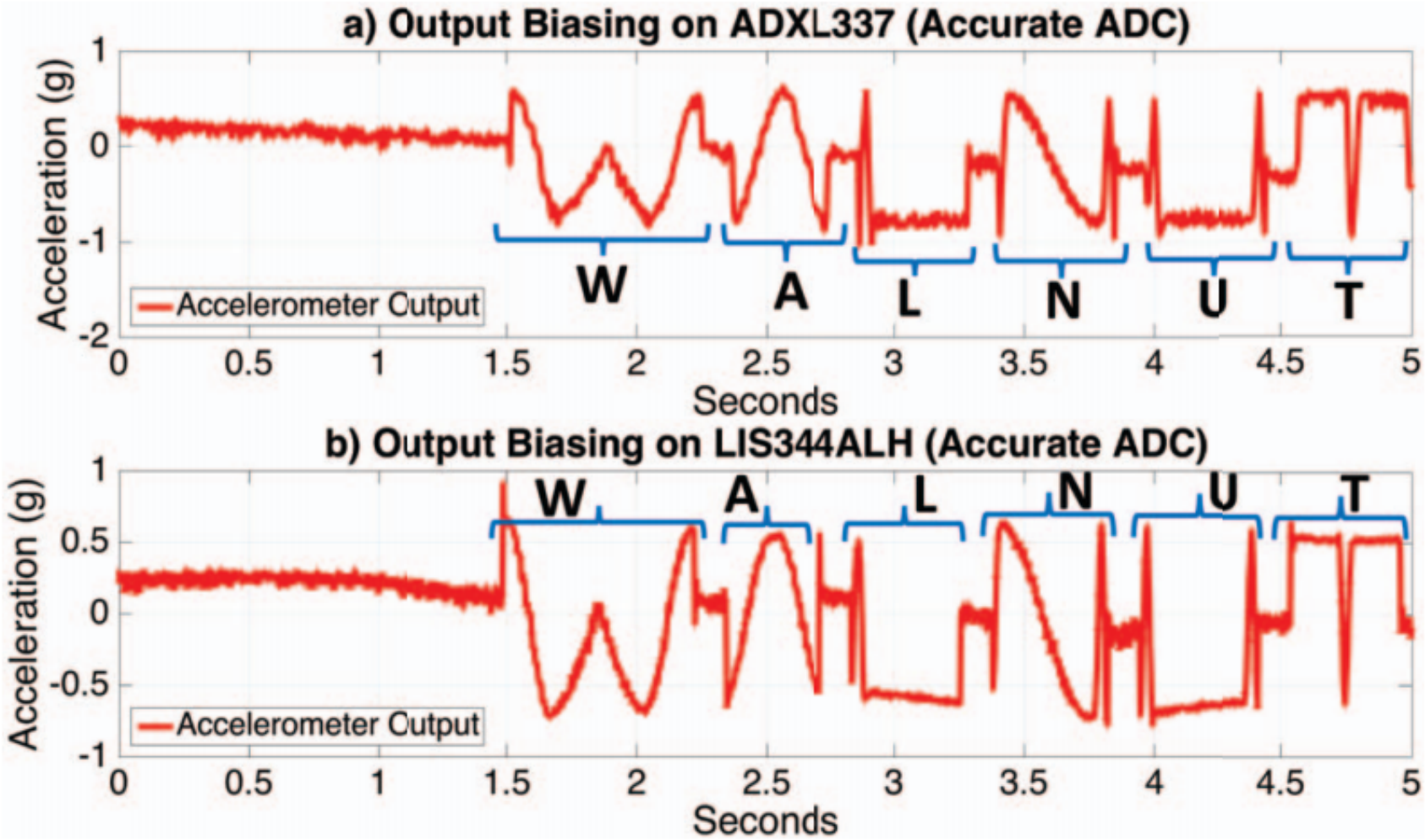
Output Biasing Attack

- Returning to (2), the sampled signal is

$$\begin{aligned}\hat{s}(t_k) =& s(t_k) + A_1 \cdot s_a(t_k) \\ =& s(t_k) + A_1 A_0 \cdot cos(2\pi F_a t_k + \phi) \\ =& s(t_k) + A_1 A_0 \cdot cos(2\pi N k + \phi) \\ =& s(t_k) + A_1 A_0 \cdot cos(\phi)\end{aligned} \tag{3}$$

- Hence we can send information via $A(t)$ or $\phi(t)$.

Output Biasing Attack

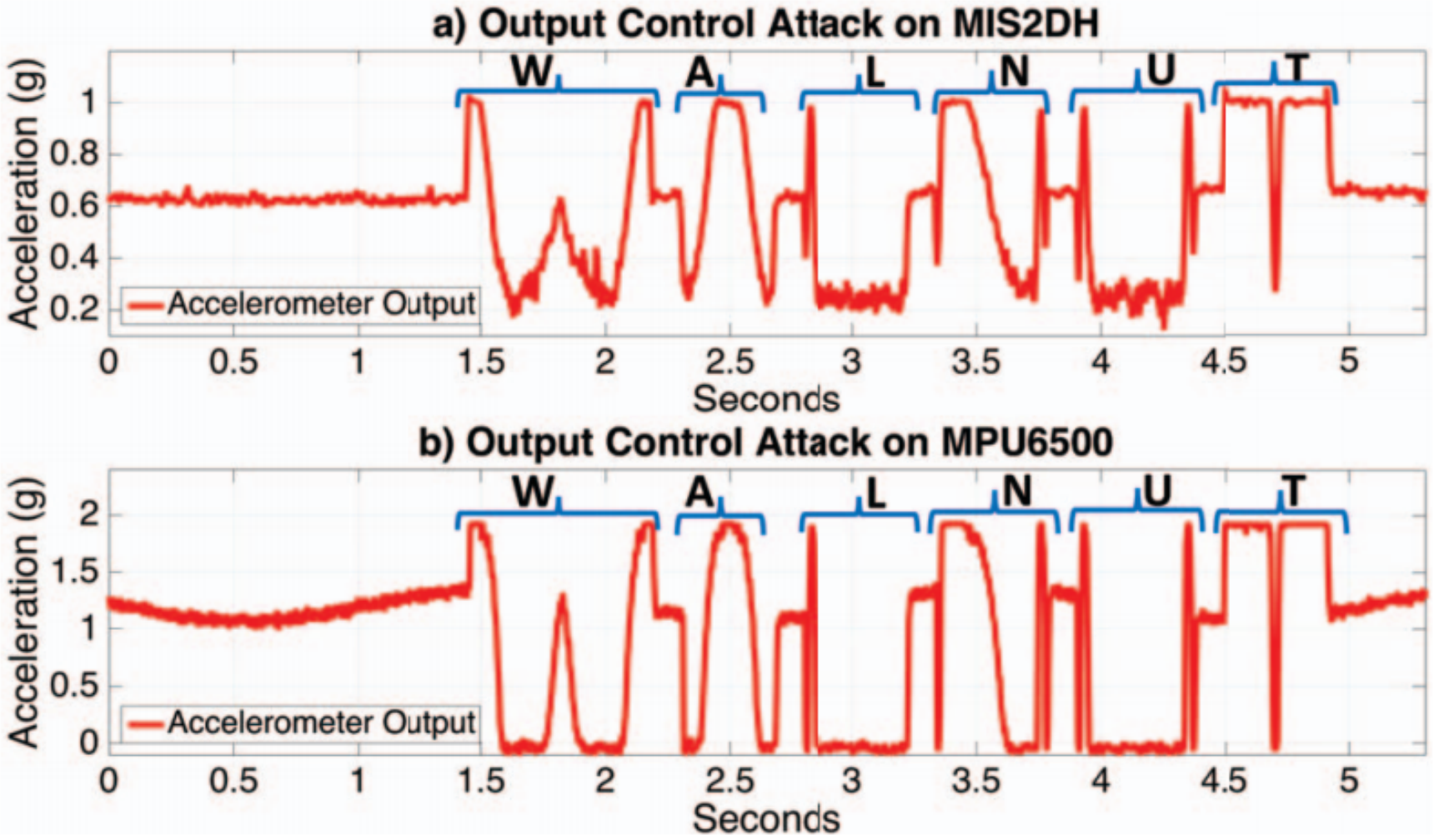


Output Control Attack

- Possible due to insecure amplifier.
- Does not need aliasing.

- Use amplitude modulation (AM) to modulate the amount of clipping at the amplifier.

Output Control Attack



How to defend?

Defense

- Ways of defense in both hardware and software:
  - LPF, Amplifier
  - Randomized Sampling, Out-of-phase Sampling

Hardware Defense

- To secure LPF
  - If there’s no LPF, add one.
  - If resonant frequency not in stop band, use another filter or re-design the system to exhibit a higher resonant frequency.
- To secure Amplifier:
  - Use one with higher dynamic range.
  - Filter out resonant frequency before the amplifier.

Software Defense (Randomized Sampling)

- Prevents output biasing attack from generating DC alias.
- Add random delay $t_{delay} \sim Unif[0, \frac{1}{F_{res}}]$ to sampling interval, get new sampling interval $t_k^* = t_k + t_{delay}$
- By (3), $2\pi F_a t_k^* \sim Unif[2\pi Nk, 2\pi(Nk+1)]$. The sinusoid is uniform over one cycle.

Software Defense (Out-of-Phase Sampling)

- Attenuates frequency around resonance.
- Acts like band-stop filter.

- Take two samples with 180° phase delay with respect to resonant frequency. Namely, two samples at t_k, $t_k + t_{delay}$ where $t_{delay} = \frac{1}{2F_{res}}$

Software Defense (Out-of-Phase Sampling)

- Examine the acoustic signal $s_a(t)$, its sampled version is

$$\begin{aligned} s_a(t_k + t_{\text{delay}}) =& A_0 A_1 cos(2\pi F_a(t_k + t_{\text{delay}}) + \phi) \\ =& A_0 A_1 cos(2\pi F_a t_k + \pi + \phi) \\ =& -s_a(t_k) \end{aligned} \tag{4}$$

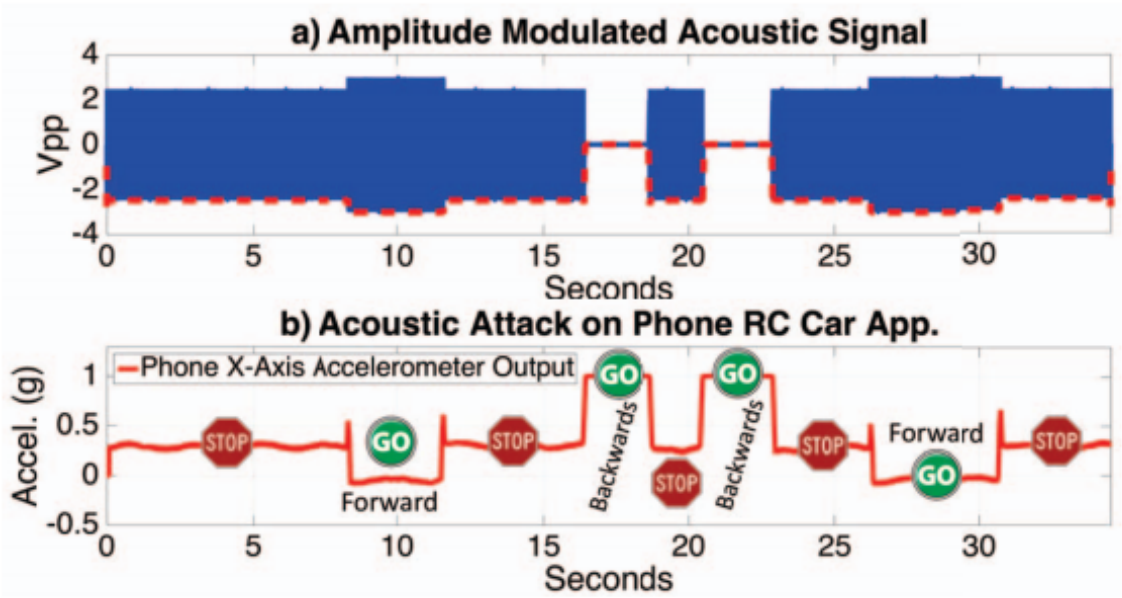
- Assuming the max frequency of the true acceleration signal is much smaller than the resonant frequency, by the virtue of (4)

$$\frac{1}{2}(\hat{s}(t_k) + \hat{s}(t_k + t_{\text{delay}})) \approx \frac{1}{2}(2s(t_k) + 0) = s(t_k) \tag{5}$$

- With this sampling scheme, acoustic signal is averaged out.

Evaluations

- Control wireless RC car



- Can also get fake Fitbit steps.


Timothy Trippel
2ND YEAR, PH.D. CANDIDATE
COMPUTER SCIENCE AND ENGINEERING
UNIVERSITY OF MICHIGAN

TABLE 1. **ACCELEROMETER RESONANT FREQUENCIES**: UNDER RESONANT ACOUSTIC INTERFERENCE, AN OUTPUT BIASING ATTACK CLASS INDICATES A SENSOR'S FALSIFIED MEASUREMENTS FLUCTUATE (INSECURE LPF) WHILE AN OUTPUT CONTROL ATTACK CLASS INDICATES CONSTANT FALSIFIED MEASUREMENTS ARE OBSERVED (INSECURE AMPLIFIER). TWO INSTANCES OF EACH SENSOR WERE TESTED.

Model	Type	Typical Usage	Resonant Frequency (kHz)			Amplitude (g)*	Attack Class‡		
			X	Y	Z		X	Y	Z
Bosch - BMA222E	Digital	Mobile devices, Fitness	5.1–5.35	–	9.4–9.7	1	B	–	BC
STM - MIS2DH	Digital	Pacemakers, Neurostims	–	–	8.7–10.7	1	–	–	BC
STM - IIS2DH	Digital	Anti-theft, Industrial	–	–	8.4–10.8, ...	1.2	–	–	BC
STM - LIS3DSH	Digital	Gaming, Fitness	4.4–5.2	4.4–5.6	9.8–10.2	1.6	BC	BC	BC
STM - LIS344ALH	Analog	Antitheft, Gaming	2.2–6.6	2.2–5.7	2.2–5.6	0.6	B	B	B
STM - H3LIS331DL	Digital	Shock detection	–	–	11–13, ...	5.2	–	–	BC
INVN - MPU6050	Digital	Mobile devices, Fitness	5.35	–	–	0.75	BC	–	–
INVN - MPU6500	Digital	Mobile devices, Fitness	5.1, 20.3	5.1–5.3	–	1.9	BC	C	–
INVN - ICM20601	Digital	Mobile devices, Fitness	3.8, ...	3.3, ...	3.6, ...	1.1	BC	BC	BC
ADI - ADXL312	Digital	Car Alarm, Hill Start Aid	3.2–5.4	2.95–4.75	9.5–10.1	1.3	B	B	BC
ADI - ADXL337	Analog	Fitness, HDDs	2.85–3.1	3.8–4.4	–	0.8	B	B	–
ADI - ADXL345	Digital	Defense, Aerospace	4.4-5.4	3.1–6.8	4.4–4.7	7.9	BC	BC	B
ADI - ADXL346	Digital	Medical, HDDs	4.3–5.1	6.1	4.95, ...	1.75	B	B	B
ADI - ADXL350	Digital	Mobile devices, Medical	2.5–6.3	2.5–4	2.5–6.8	1.8	B	B	B
ADI - ADXL362	Digital	Hearing Aids	4.2–6.5, ...	4.3–6.5, ...	4.5–6.5	1.4	BC	BC	BC
Murata - SCA610	Analog	Automotive	–	–	–	–	–	–	–
Murata - SCA820	Digital	Automotive	24.3	–	–	0.13	C	–	–
Murata - SCA1000	Digital	Automotive	–	–	–	–	–	–	–
Murata - SCA2100	Digital	Automotive	–	–	–	–	–	–	–
Murata - SCA3100	Digital	Automotive	7.95	–	8	0.15	C	–	C

* Amplitude is taken as the maximum false output measurement observed.
‡ **B** = Output Biasing Attack; **C** = Output Control Attack (Red Highlight)
STM = STMicroelectronics; ADI = Analog Devices; INVN = InvenSense
– Experiments found no resonance
... Additional ranges of resonance elided

Summary

- Can take control over MEMS accelerometers by exploiting the hardware deficiencies.
- The desired attack is realized.
- Software is tricked as well.
- Proposed methods that could protect new sensors (hardware defense) as well as existing ones (software defense).

- The setup is ideal, where the distance between sensor and speaker is fixed and evaluated in sound-isolating chamber → what is the working range and if attack is defined in the way the paper does it, how robust is it in real life?