WIRELESS SENSING

CSE 599 N1: Modern Mobile Systems

modernmobile.cs.washington.edu

Content borrowed from Rajalakshmi Nandakumar

FINGERIO

FingerIO

Achieves finger tracking for near device interaction without instrumentation the finger and in occluded (non-line of sight) scenarios

1) Making anything an input surface
2) Move beyond tiny screens

Existing approaches: Doppler Radar (Soli)

Complex processing

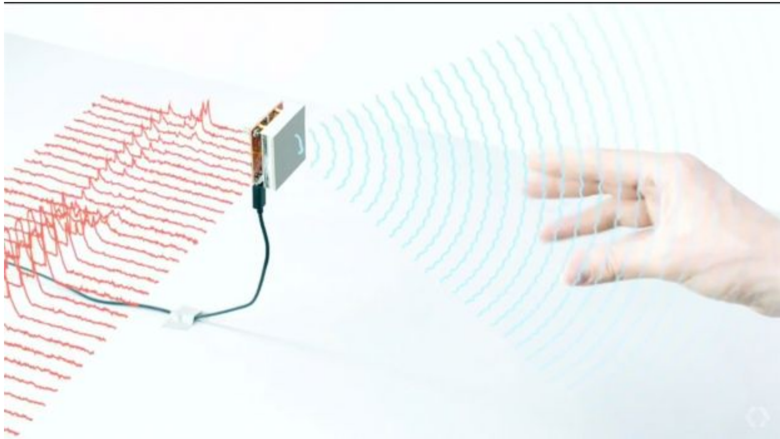
RF travels at the speed of light ⇒ needs GHz of bandwidth (for sampling)

Mobile devices have limited processing

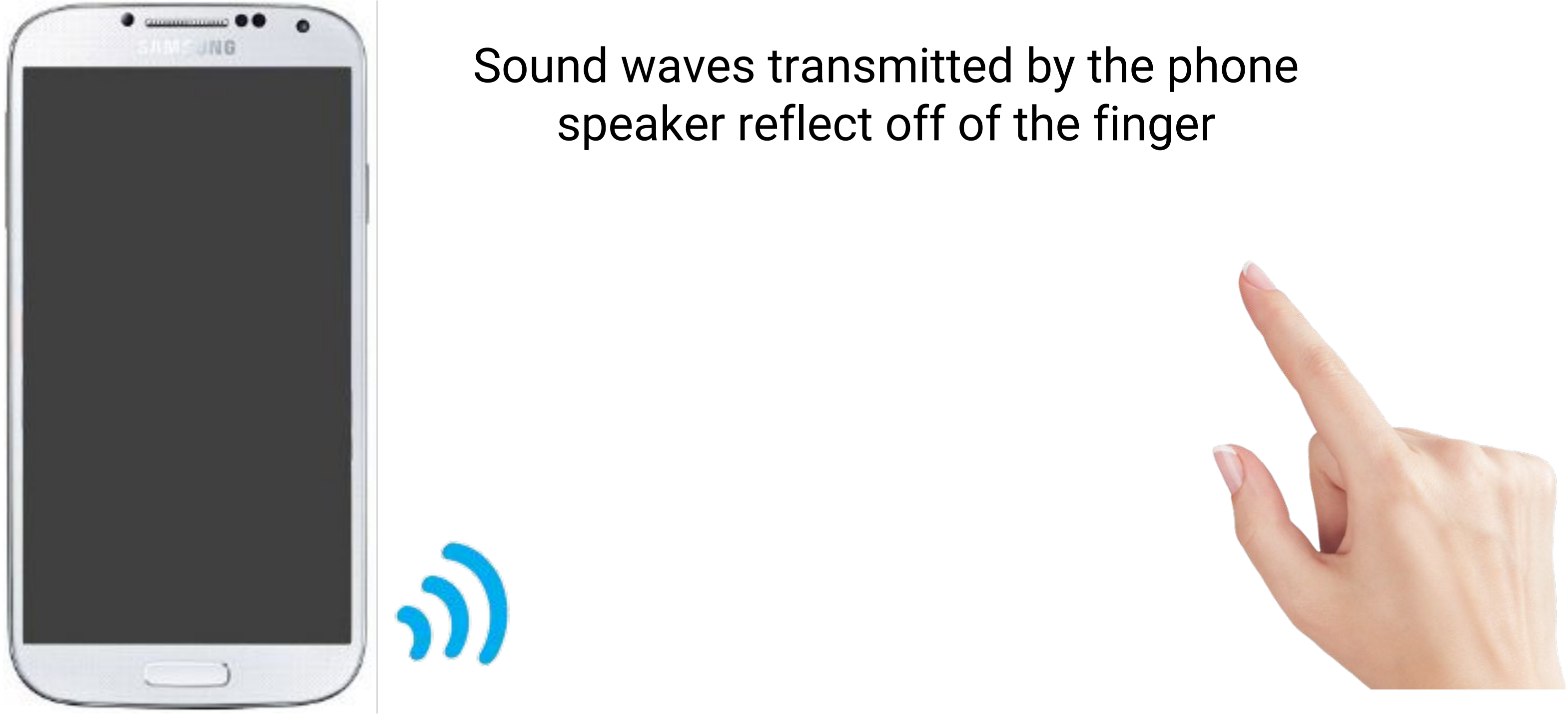
Custom hardware

Requires yet another radio on mobile devices

Consumes additional power and area on devices



Key Idea: Transform the device into Active Sonar

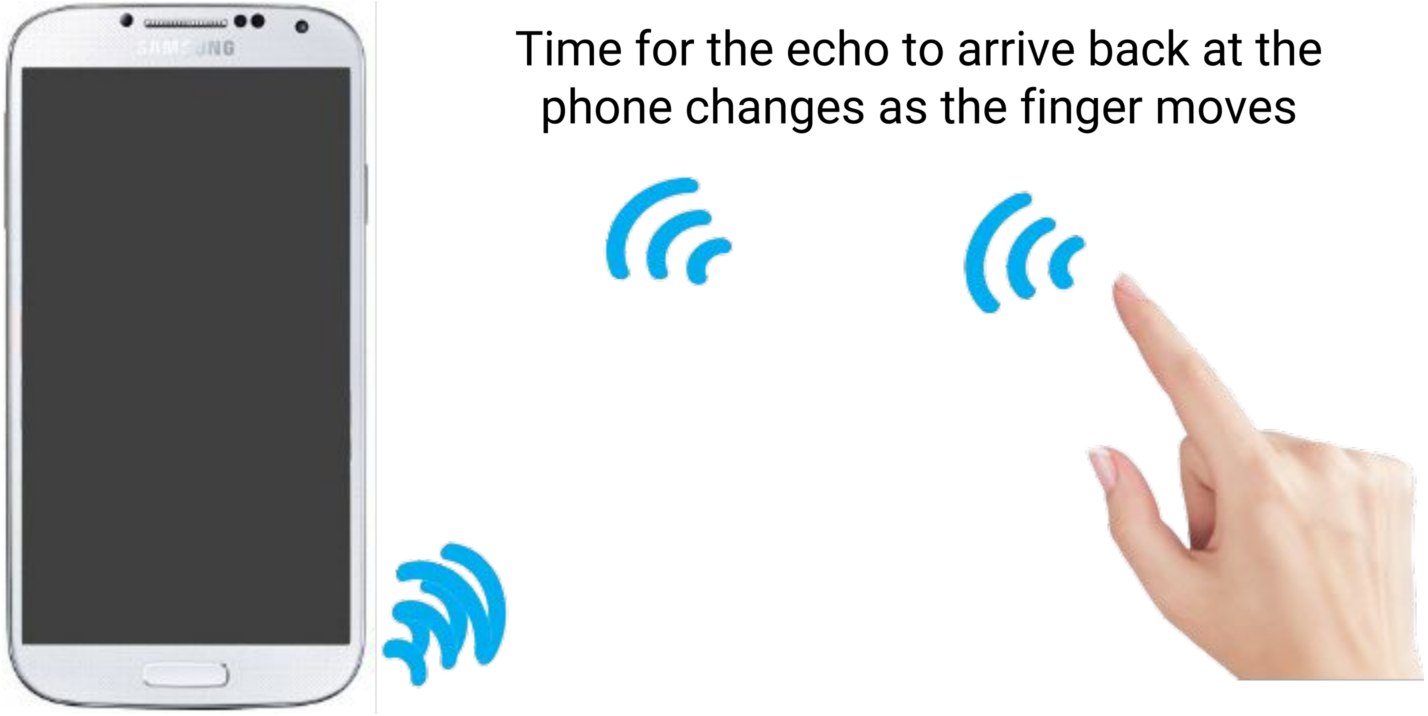


Sound waves transmitted by the phone speaker reflect off of the finger

Key Idea: Transform the device into Active Sonar

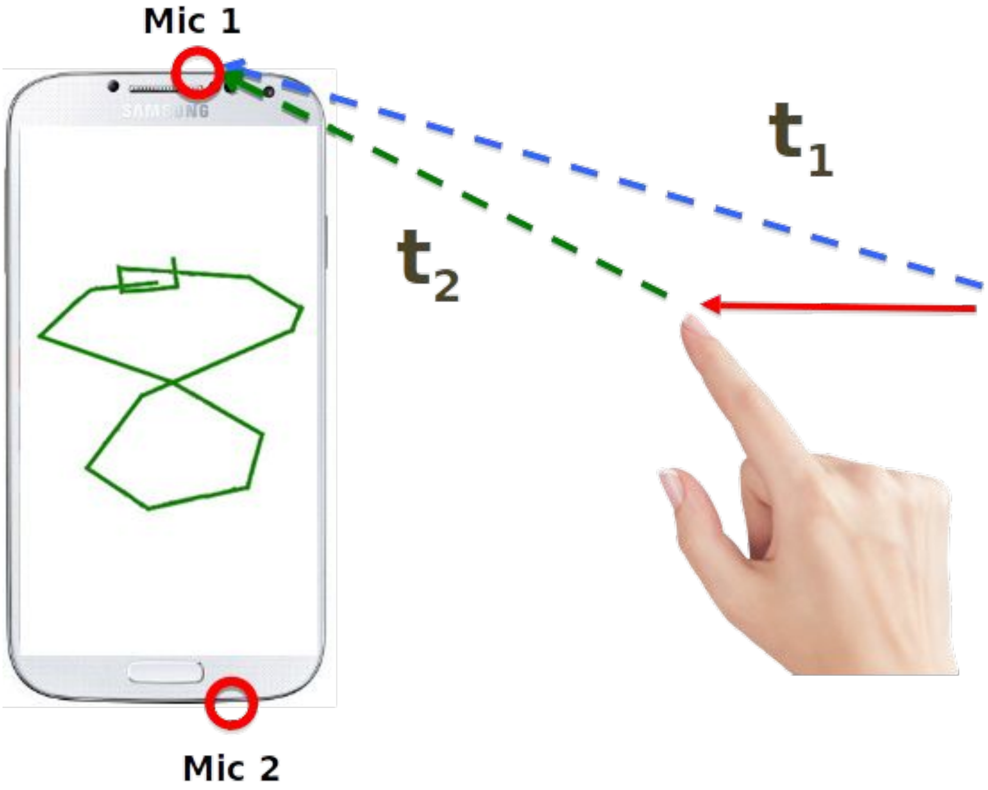
Echo from finger is recorded
by 2 microphones

Key Idea: Transform the device into Active Sonar



Time for the echo to arrive back at the phone changes as the finger moves

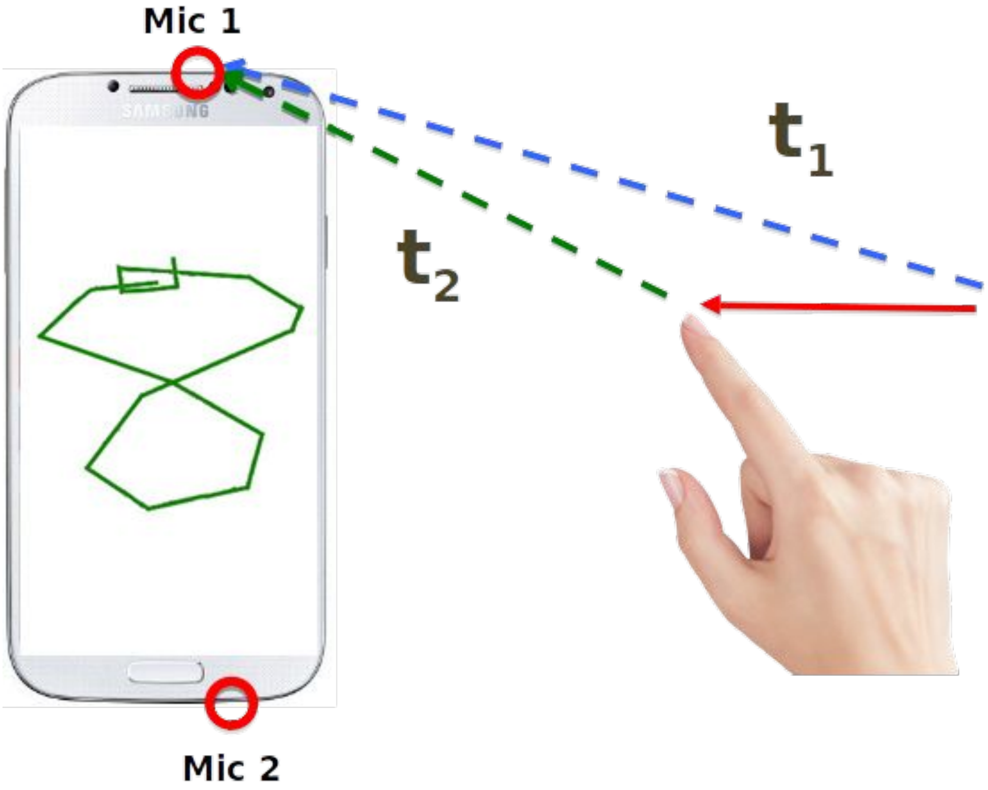
Accuracy Depends on Time Measurement



If our sampling rate is 48 kHz and the speed of sound of 343 m/s

How much does one sample correspond to?

Accuracy Depends on Time Measurement



If our sampling rate is 48 kHz and the speed of sound of 343 m/s

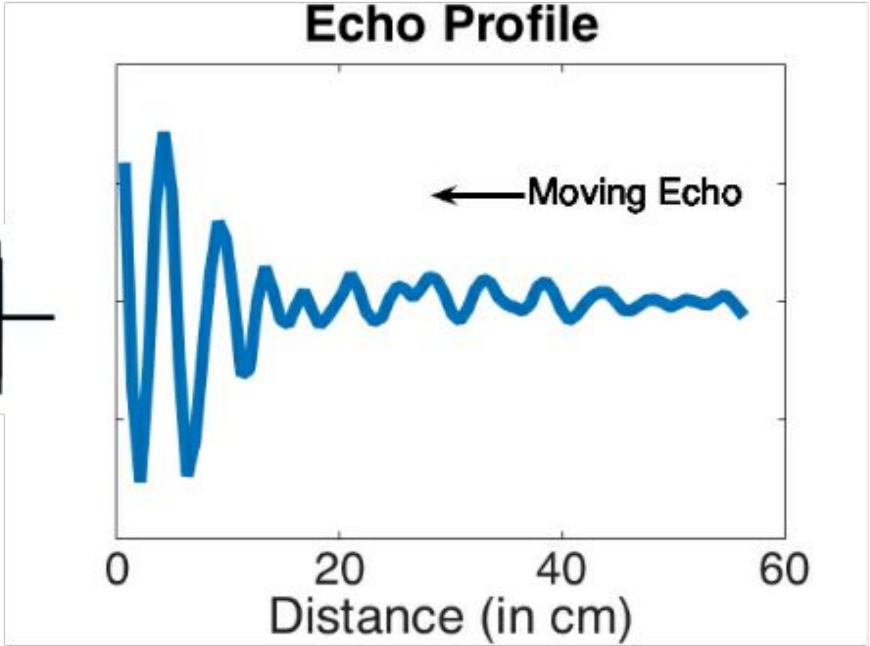
How much does one sample correspond to?

48000 times per second
343 / 48000 = 0.007m = 0.7 cm
Sub-centimeter!

How can we measure arrival time?

Transmit chirp signals and use autocorrelation to determine arrival times

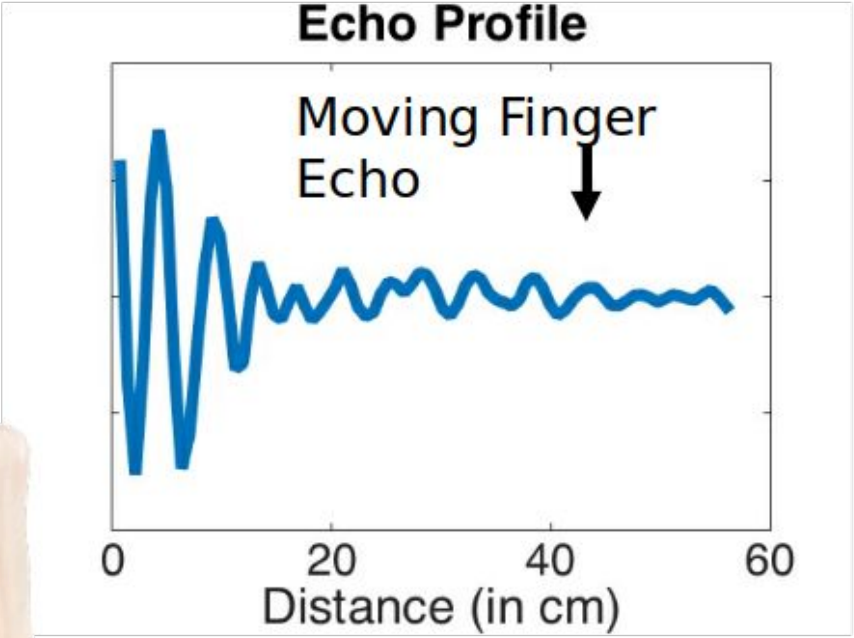
Chirp



First order solution: correlation

We use the closest moving echo to achieve finger tracking

Chirp



Correlation in practice

Estimate echo arrival with 2 - 3 sample error!

How much of an error is that in centimeters?

Correlation in practice

Estimate echo arrival with 2 - 3 sample error!

How much of an error is that in centimeters?

2 * 0.7 cm = 1.4 cm

3 * 0.7 cm = 2.1 cm

How to get the exact arrival time of the echoes?

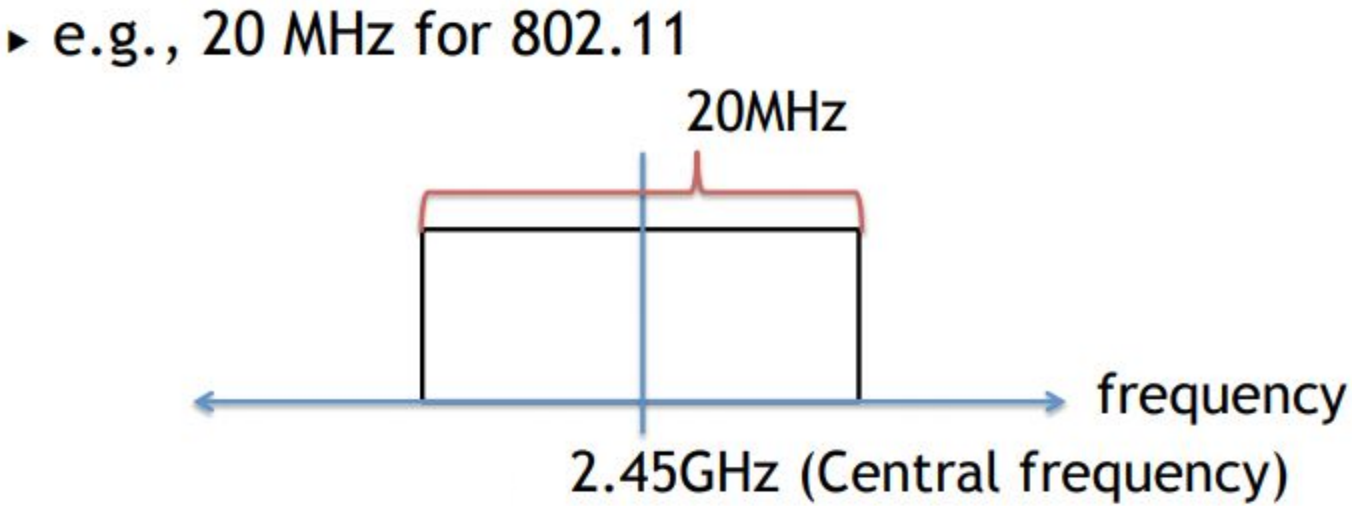
Inspiration from WiFi networks

Transmitters and receivers do not share a common, synchronized clock

Receivers need to determine the start of a message to successfully decode

OFDM

Wi-Fi has 20 MHz of bandwidth to transmit information.

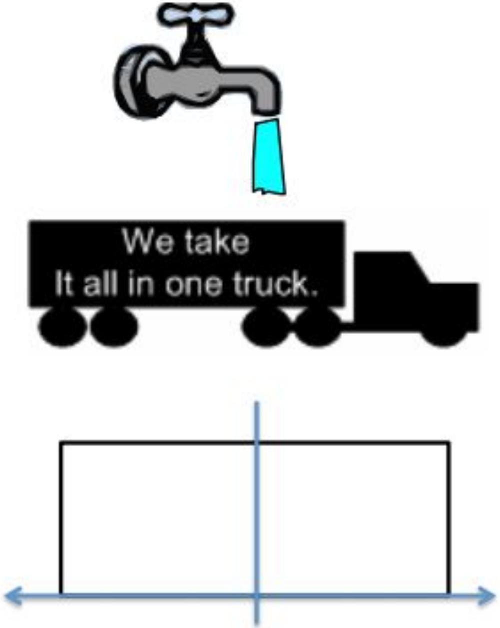


- e.g., 20 MHz for 802.11

Basic Concept of OFDM

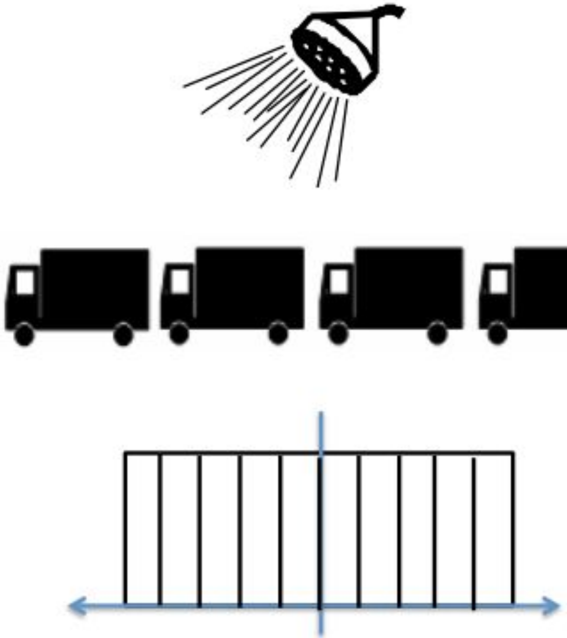
Each ‘truck’ is a subcarrier

Wide-band channel



Send a sample using the entire band

Multiple narrow-band channels

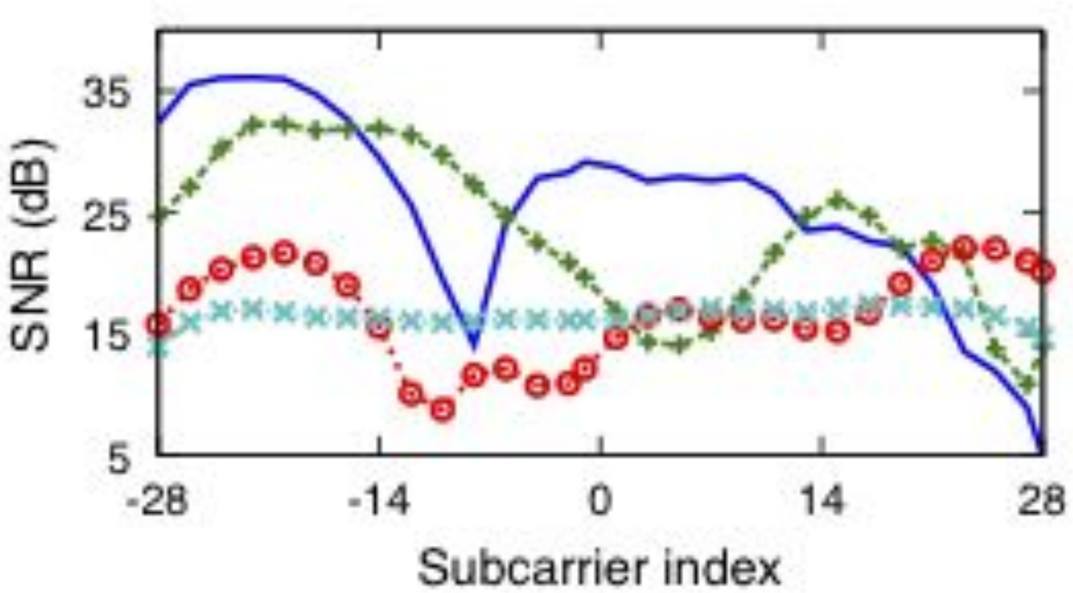


Send samples concurrently using multiple **orthogonal sub-channels**

Guards against non-uniform ‘frequency response’

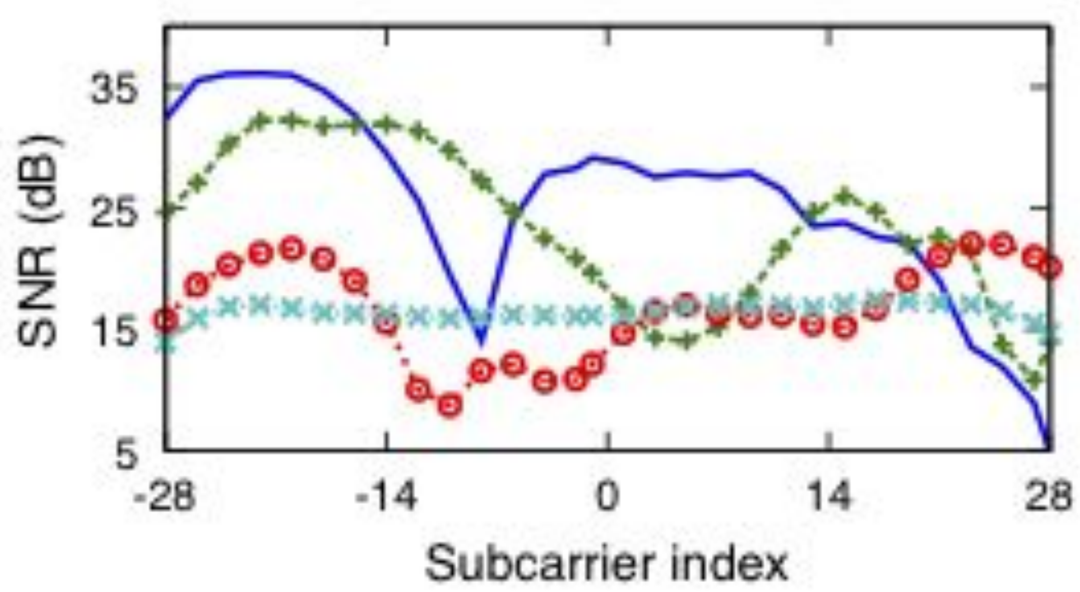
Remember how across every channel, frequency response is different.

Frequency response for different SISO links



Possible reasons?

Frequency response for different SISO links



Possible reasons?

Interference (Wi-Fi ISM band is super-crowded) Microwaves etc.

Multi-path (reflections)

Fading

Capacity of channel

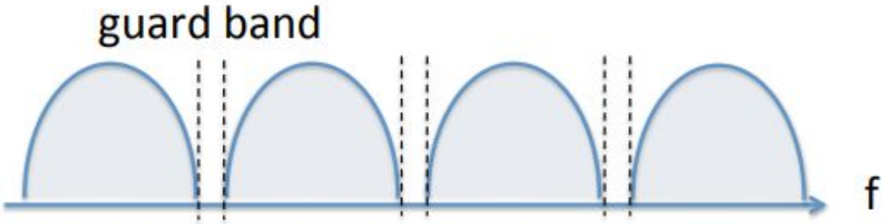
The wider your bandwidth, the more data you can send

The capacity of a channel is related to bandwidth and SNR

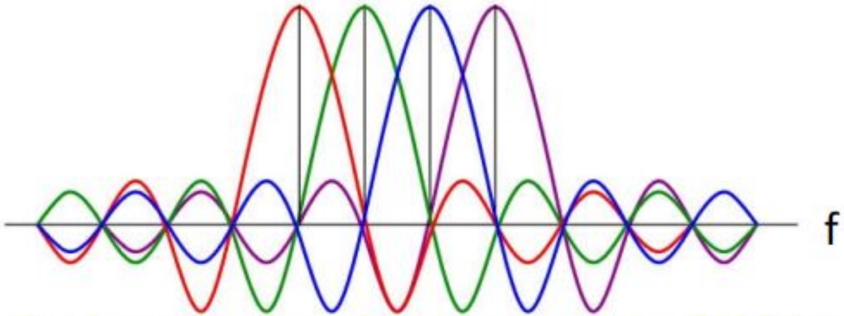
$$\text{Capacity} = \text{BW} * \log(1+\text{SNR})$$

OFDM has almost same bandwidth as wide-band channel

Difference between FDM and OFDM

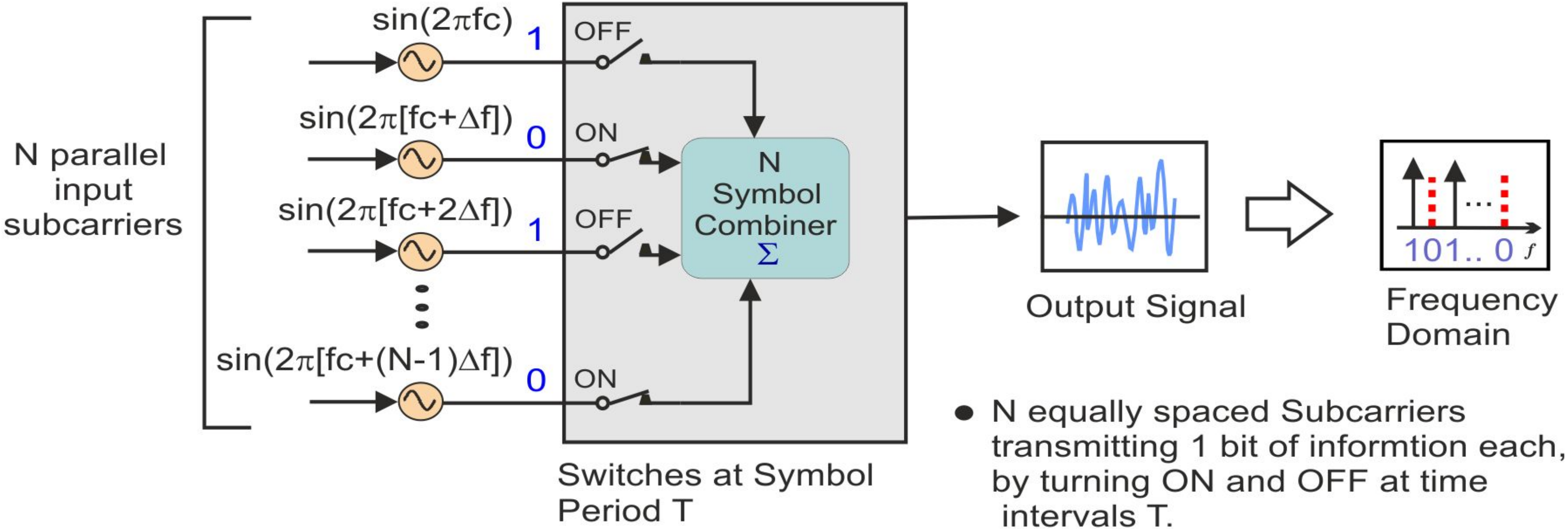


Frequency division multiplexing



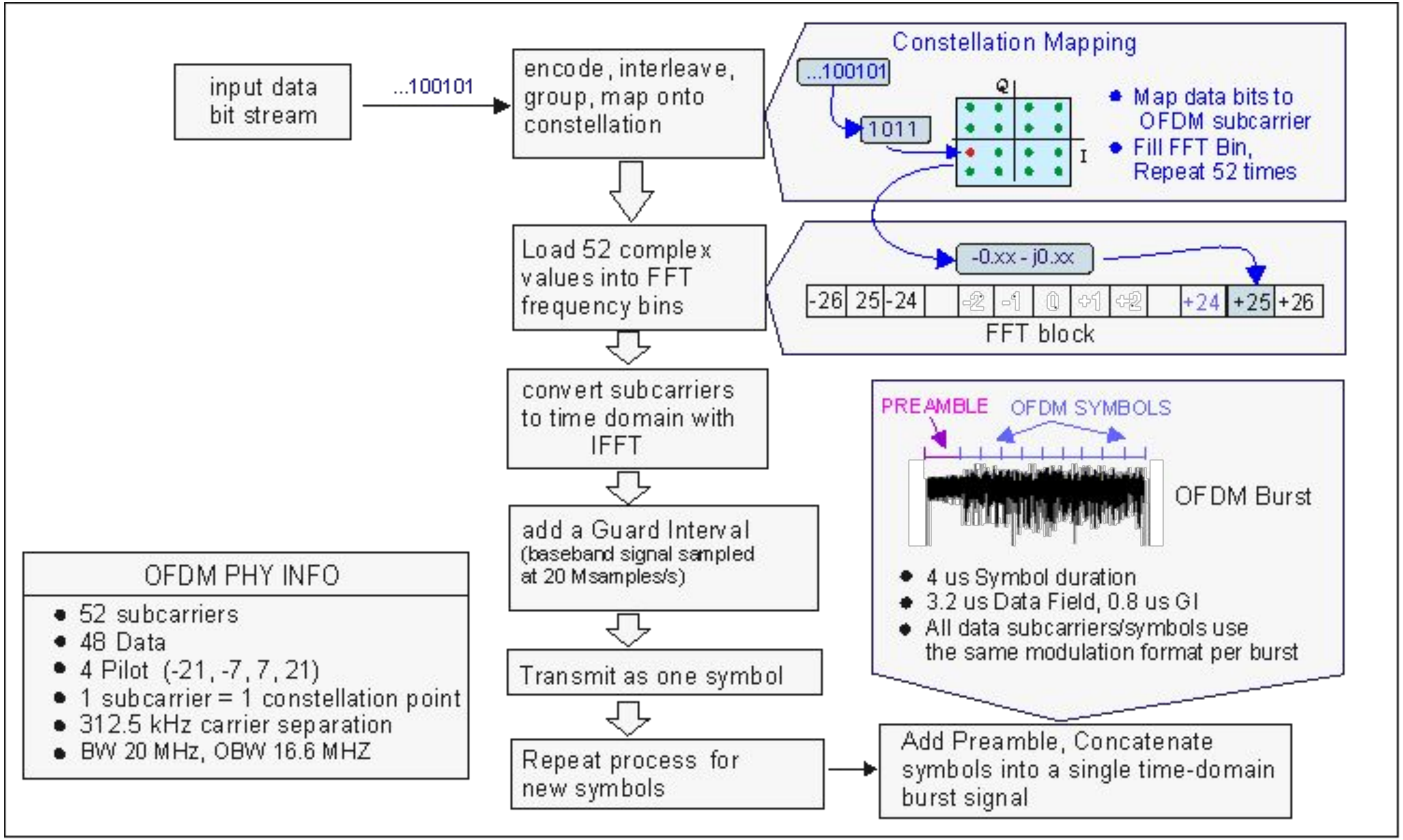
Orthogonal sub-carriers in OFDM

Don’t need guard bands



- N equally spaced Subcarriers transmitting 1 bit of informtion each, by turning ON and OFF at time intervals T.

Simple OFDM Generation



input data bit stream

...100101

encode, interleave, group, map onto constellation

Constellation Mapping

...100101

1011

Q

I

- Map data bits to OFDM subcarrier
- Fill FFT Bin, Repeat 52 times

Load 52 complex values into FFT frequency bins

-0.xx - j0.xx

-26	25	-24		-2	-1	0	+1	+2		+24	+25	+26

FFT block

convert subcarriers to time domain with IFFT

add a Guard Interval (baseband signal sampled at 20 Msamples/s)

Transmit as one symbol

Repeat process for new symbols

Add Preamble, Concatenate symbols into a single time-domain burst signal

PREAMBLE

OFDM SYMBOLS

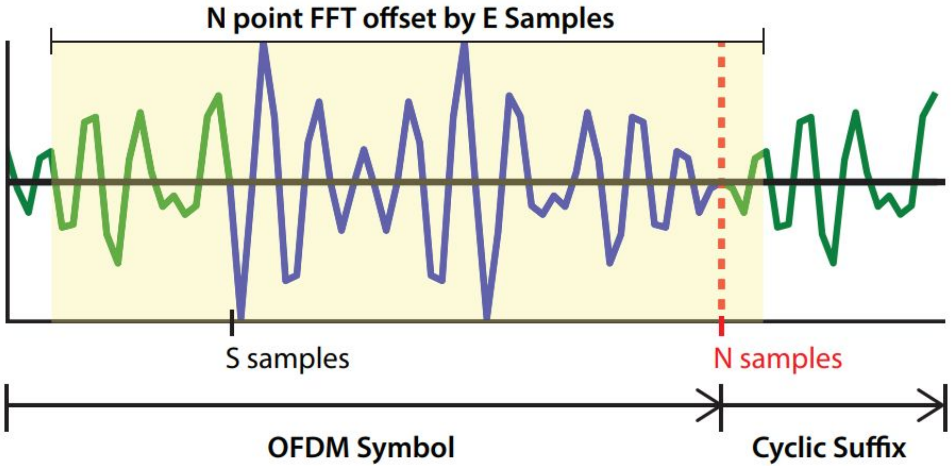
OFDM Burst

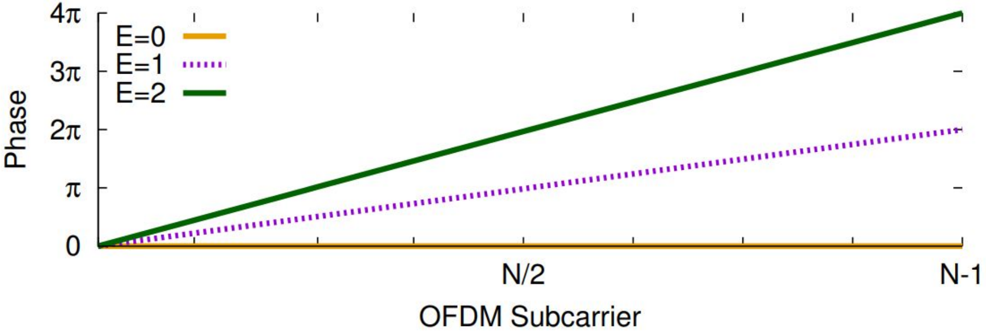
- 4 us Symbol duration
- 3.2 us Data Field, 0.8 us GI
- All data subcarriers/symbols use the same modulation format per burst

OFDM PHY INFO

- 52 subcarriers
- 48 Data
- 4 Pilot (-21, -7, 7, 21)
- 1 subcarrier = 1 constellation point
- 312.5 kHz carrier separation
- BW 20 MHz, OBW 16.6 MHZ

802.11a OFDM Signal Generation Process





1) At transmitter, run IFFT to get OFDM symbol
Prepend + append the green samples to get a cyclic suffix/periodic signal

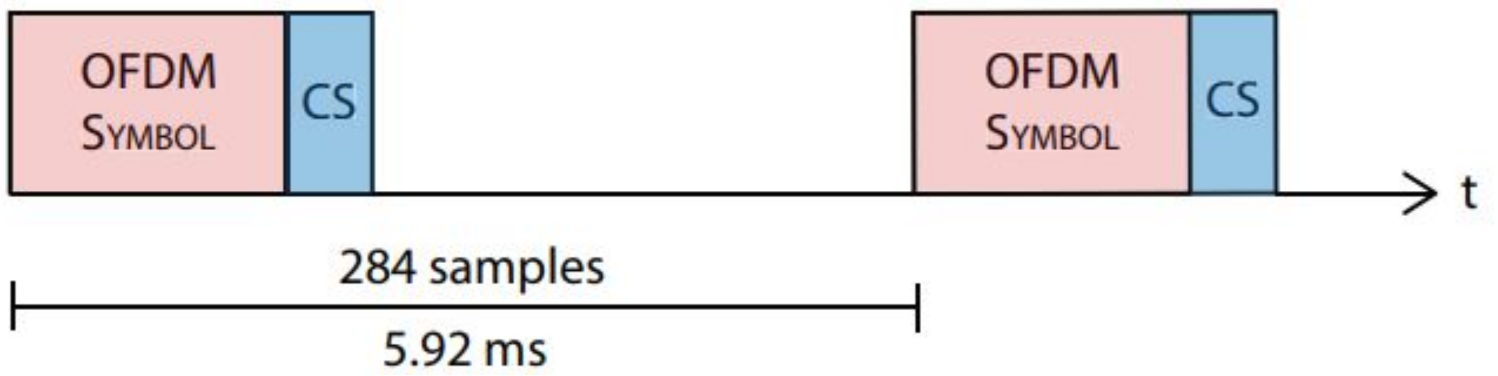
2) The yellow box is where you take the FFT over at receiver
If the receiver window is aligned perfectly over the signal, the phase is 0 across all subcarriers

3) If the receiver window is off by E samples, we notice phase offsets

Now we can correct the receiver window to sample precisely!

Now we get sub-centimeter resolution

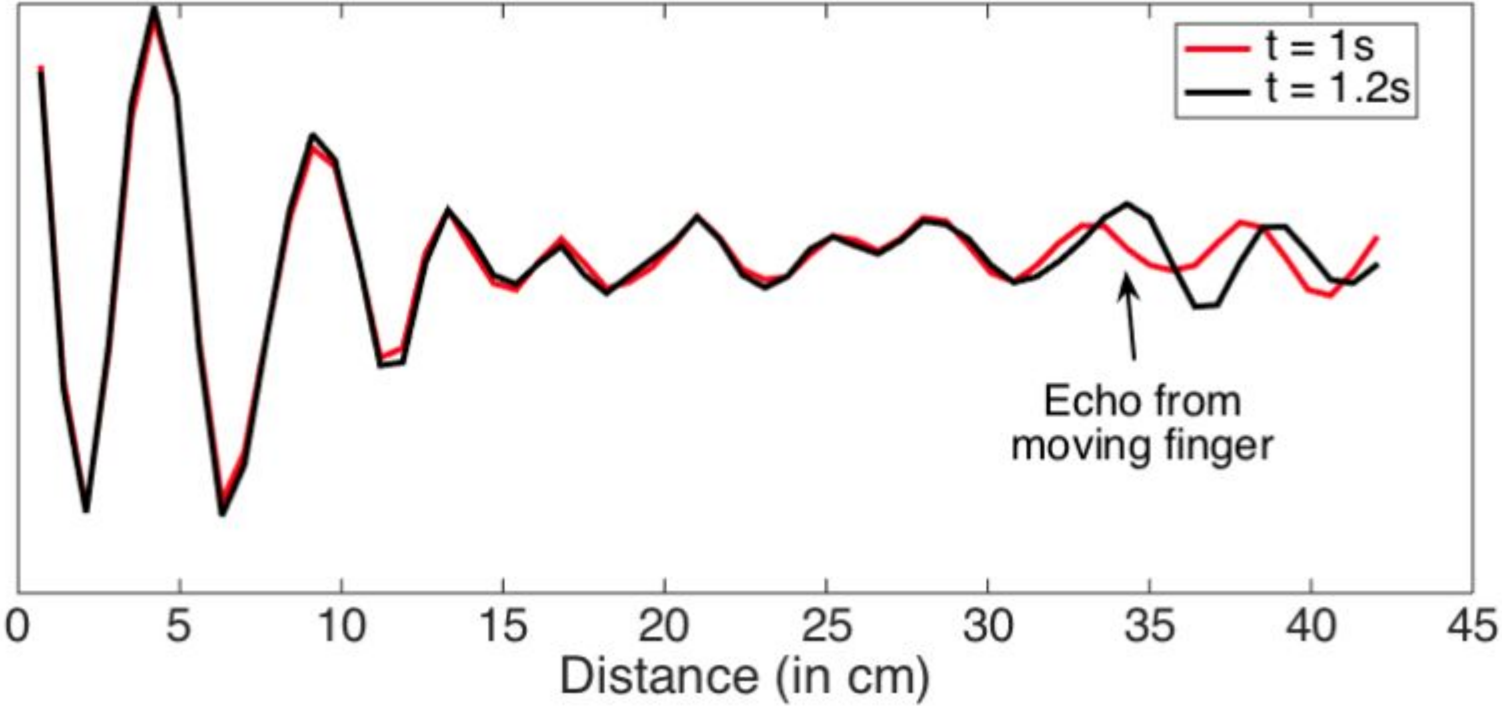
Putting it all together



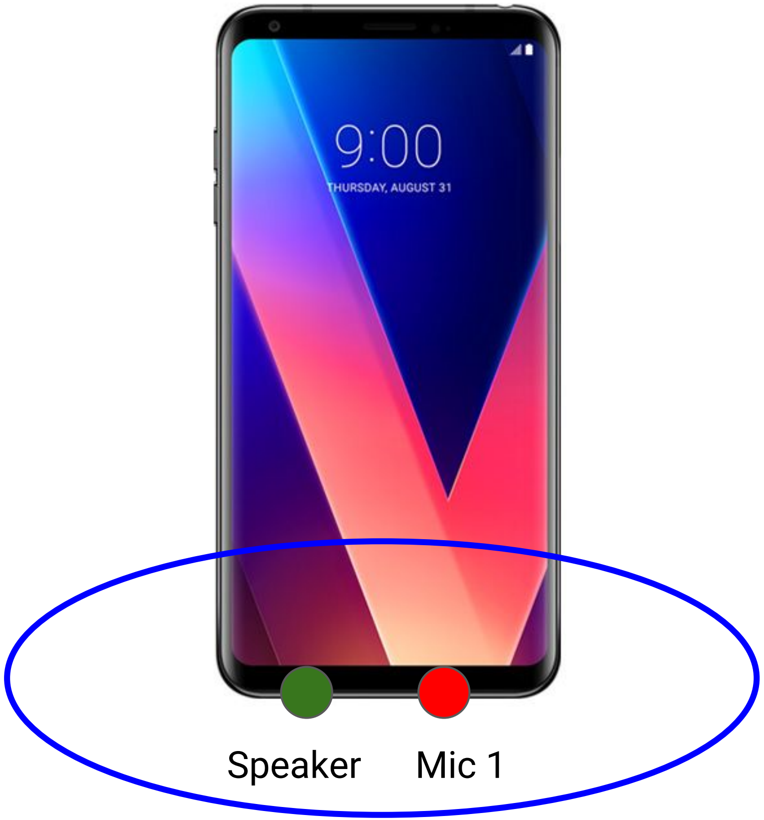
1. Transmit 18-20 kHz OFDM symbols every 5.92 ms
2. Use correlation to get a coarse timing estimate within 2-3 samples
3. Correct error using phase properties of OFDM to achieve < 1 cm accuracy

Determining user motion

Do a ‘diff’ on the echo profiles to see if there’s motion



How to get 2D location?



The speaker and mic are the focii of an ellipse

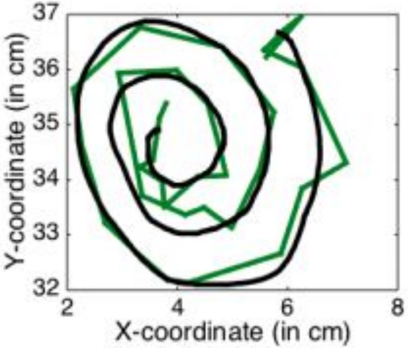
How to get 2D location?

The speaker and mic are the focii of an ellipse

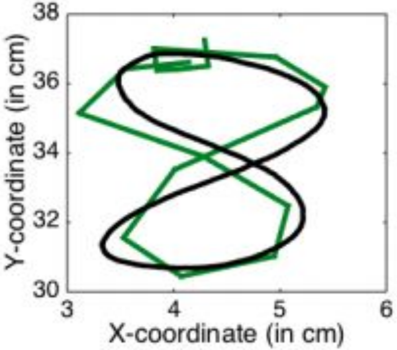
With two mics, you get 2 intersection points

User can interact on either side of the phone

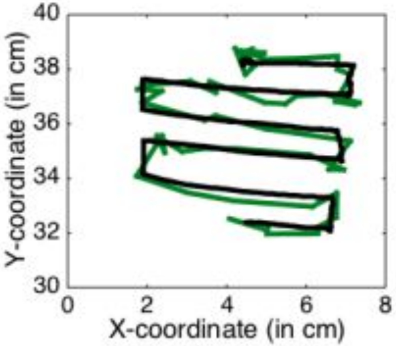
Results



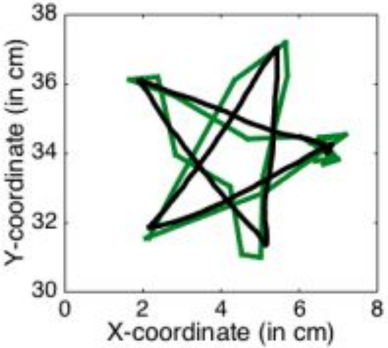
37
36
35
34
33
32
Y-coordinate (in cm)
2
4
6
8
X-coordinate (in cm)



38
36
34
32
30
Y-coordinate (in cm)
3
4
5
6
X-coordinate (in cm)

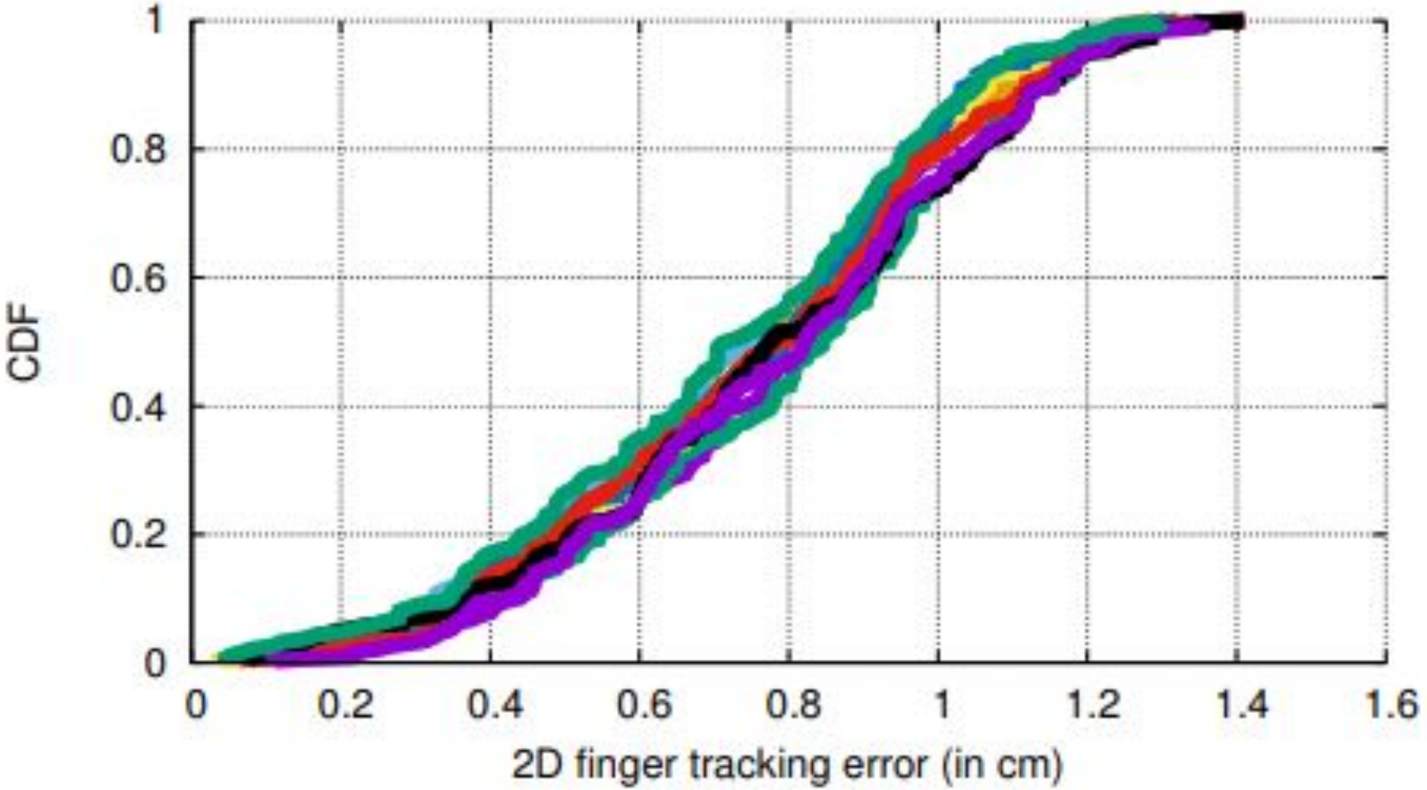


40
38
36
34
32
30
Y-coordinate (in cm)
0
2
4
6
8
X-coordinate (in cm)



38
36
34
32
30
Y-coordinate (in cm)
0
2
4
6
8
X-coordinate (in cm)

Tracking error/how to read a CDF



WISEE

Doppler effect

f_r = frequency at receiver

f_t = frequency at transmitter

c = velocity of waves, speed of light = 3e8

v = velocity of hand, tests show fastest speed is 3.9m/s

Away from laptop = lower frequency shift

Toward laptop = higher frequency shift

$$f_r = f_t \cdot \left(\frac{c + v}{c - v} \right)$$

Frequency shift $\frac{2f}{c} v$

$$\Delta f \propto \frac{2v cos(\theta)}{c} f$$

Doppler shift maximized when object moves *towards* the receiver

FFT bin width?

Sampling rate is 44.1kHz

2048 point FFT

Pilot tone of 20 kHz

FFT bin width?

Sampling rate is 44.1kHz

2048 point FFT

Pilot tone of 20 kHz

Spectral width = 22.05 kHz

1024 FFT bins because we only take the left side of the FFT

22050/1024 = 21.5 Hz per bin

How many bins to search through?

Speed = 6m/s

$fr=ft*((c+v)/(c-v))$

Fr = 20.712 kHz

Frequency shift is 712 Hz

712 Hz / 21.5 Hz = 33 bins

Slowest speed we can detect?

Min bin size is 21.5 Hz

f_r = f_t + 21.5 Hz

v=(c*(f_r-f_t))/(f_t+f_r)

v= 0.18 m/s = 18 cm/s

Wi-Fi doppler shifts

Wi-Fi f_t = 5 GHz

v = 6 m/s

Frequency shift is 200 Hz

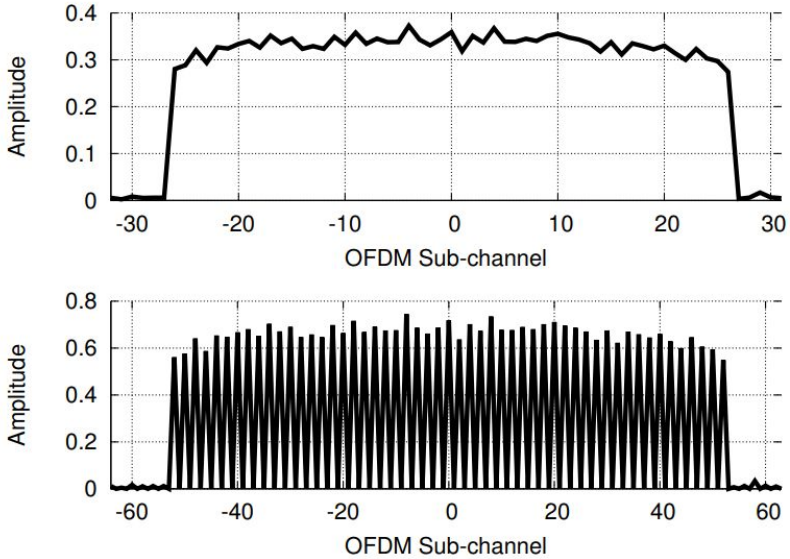
The ‘typical’ wireless channel has 20 MHz (or more) of bandwidth

Why 5 GHz and not 2.4 GHz?

At higher f_t, the Doppler shift will be greater

Getting fine grained resolution

Take FFT over M OFDaM symbols (with a single symbol the N-point FFT has a low N because of the low duration, with multiple symbols, you can take an MN point FFT). Each frequency bin has higher resolution



FFT over 2 OFDM symbols

Both symbols carry the same information, so your bandwidth effectively halves.

Odd sub-channels are zero

Bandwidth reduces by a factor of M

You now have a narrow-band signal

Arbitrary OFDM symbols

Perform FFT over symbol, when you know the bits that were transmitted you can ‘equalize’ it to the first OFDM symbol

$\mathbf{X}_n^i$ i = symbol, n = subcarrier

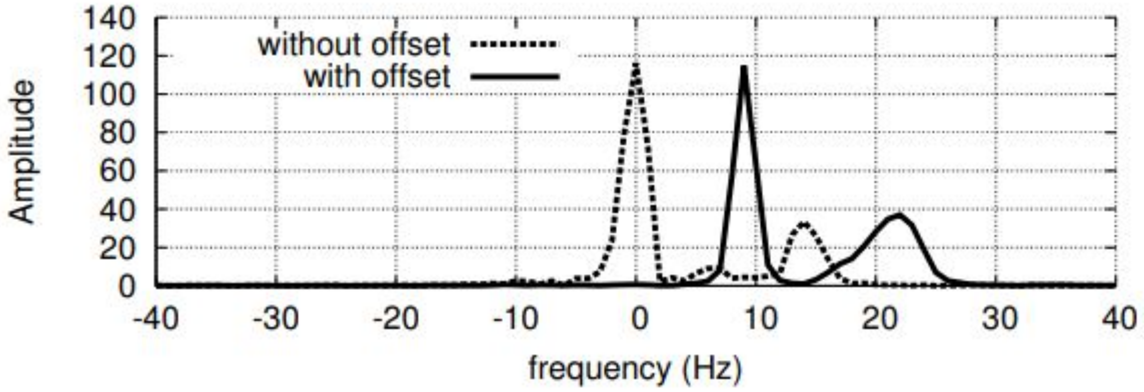
$\frac{\mathbf{X}_n^1}{\mathbf{X}_n^i}$ Multiply this term over every incoming channel on every symbol

Frequency offsets

Frequency offsets - oscillator (made of crystals) are not perfect

Transmitter and receiver don't share the same oscillator

Frequency offset is the combination of the oscillator imperfections on both



We can track the DC energy (center peak frequency) as it moves due to frequency offset.

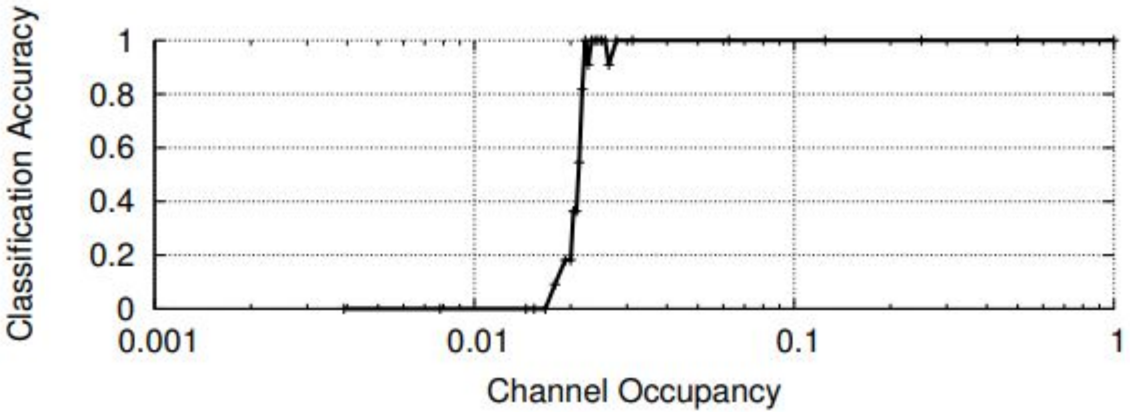
After tracking we can observe the doppler shift with respect to it

Non-continuous transmissions

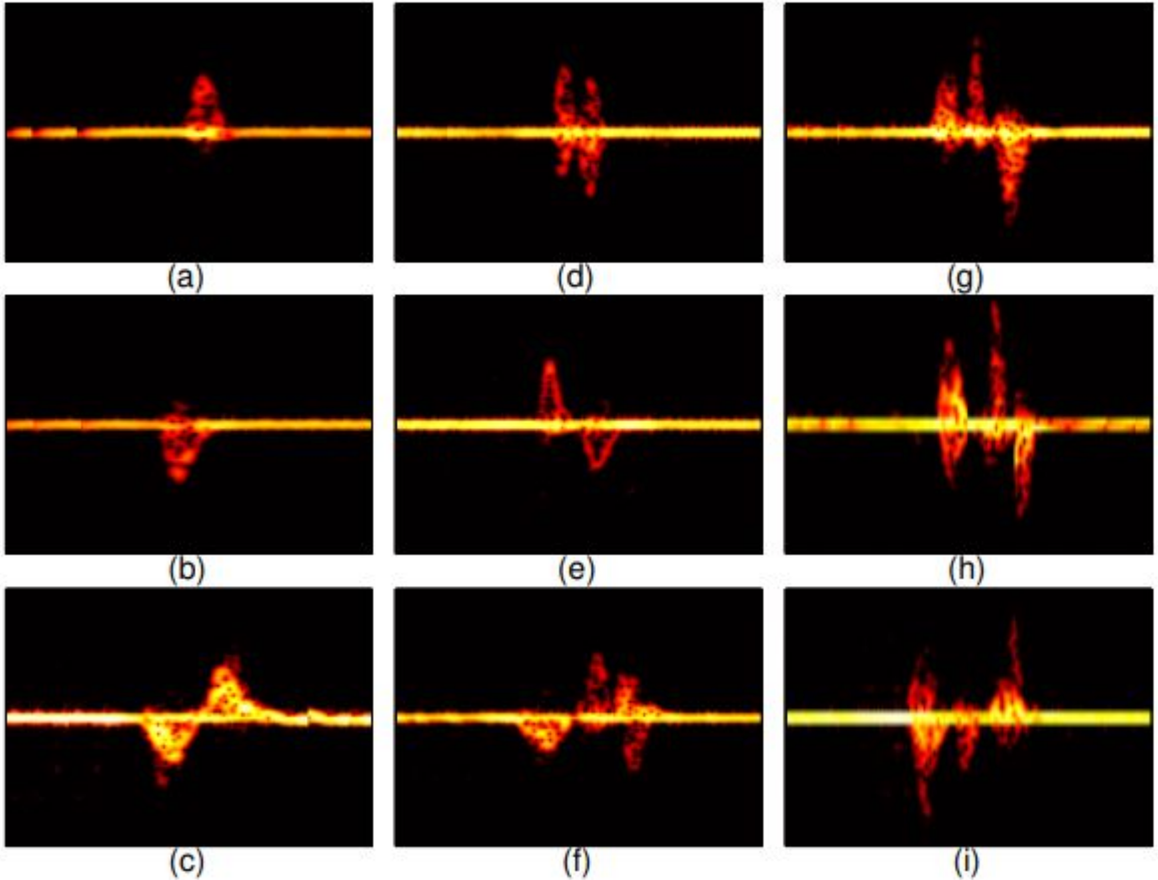
Channel is not always occupied in reality, OFDM symbols interspersed with silence

We can linearly interpolate the brief periods of time between symbols

This is ok because a gesture is a continuous motion



Of course as the symbols become more spread out (to the point that it exceeds the time to make a gesture), it becomes harder to track the doppler shifts.



Maps positive and negative shifts to a series of numbers e.g. 1, -1, 2, using thresholding

Other techniques:

Put spectrogram through ML

Dynamic time warping (provides an edit distance between signals)