Lecture 10: Orthogonal Frequency Domain Modulation (OFDM)

John M Pauly

March 8, 2026

Orthogonal Frequency Domain Modulation (OFDM))

Modern systems like WiFi, cable systems, and cell phones want to get more bits through a channel.

The simple solutions is a combination of

- More bits per symbol,
- More symbols per second

An example is cable modems that might use 256-QAM at 5 Msymbols/s

However, this approach has limitations for RF links, and even cable

A solution that solves these problems is OFDM

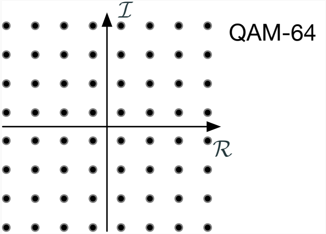
As a result, everything is going to OFDM

Based on notes from Miki Lustig

Increasing the Capacity of a Channel

So far we have been trying to get as many symbols per second through the channel, each encoded for as many bits as possible.

A good example is QAM, used in cable TV



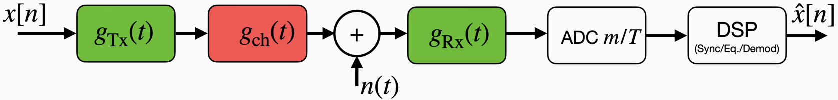
As symbol rates go up, it makes it harder to decode the symbols accurately

Timing becomes much more critical.

What we'd like is another way to increase capacity

Channel Model

To see the issues, we'll start with the channel model



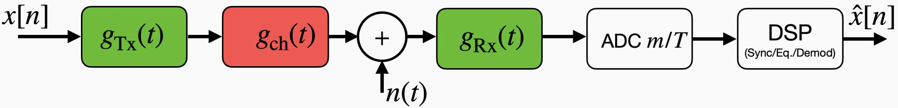
The channel:

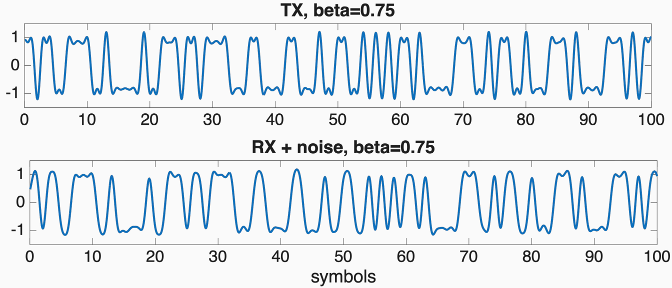
- Linear system that causes dispersion (time and frequency dependent delays)
- Impulse response $g_{ch}(t)$, and frequency response $G_{ch}(f)$
- Also adds noise

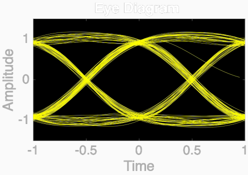
What are the effects of this?

Channel Model: Ideal

With no noise or dispersion:



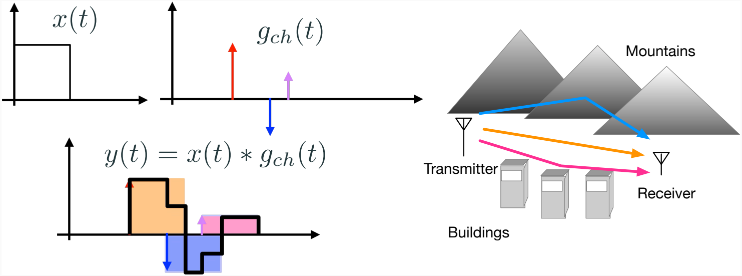




Nice clean eye diagram

Channel Model: Real

In practice we will have multipath:

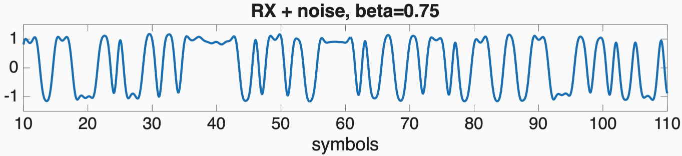


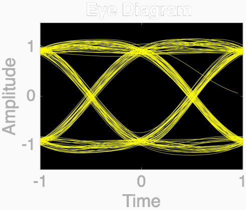
The impulse response of the channel can be represented by shifted, weighted, delta functions

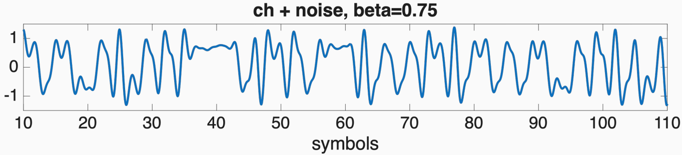
$$h(t) = a_0\delta(t - t_0) + a_1\delta(t - t_1) + a_2\delta(t - t_2)$$

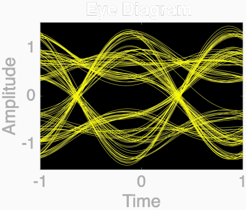
Channel Model: Real

With noise and dispersion:









$$g_{\mathrm{ch}}(t) = \delta(t) - 0.5\delta(t - 0.75T_b) + 0.25\delta(t - 1.5T_b)$$

Much harder to decode!

Channel Model: Delay and Coherence

Multipath has a set of delays, $\tau_k = \tau_0 + \Delta\tau_k$

$$g_{\text{ch}}(t) = \alpha_0 \delta(t - \tau_0) + \sum_{k=1}^{m} \alpha_k \delta(t - \Delta\tau_k - \tau_0)$$

For analysis, we can neglect τ_0, since it is a constant

$$g_{\text{ch}}(t) = \alpha_0 \delta(t) + \sum_{k=1}^{m} \alpha_k \delta(t - \Delta\tau_k)$$

The transfer function is then

$$G_{\text{ch}}(f) = \alpha_0 + \sum_{k=1}^{m} \alpha_k e^{-j2\pi f \Delta\tau_k}$$

Channel Model: Delay and Coherence

The impulse response and frequency response are

$$g_{\mathrm{ch}}(t) = \alpha_0 \delta(t) + \sum_{k=1}^{m} \alpha_k \delta(t - \Delta\tau_k)$$

$$G_{\mathrm{ch}}(f) = \alpha_0 + \sum_{k=1}^{m} \alpha_k e^{-j2\pi f \Delta\tau_k}$$

We have

- Delay spread: $\tau_d = \Delta\tau_m$
- Coherence bandwidth: $B_c = 1/\tau_d$

As long as the signal bandwidth $B << B_c$ the spectrum will be approximately flat. Something like 1/10 is a good estimate.

Then delays won't cause significant cancellation in the spectrum.

Channel Model Example

The impulse response

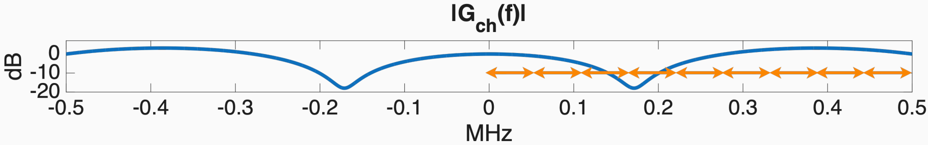
$$g_{\text{ch}}(t) = \delta(t) - 0.5\delta(t - 0.5T_b) + 0.5\delta(t - 2.5T_b)$$

Channel frequency response

$$G_{\text{ch}}(f) = 1 - 0.5e^{-j2\pi \cdot 0.5T_b f} + 0.5e^{-j2\pi \cdot 2.5T_b f}$$

Let $T_b = 1\mu\text{s}$ and $\tau_d = 2.5\mu\text{s}$, then $B_c = 1/\tau_d = 400\ \text{kHz}$

The frequency response then looks like



It changes significantly over any 400 kHz, which will cause distortion

Channel Equalization

One possible approach is to equalize the channel.

Find a filter $g_{eq}(t)$ that undoes the effect of the channel

$$x(t) * g_{ch}(t) * g_{eq}(t) \approx x(t - \tau_{gd})$$

where τ_{gd} is the group delay of the cascaded channel and equalizer.

How do you find $g_{eq}(t)$?

- Use a known preamble in the signal, and deconvolution
- Use blind deconvolution

Madhow Ch 8.1-8.2 covers this.

Limited by the depth of the cancellation, and the noise level. As some point you can't recover.

Multi-Carrier Modulation

We want high data rates

Symbol duration T_s must be much greater than the delays τ_d to avoid multipath cancellation

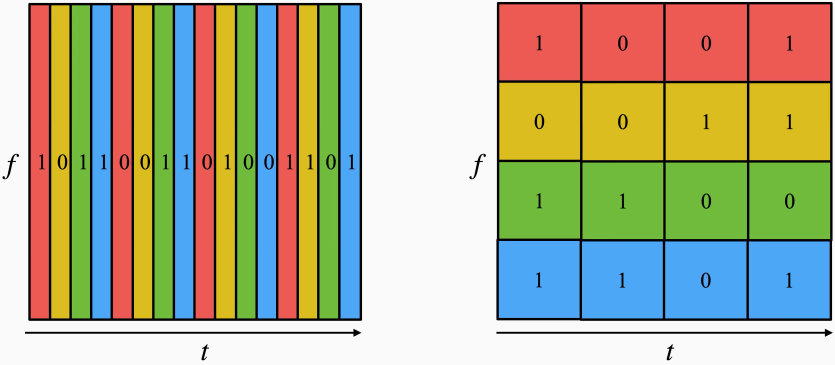
We need wide band channels then, and that also leads to multipath cancelation

Solution

- Divide high speed stream into N lower rate substreams
- Choose $\tilde{T}_s = NT_s \gg \tau_d$
- Transmit each stream on N subcarriers
- Bandwidth of each subcarrier is B/N, so the total bandwidth is preserved

Frequency Tiling

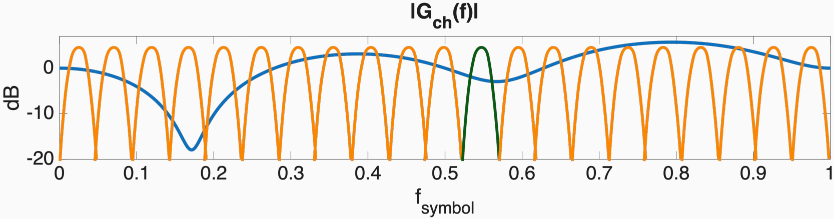
Instead of one wide-bandwidth channel we have four narrower channels



Bits are allocated to all four carriers, so the total data rate is the same.

Each symbol is longer, and on a narrow channel, so multipath is much less of a problem.

Multicarrier Modulation



We'd like the channels to be orthogonal and tightly packed

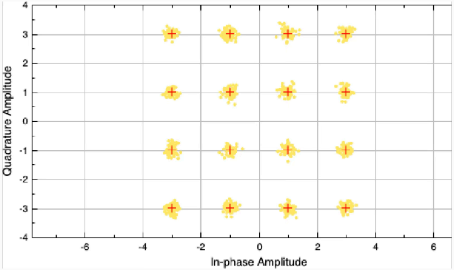
If there is a flat frequency response for each channel ($T_s << \tau_d$) the effect on channel k is

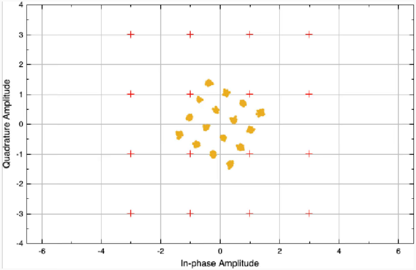
$$y_k(t) = G_{ch}(f_k)x_k(t)$$

This is just a scaler multiplier!

This makes equalization easy.

Equalization Example





16-QAM with

$$y_k(t) = G_{ch}(f_k)x_k(t)$$

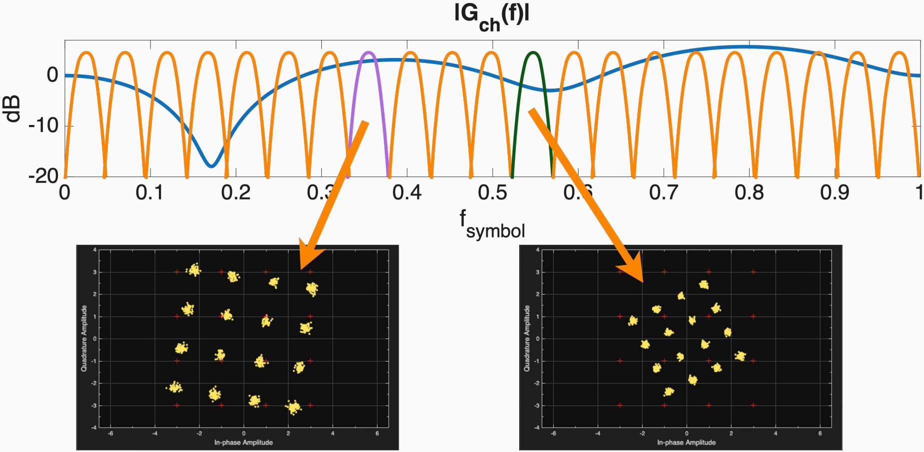
where $G_{ch}(f_k) = \frac{1}{3}e^{j\pi/3}$.

The constellation is rotated by $\pi/3$, and scaled to $1/3$.

Easy to correct.

Multicarrier Modulation

Each subcarrier will have its own constant channel coefficient



Each can be equalized independently

Orthogonal Frequency Domain Multiplexing, OFDM

This multicarier modulated signal is OFDM. How do we do create the OFDM signal in practice?

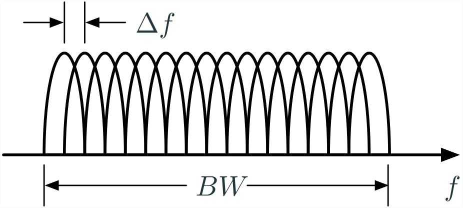
Assign groups of bits $x_k(t)$ to symbols c_k

We take the sequence c_k of QAM encoded symbols and modulate (FSK encode) one sample as

$$y_k(t) = c_k e^{j2\pi k\Delta f t}$$

Each symbol c_k is transmitted at its own FSK frequency $k\Delta f$

Each frequency we can think of as an independent subchannel



OFDM

We can send data on all of them at once by adding the signals up

If we combine all of the samples in an interval (frame) we get

$$y(t) = \sum_{k=0}^{N-1} c_k e^{j2\pi k \Delta f t}$$

If there are N samples, then there are N output FSK frequencies

In practice $y(t)$ is sampled, and this operation is computed with an inverse FFT.

Very effectively uses the entire available spectrum

Different channels will have different noise and propagation properties

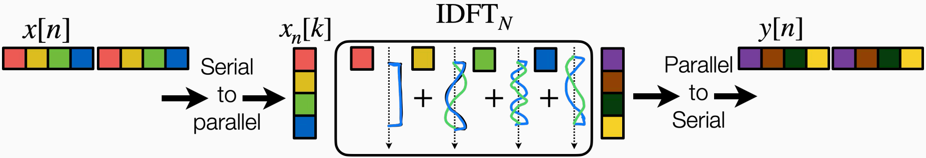
We can equalize each independently, and allocate more or less bits to each

OFDM using an FFT

The sequence of output values are computed with

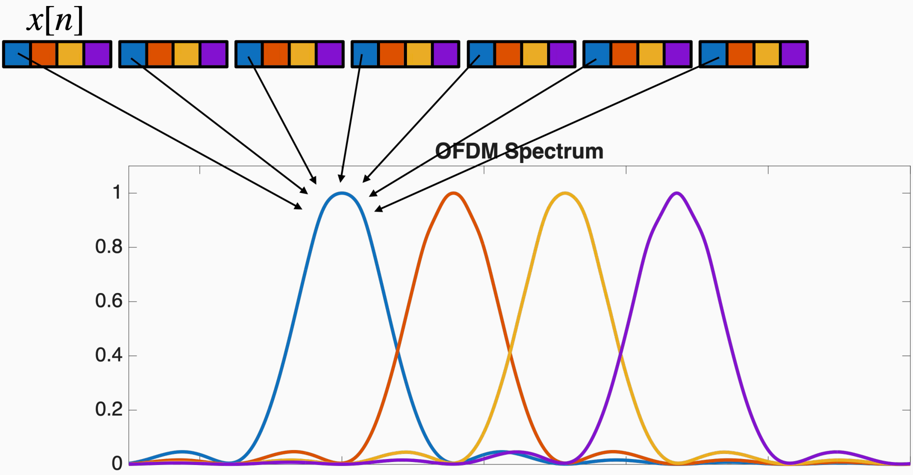
$$y[n] = \sum_{k=0}^{N-1} x_k[n] e^{j2\pi \frac{kn}{N}}$$

For a given N, $y[n] = \mathrm{IDFT_N}\left(x_n[k]\right)$!

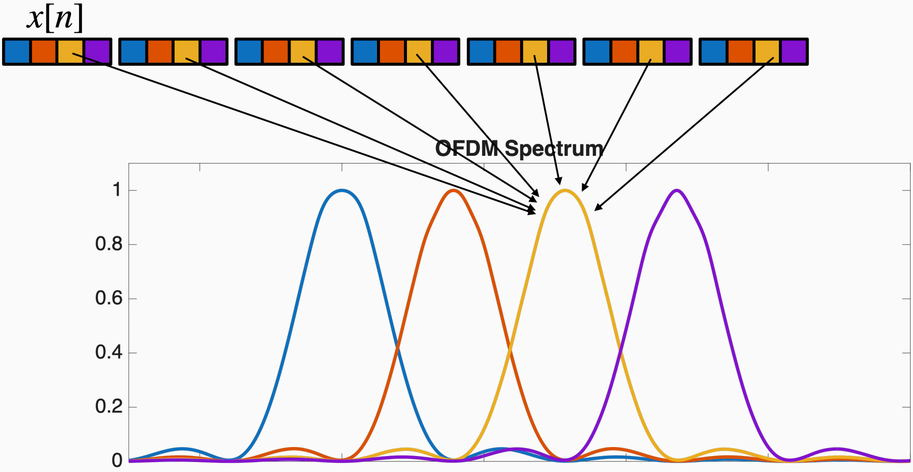


We break the input into N symbol blocks, compute the IDFT, and then serialize the outputs.

Spectrum of OFDM



Spectrum of OFDM

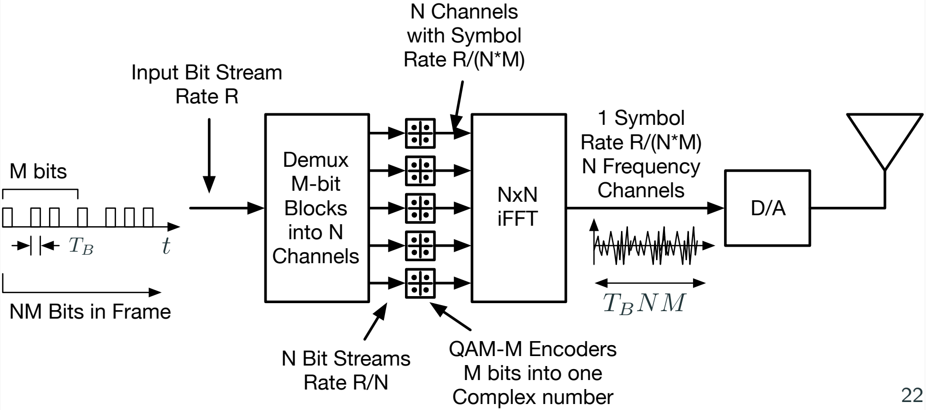


OFDM Encoder

The encoder takes serial bits and allocates them to symbols.

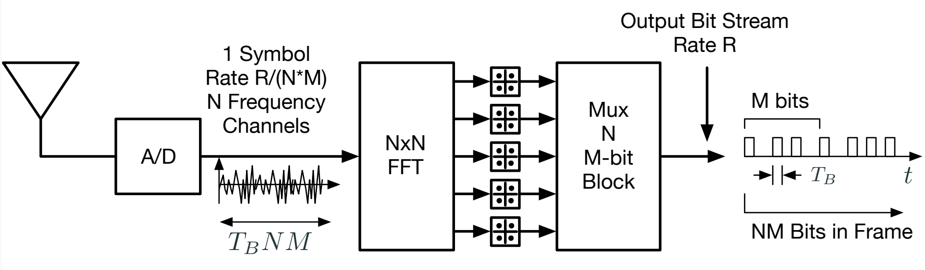
The symbols are the coefficients to an N-point IDFT

The output of the IDFT is then serialized and transmitted



OFDM Decoder

The receiver basically just inverts the operations of the transmitter



Serial bits are allocated to symbols.

These are input to a DFT.

The output symbols are then serialized to decode the original bit stream

Linear vs Cyclic Convolution

One issue is that the channel dispersion can blur one OFDM symbol into the next. This is because it is a linear convolution.

TX:

RX corrupted by channel $*g_{ch}(t)$

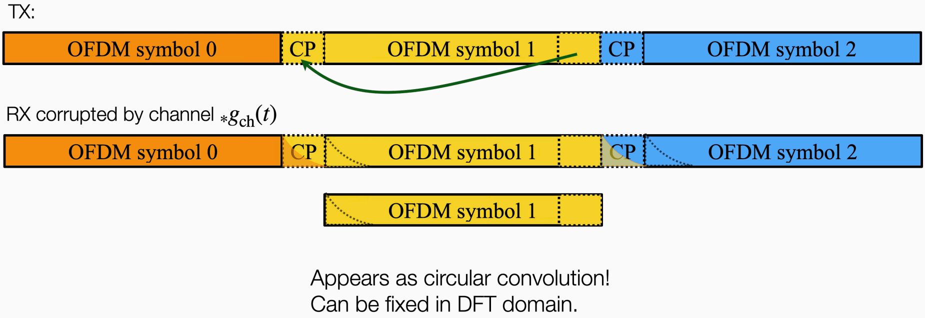
$$X(f)Y(f) \Leftrightarrow x(t) * y(t)$$ Linear convolution

$$DFT(x[n])DFT(y[n]) \Leftrightarrow x[n] \otimes y[n]$$ Circular convolution

Can not correct for corruption from previous symbol using DFT!

Circular Convolution with a Cyclic Prefix

We can fix this by appending a part of the *end* of the symbol to its beginning!

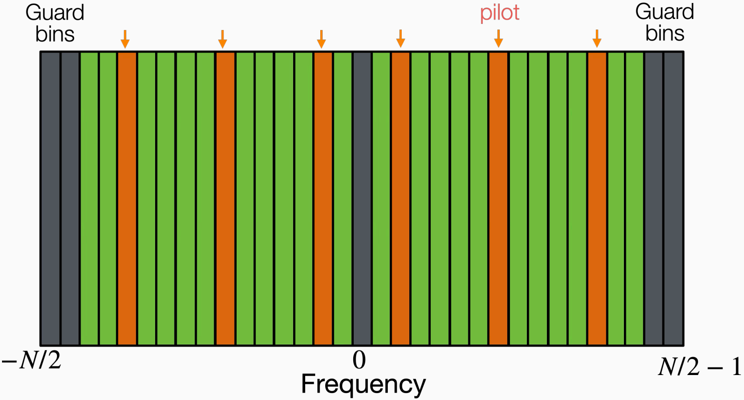


Then the linear convolution from the channel impulse response will blur the correct data into the symbol.

It will be a circular convolution, which can be corrected perfectly.

Frequency Structure

In general, not all of the subcarriers are used for data.



The guard band protects for interference from adjacent signals

The pilot channels are used for estimating the channel for equalization.

WiFI 802.11a/g

Bandwidth = 20MHz

N = 64 subcarriers

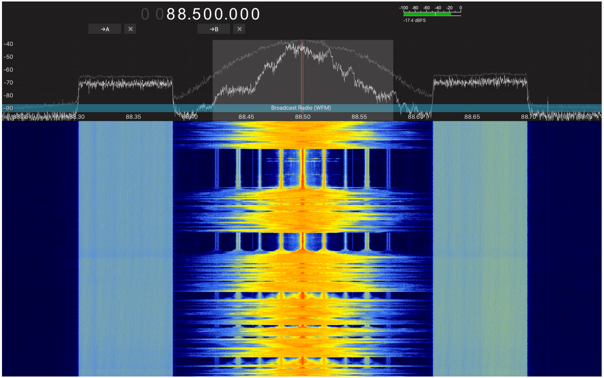
- 48 used for data
- 12 outer unused
- 4 used as pilots

Circular prefix length of 16

Modulations: BPSK to 64QAM

Convolutional Codes 1/2, 2/3, 3/4

HD Radio



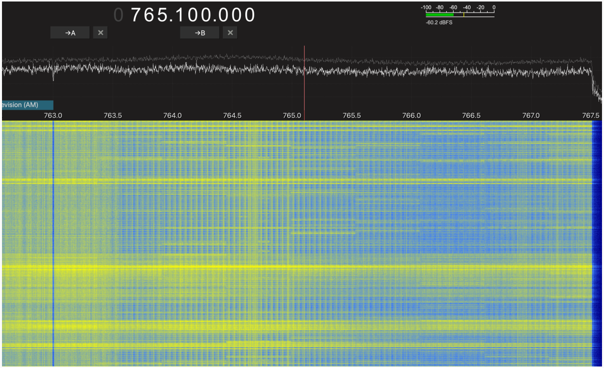
Digital bands from 115 Hz to 200 kHz

About 520 subcarriers allocated to up to three HD channels plus metadata

Symbol length 2.903 ms, with a 0.726 ms cyclic prefix

Symbol Modulation QPSK, 16-QAM, or 64-QAM

LTE Cell Phones



Upper edge of band 14 (758-768 MHz) downlink, AT&T

15 kHz subchannels, 600 subcarriers in 50 resource blocks of 12 subcarriers

Modulation QPSK, 16-QAM, or 64-QAM

Symbol 66 μs with a 5.2 μs circular prefix, 1 ms frames of 14 symbols

OFDM

OFDM has many practical advantages

- The many different frequency channels are resistant to channel variations
- The much lower symbol rate makes timing and pulse shaping much easier

The number of frequencies can be anywhere from 64 to 8k or more

The constellation encoder may be anything from BPSK through QAM-256

That can be a lot of bits per symbol!

OFDM

OFDM is widely used

- Cell Phones: 4G-LTE, 5G
- Cable Modems: DOCSIS 3.1
- Digital TV: DVB-T, ATSC 3.0
- WiFi: 802.11a/g/n and later
- Digital Radio: Digital Radio Modiale (DRM), HD FM

Essentially, any high capacity channel is or will be OFDM

It solves many of problems at the cost of a lot of computation, but that is free now.