Internet of Things: Software Defined Radios

John L. Manferdelli

johnmanferdelli@hotmail.com

April 6, 2020, 14:30

Software defined radio, part 1, radios, spectrum and modulation

- Radio and transmission lines
- Spectrum
- Modulation

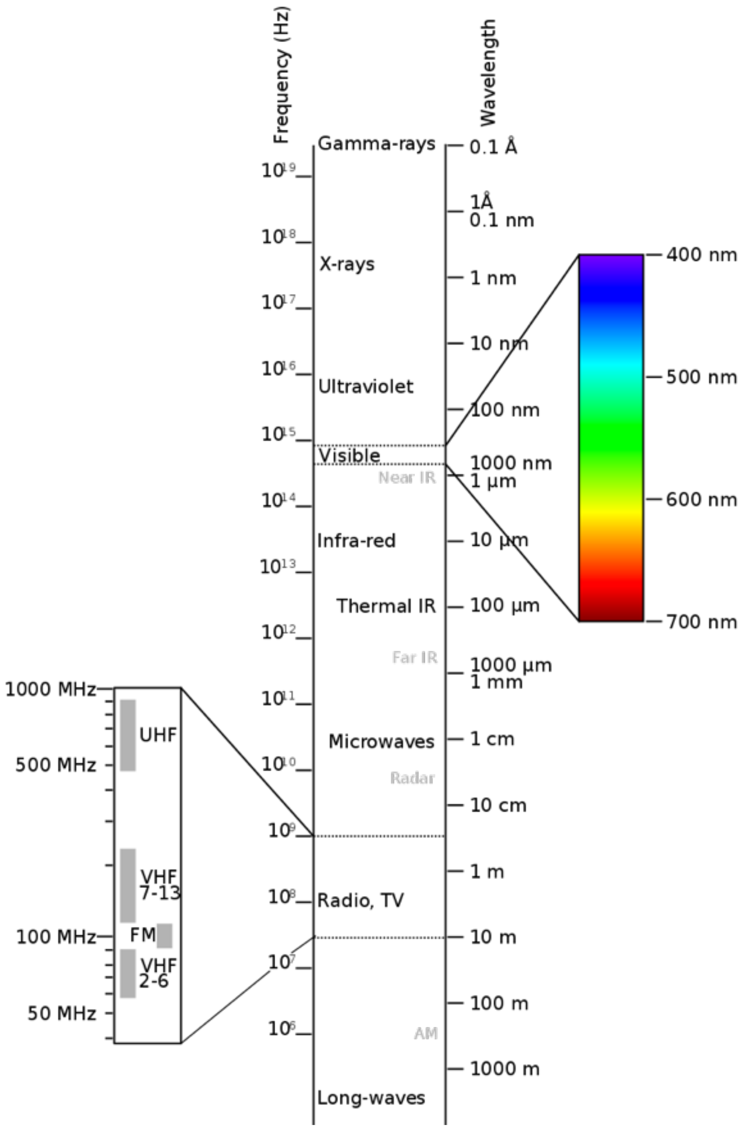
Radio and transmission lines

- Accelerating charges generate travelling waves of electric and magnetic fields in accordance with Maxwell's equations ("EM waves"). These waves travel at the speed of light (in a vacuum) and their intensity drops off linearly rather than on the square of the distance.
- Waves can travel wirelessly and be distributed radially (isotropically) from a source or an-isotropically (using, say, a parabolic antenna) so the majority of the energy travels in preferential directions. These waves can also travel in a waveguide like a coaxial cable or twisted pair.
- Waves that carry information (music, images, digital bit streams) consist of a carrier wave which is constant frequency mixed with or modulated by a signal source that represents information (like a Paul Simon song). We call these waves radio waves.
- Modulated, information bearing waves are produced by a transmitter and transmitted through an antenna then received by another antenna, amplified and demodulated at a receiver. The receiver selects (approximately) the right frequency bearing signal, demodulates it and rejects the rest of the spectrum. It's a miracle.
- Light is also an EM wave and everything we do with "radio frequency" waves has a much higher optical frequency analog.

Spectrum

Figure from Wikipedia

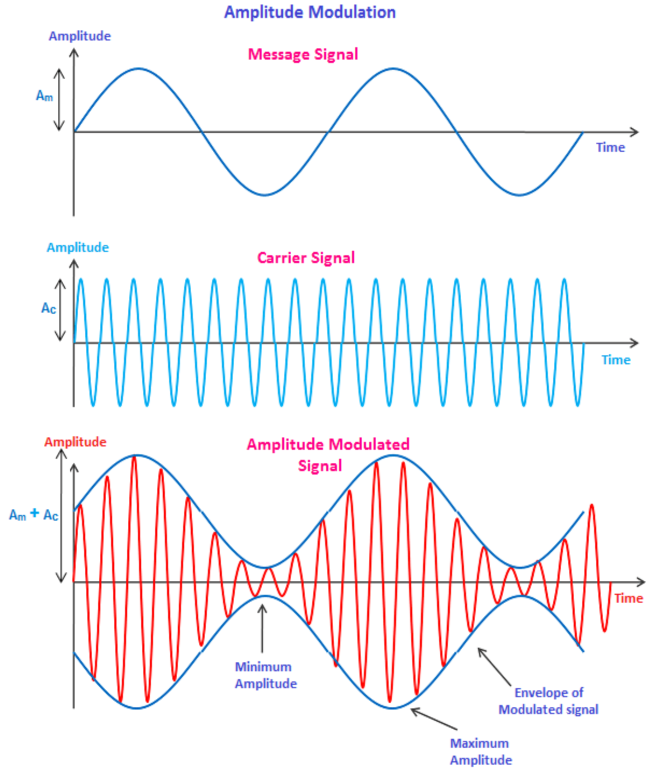
- The spectrum of commonly used EM waves is picture on the right.
- In all cases, f λ= c, the speed of light. ω = 2**π** f. ω is called the angular frequency.
- We will be exploring radio transmission for "AM" of frequency around a megahertz (MHz) and "FM" waves around 100 MHz.
- IoT devices, transmitting at less than .25 watts of power, as well as your wireless router, employ unlicensed spectrum at:
  - 2.4 gigahertz (GHz) – three channel, B, G, and N, and
  - 5GHz, 23 channels, A, N, and AC.
  - 933 MHz for toy controllers and off-net IoT radios
- Transmission characteristics vary with frequency
- A Wireless-A device that runs on at 5 GHz can support a maximum data rate of up to 54 Mbps.
- Routers employ a very efficient and error tolerant modulation scheme called Orthogonal Frequency Division Multiplexing (OFDM).



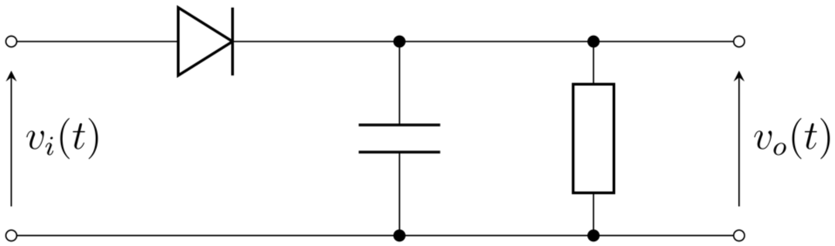
Modulators and demodulators

- A mixer is an electronic circuit that “multiplies” two signals: a carrier at angular frequency $\boldsymbol{\omega}_c$ and an information signal (say a pure tone) at angular frequency $\boldsymbol{\omega}_m$.
- By “multiplying” these, we get a modulated wave:
  - $\cos(\boldsymbol{\omega}_c t) \cos(\boldsymbol{\omega}_m t) = \frac{1}{2} [\cos(\boldsymbol{\omega}_c t + \boldsymbol{\omega}_m t) + \cos(\boldsymbol{\omega}_c t - \boldsymbol{\omega}_m t)]$
- When we have signals composed of multiple frequencies (like a Paul Simon song), they combine linearly so the same equations hold in superposition.
- A demodulator combines a (known) carrier with the received, modulated wave to recover the information signal.
  - $\cos(\boldsymbol{\omega}_c t + \boldsymbol{\omega}_m t) \cos(\boldsymbol{\omega}_c t) = \frac{1}{2} [\cos(2\boldsymbol{\omega}_c t + \boldsymbol{\omega}_m t) + \cos(\boldsymbol{\omega}_m t)]$
  - A filter then rejects the much higher frequency component $\cos(2\boldsymbol{\omega}_c t + \boldsymbol{\omega}_m t)$ to give the original information signal $\cos(\boldsymbol{\omega}_m t)$.
- When signals are received, they are very weak but circuits know as amplifiers increase their power.
- Like I said, it’s a miracle.

AM



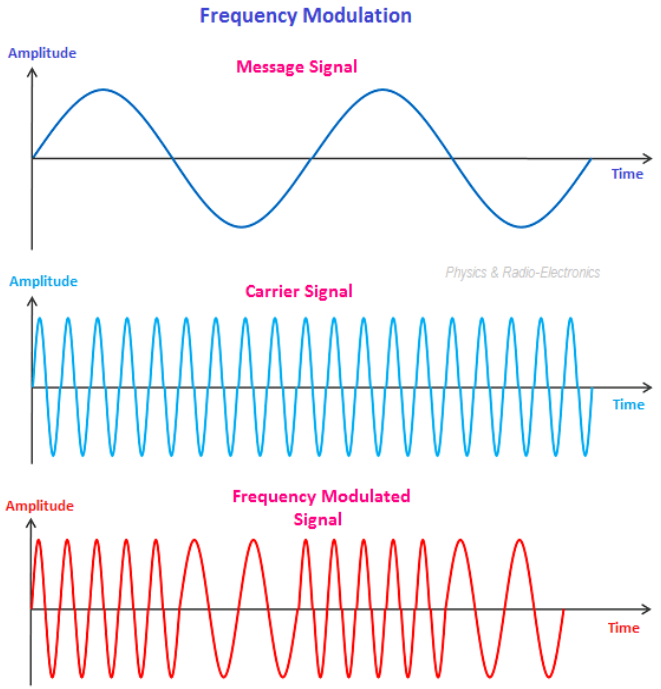
- The simplest modulation scheme is Amplitude modulation (AM).
- Given the carrier wave, cos($\boldsymbol{\omega}_c$t), the AM wave bearing a single frequency information component (a single tone, of frequency, ω_m) is:
    - cos($\boldsymbol{\omega}_c$t) (B cos($\boldsymbol{\omega}_m$t) + 1). The constant term keeps the ”envelope” of the carrier positive. The analytic representation with a scale factor is $M(t) = \frac{B}{2} e^{j\omega_c t}(1 + \ e^{j\omega_m t})$.
    - To demodulate, we simple mix again with $e^{j\omega_c t}$; however notice that in the analytic representation $|M(t)| = \frac{B}{2}(1 + e^{j\omega_m t})$.
    - In the analog world, AM demodulators are fairly simple.



Analog AM Demodulator

Figure from https://www.physics-and-radio-electronics.com/blog/amplitude-modulation/

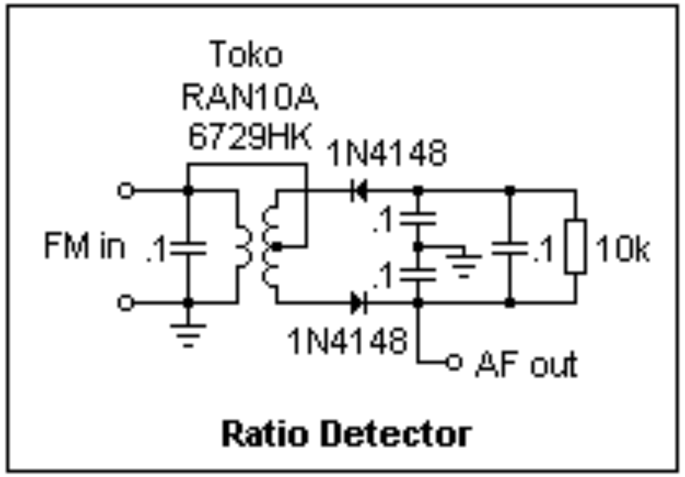
FM



- Our second modulation scheme, Frequency modulation, mixes the carrier with an information signal by varying the frequency (but not the amplitude) of the carrier in a manner related to the signal.
- An FM modulated carrier bearing a single frequency information component has the following equation:
- $M(t) = A_C \cos\big((\omega_c + A_m \cos(\omega_m t))\, t\big)$

Figure from https://www.physics-and-radio-electronics.com/blog/frequency-modulation/

- $\cos(\omega_m)$ recovers the original information signal
- Analog FM demodulators are more complicated than AM demodulators but FM signals have much better noise immunity. They usually work (as in the case of the ratio detector) by converting frequency variations to amplitude variations.



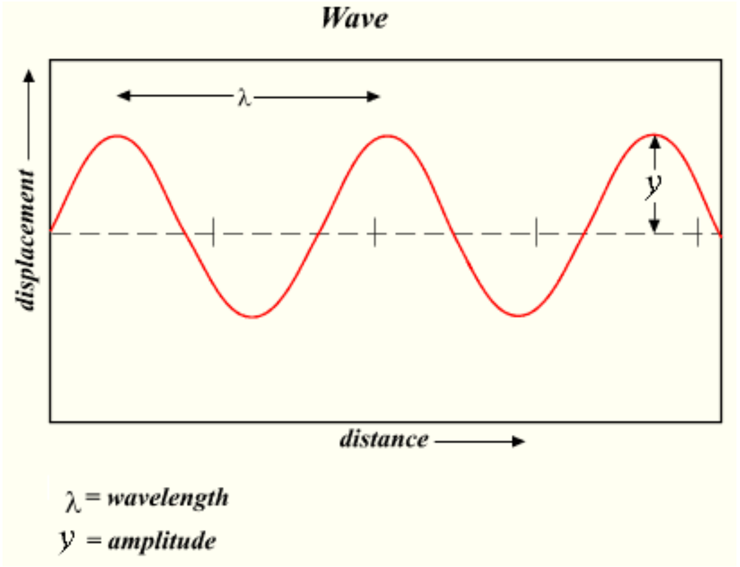
Analog FM Ratio Detector

Software defined radio, part 2, sinusoidal signals and representations

- Oscillatory signals and Fourier analysis
- Analytic signals
- Noise, analytic signals and representing modulation
- Sampling and aliasing
- The frequency domain

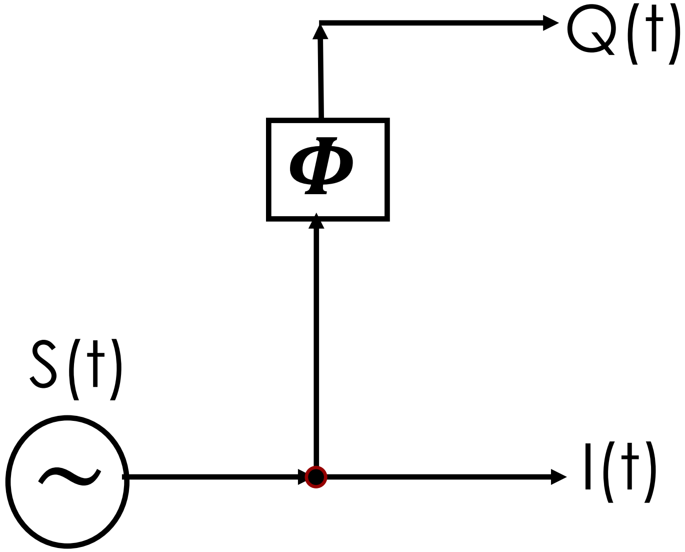
Oscillatory signals and Fourier analysis

- Sinusoidal signals are a natural representative of signals for information transmission.
  - Recall that electromagnetic radiation (the radio waves that carry information) are generated by accelerating charges, caused by potentials with non-zero second derivatives. Sinusoidal signals are always accelerating.
  - Fourier decomposition allows us to decompose any signal into a sum of simple sinusoidal components (in the real domain or the complex domain) of different frequencies. For example, $s(t) = \sum_{m=-\infty}^{\infty} a_m \sin(m\omega t) + b_m \cos(m\omega t)$.
  - We can easily analyze signal processing on a complex signals by determining the behavior on simple sinusoidal functions and combining them.
  - Signals can equally well represented in either the time domain or the frequency domain. For example, the simple function $f(t) = \sin(\alpha t)$ in the time domain is represented as $F(\omega) = \alpha$, in the frequency domain. The (continuous or discrete) Fourier transform converts between these representations.



Analytic signal representation

- We want to split a signal S(t) into two components:
  - The signal itself or the in-phase component, I(t); and,
  - The signal shifted by 90 or $\frac{\pi}{2}$ or out of phase (or quadrature) component, Q(t).
  - We combine these into a single *complex* number x(t)= I(t)+jQ(t)
- It turns out, this is the “natural” representation of a signal for most signal processing and is often the implicitly assumed representation in SDR applications. It is called the analytic representation.
- Almost all modulation schemes are best explained using this representation. Further, it is the natural representation when we apply the Fourier transform (continuous or discrete), which employs complex numbers. It also demystifies the concept of “negative frequencies” in signal processing.
- In the diagram on the right, Φ, represents a circuit that introduces a 90 or $\frac{\pi}{2}$ phase delay.

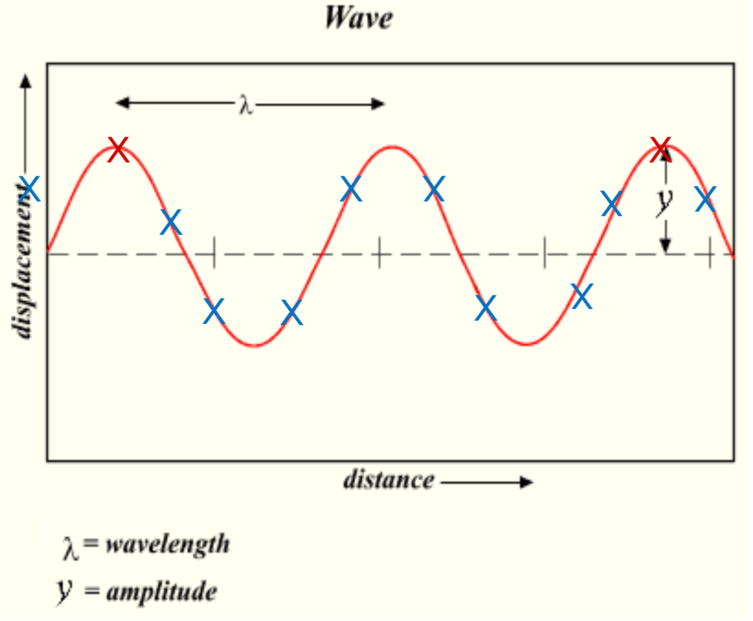


Noise, analytic signals and modulation

- Signals, especially ”wireless” signals are subject to radio frequency noise caused by thermal motion causing random acceleration of charges as well as environmental (and even intentional) noise sources.
- When receiving signals for demodulation, a critical figure of merit is the “signal to noise ratio” measured in dB. The definition is SN= 10 log(S/N) where S and N are respectively the power in some relevant frequency band of the signal and the ambient noise.
- AM modulation and demodulation using the analytic representation.
  - The analytic representation with a scale factor for AM modulation is $M(t) = \frac{B}{2}e^{j\omega_c t}(1 + e^{j\omega_m t})$.
  - To demodulate, we simple mix again with $e^{j\omega_c t}$; however notice that in the analytic representation, $|M(t)| = \frac{B}{2}(1 + e^{j\omega_m t})$.
- FM modulation and demodulation using the analytic representation.
  - $M(t) = A_C e^{j(\omega_c + A_m \cos(\omega_m t))t}$
  - $\angle M(t) = \arctan(\frac{sin(A_m \cos(\omega_m))}{\cos(A_m \cos(\omega_m))})$ = $A_m \cos(\omega_m)$ recovers the original information signal.

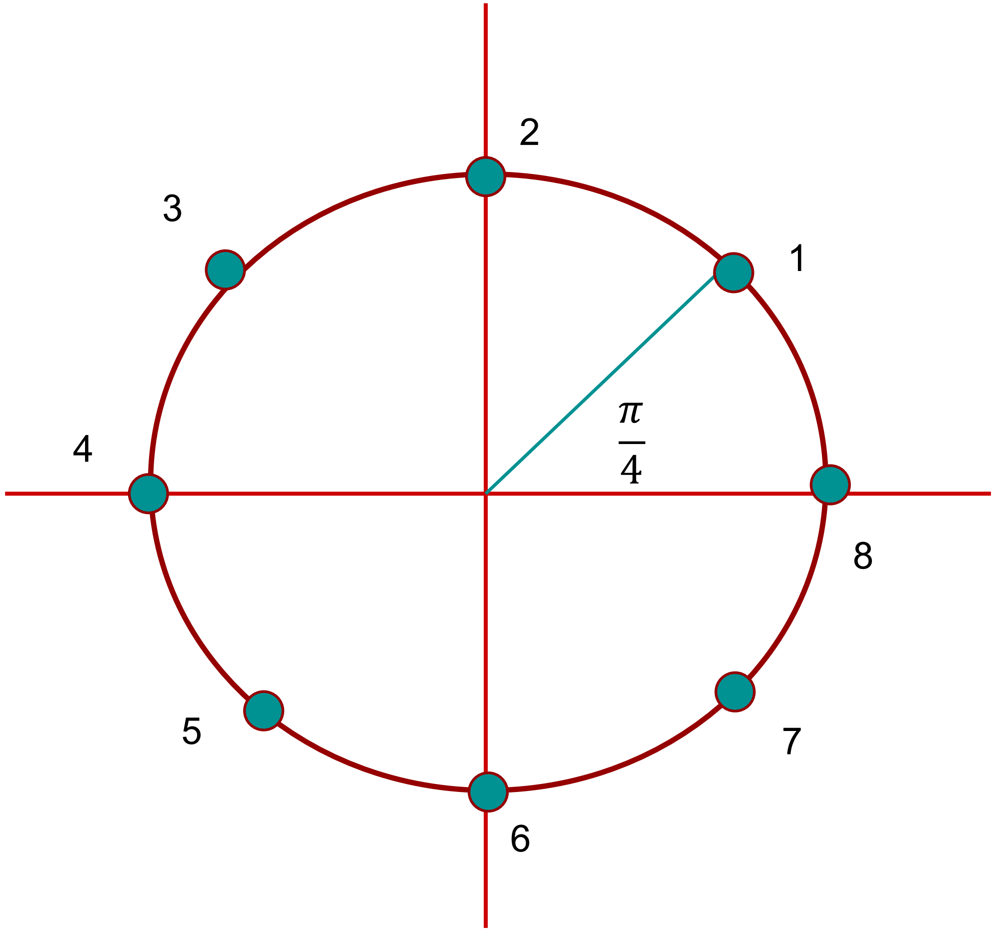
Sampling and aliasing

- Digital signal processing requires converting analog signals into digital samples (ideally, using the analytic signal representation) so we can use a computer to process them.
- If we do not sample an analog signal frequently enough, we will get a distorted sense of the analog signal from the digital samples. For example, if we sample the sine wave on the right every two wavelengths (or at frequency $\frac{c}{2\lambda}$), as indicated by the red x's, we get the values 1, 1, 1 …. which tells us very little about the original function. This is called aliasing, the two red samples are *aliases* of the same values in two different cycles.
- If we sample every quarter wavelength (or at frequency $\frac{4c}{\lambda}$), as indicated by the blue x's, we get an accurate sense of the frequency of the wave.
- It turns out, if we sample (in the analytic signal representation) at least as frequently as twice the frequency of simple waves or at least twice the frequency of the highest frequency component of a complex wave, we can exactly reconstruct the analog signal. This is called Nyquist's theorem.
- With modern computers we can sample fast enough and process those sample "in real time" making "digital signal processing" practical.

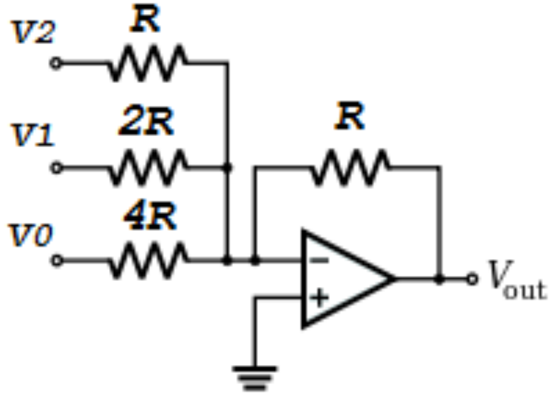


Negative frequency

- If we look at an analytic signal, and watch the numbered successive observations in the time Δt as is pictured on the right, there are two explanations. The first is that the angular velocity, ω, is positive and moves $\frac{\pi}{4}$ each time period Δt. Another explanation is that the angular velocity is negative and moves $-\frac{7\pi}{4}$ each time period Δt.
- This shows that negative frequencies are just as natural as positive frequencies in the analytic representation.



Signal reconstruction



Simple D/A converter
Credit: Electronics course

- The idea of digital sampling is fairly intuitive. It is important, we discovered, before sampling to first low pass filter the incoming signal to avoid aliasing. The components that are used to sample analog signals are called A/D converters. They are fairly common parts that vary with respect to the sampling interval. To digitize high bandwidth signals, A/D converters should be very fast and ideally sample with high precision (16 to 24 bits).
- Commercial A/D converter: ADC32RF45.
- Reconstructing an analog signal from digital samples is the job of a D/A converter. They are slightly more complicated but fortunately, there are standard parts to do that as well.
- Commercial D/A converter: D/A.

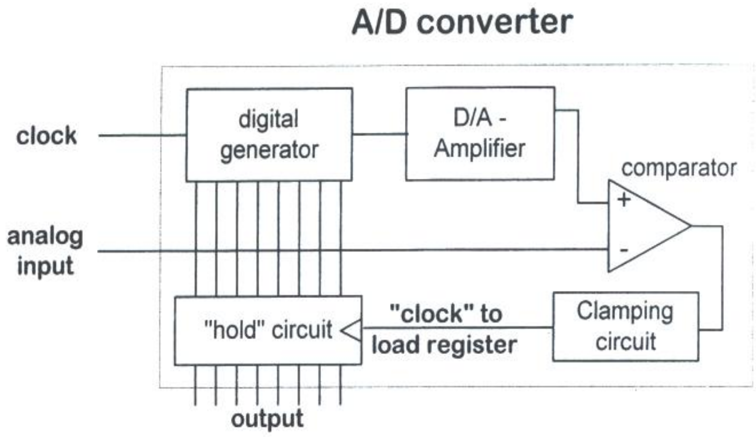
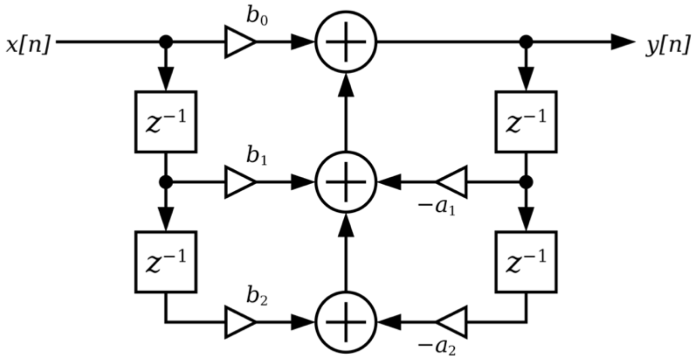


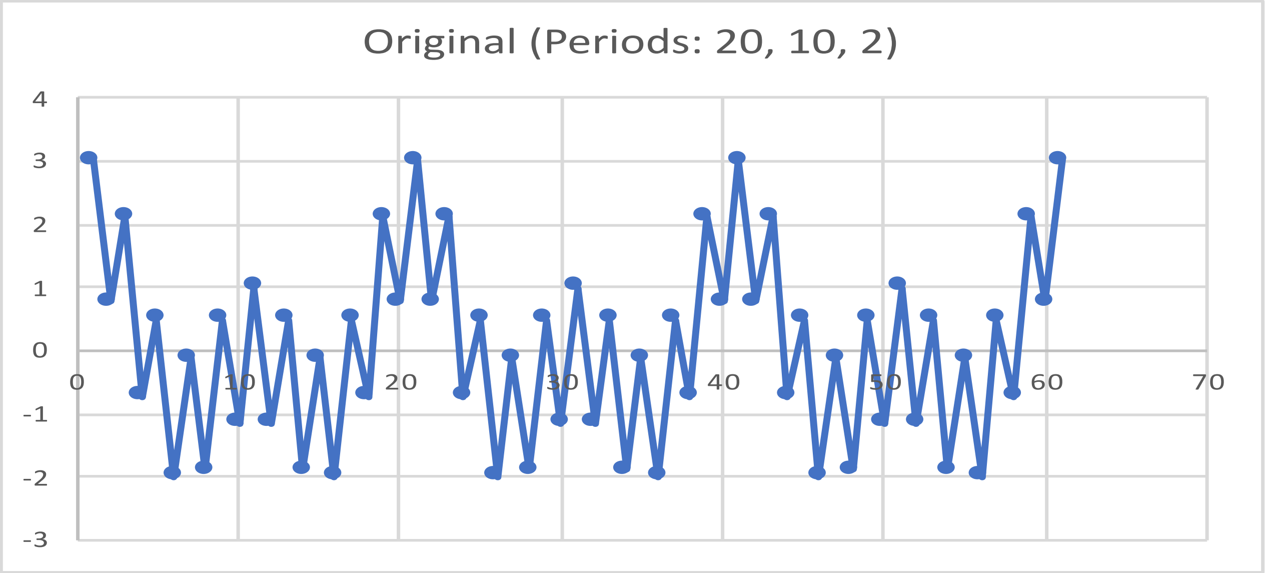
Figure 18.3 The A/D circuit layout.

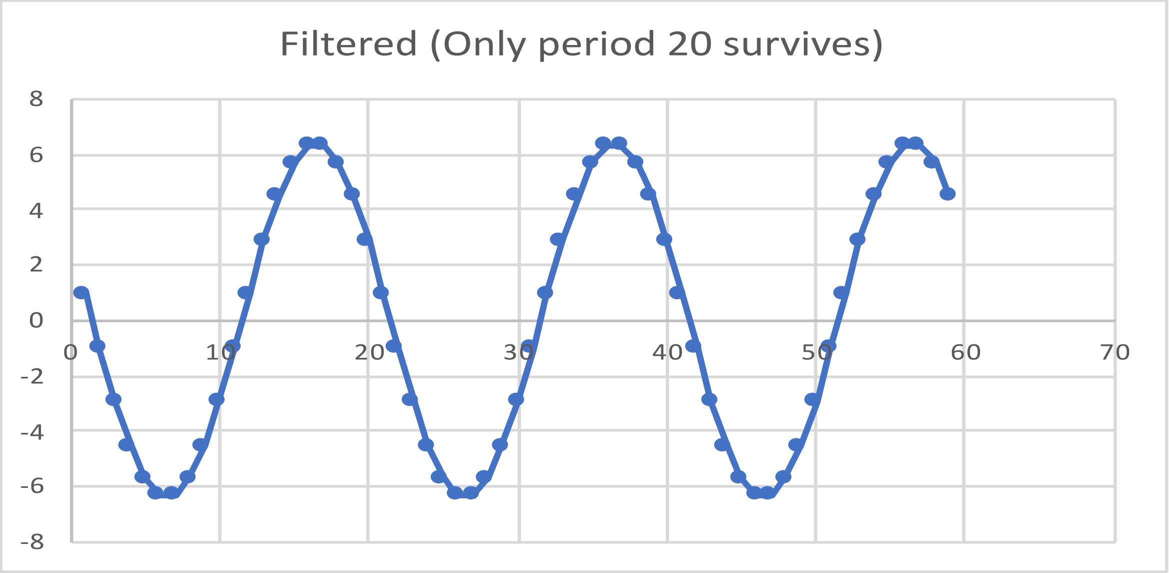
ece.umd.edu

Digital Filters



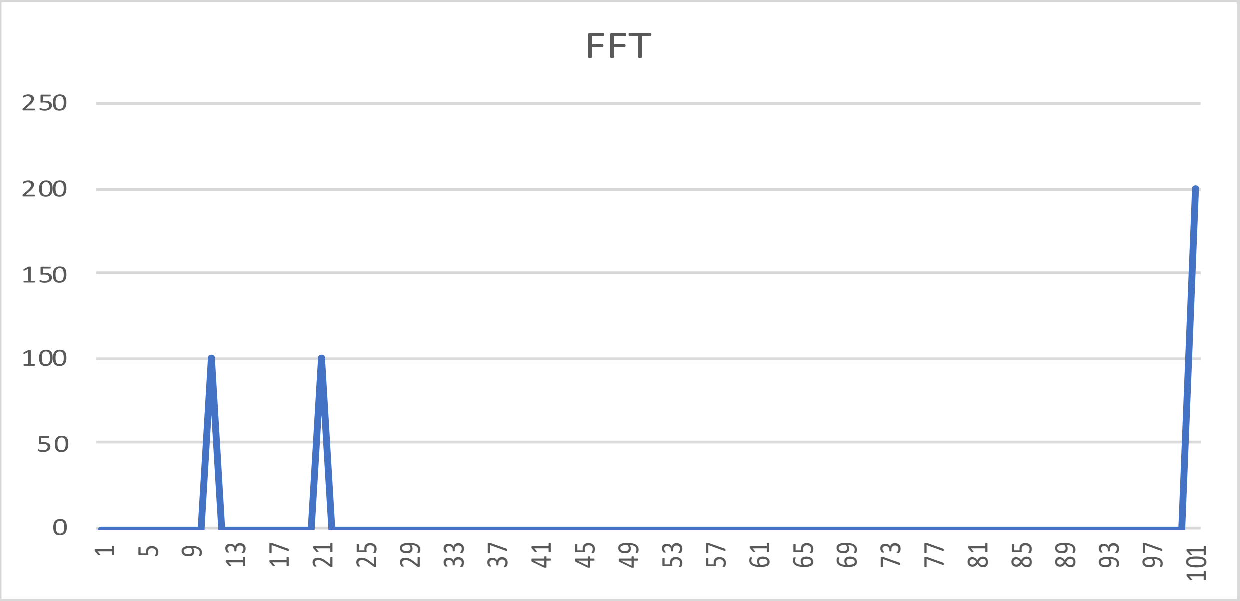
- A digital filter is just a linear sum of sequential samples with coefficients: $y_n = \sum_{k=0}^{N} a_k x_{n-k}$.
- In the example below, the original is the sum of 3 periods (20, 10, 2), the filter is the sum of 10 successive samples. Only the component of period 20 survives, so it is a low pass filter.
- Now, how would you build a high-pass filter?





Fourier Transforms and the frequency domain

- Fourier Transform: $F(w) = \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{\infty} f(t)e^{-2\pi i\omega t}dt$
- Inverse Fourier Transform: $f(t) = \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{\infty} F(\omega)e^{2\pi i\omega t}d\omega$
- Discrete Fourier Transform: $X_k = \sum_{n=0}^{N-1} x_n e^{\frac{-j2\pi kn}{N}}$
- Inverse Discrete Fourier Transform: $x_n = \frac{1}{N}\sum_{k=0}^{N-1} X_k e^{\frac{j2\pi kn}{N}}$



Graph of the real part of the original data set from previous page. Note peaks corresponding to the periods 20, 10, 2.

Working in the frequency domain

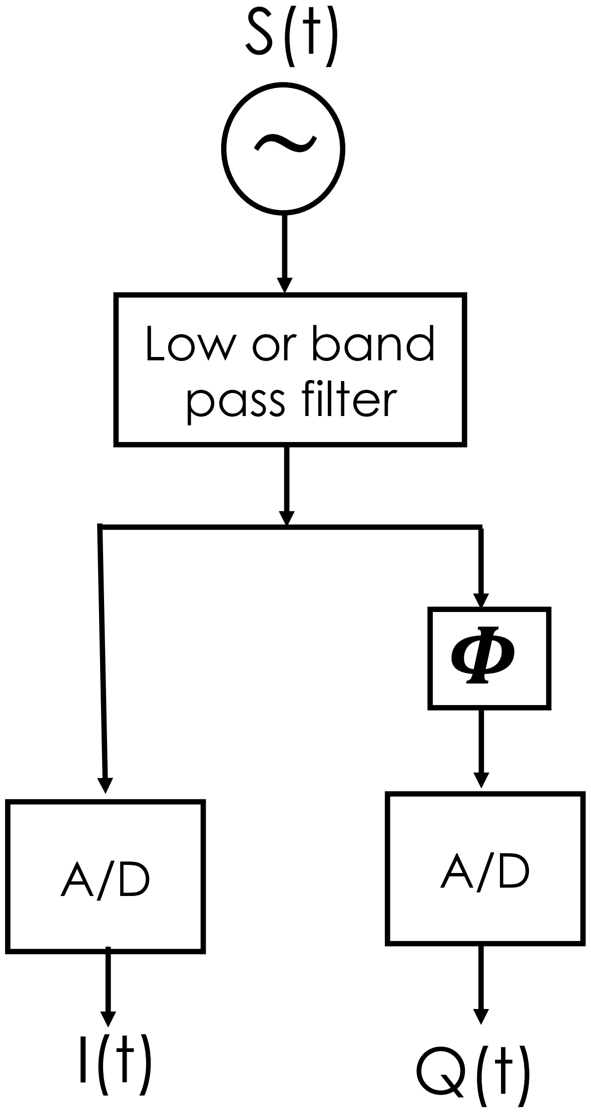
- It is often easier to do some signal processing in the frequency domain. This is easily accomplished with digital signals, by performing a DFT (Discrete Fourier Transform). There are algorithms that do this efficiently.
- Once we are in the digital domain some things that are difficult or impossible in the time domain can be done easily. For example, we can accurately choose the frequency band of a signal we are interested in by simply discarding frequency components outside the range of interest.
- Complex processing like spread spectrum transmissions, frequency hopping and adaptive processing would not be practical without frequency domain processing.
- The inverse discrete Fourier transform can be used after all frequency domain processing is completed to convert back into the time domain.

Software defined radio, part 3, what's an SDR and how do you use it?

- Components of a software defined radio
  - Analog front end
  - Digital sampling
- Signal processing
- Building blocks
- GNU radio
- AM, FM modulation and demodulation in a SDR
- FSK
- BSK

What’s an SDR?

- An SDR is a signal processing subsystem end that takes, as input, a continuous analog signal and converts it into digital samples that can be processed.
- An SDR also does the inverse, taking a time series of digital samples, resulting from some processing and turning those samples into the desired analog output.
- As we discussed in previous slides, we need to do this with a clear understanding of aliasing so we can get accurate signal reconstruction.
- An SDR can be used with a digital computer to perform the signal processing. The entire function of an SDR/processor combination can be changed entirely by changing software. Digital samples are easy to process with “perfect accuracy” and complete flexibility, unlike most analog circuits.
- The picture on the right shows the components of half an SDR system.

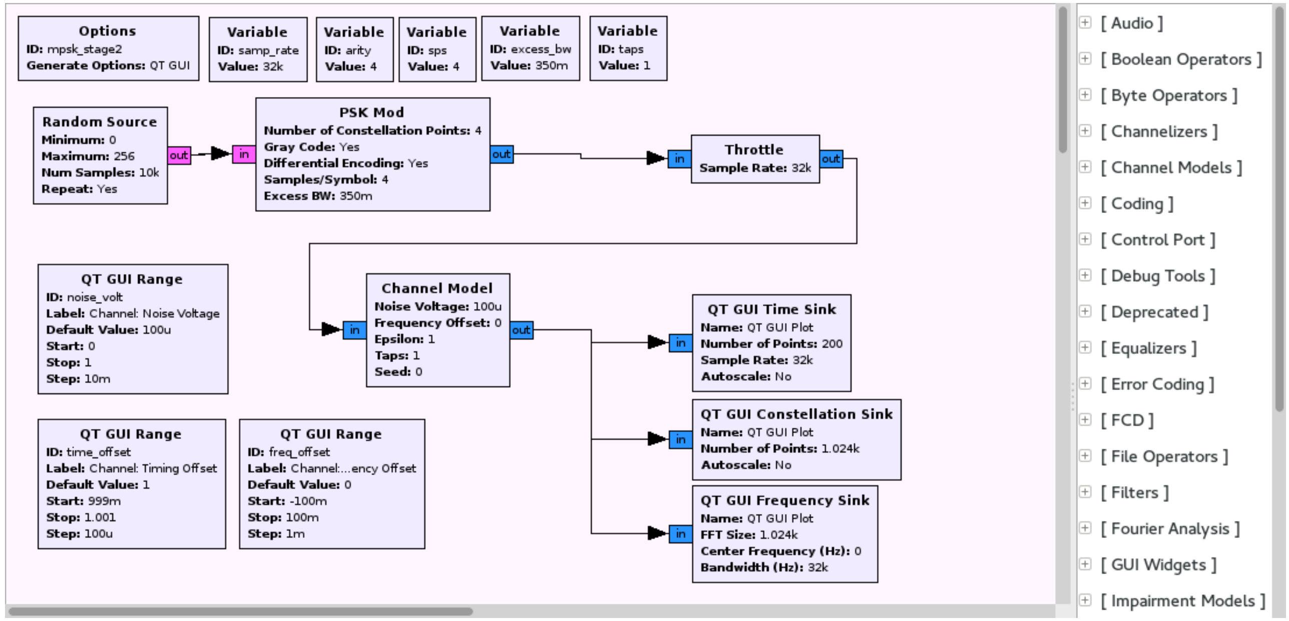


Building blocks

- SDR processing uses program elements, often represented graphically, in blocks.
- Common blocks include:
    - Oscillators
    - Multipliers
    - Adders
    - Filters
    - FFT/IFFT blocks
    - Modulators and demodulators
    - Blocks representing functions that sample or synthesize analog signals (A/D and D/A converters).
    - Virtual sources and sinks for simulating signals.

Gnu Radio

- A very flexible and widely used tool to design and build these processing elements is GNU radio.
- The GNU Radio Companion (GRC) is a graphical UI used to develop GNU Radio applications consisting of python code.
- GNU Radio companion has an extensive library of components and works with SDR's like HackRF One and USRP.
- All our examples use GNU Radio Companion.



Options
ID: mpsk_stage2
Generate Options: QT GUI

Variable
ID: samp_rate
Value: 32k

Variable
ID: arity
Value: 4

Variable
ID: sps
Value: 4

Variable
ID: excess_bw
Value: 350m

Variable
ID: taps
Value: 1

Random Source
Minimum: 0
Maximum: 256
Num Samples: 10k
Repeat: Yes

PSK Mod
Number of Constellation Points: 4
Gray Code: Yes
Differential Encoding: Yes
Samples/Symbol: 4
Excess BW: 350m

Throttle
Sample Rate: 32k

QT GUI Range
ID: noise_volt
Label: Channel: Noise Voltage
Default Value: 100u
Start: 0
Stop: 1
Step: 10m

Channel Model
Noise Voltage: 100u
Frequency Offset: 0
Epsilon: 1
Taps: 1
Seed: 0

QT GUI Time Sink
Name: QT GUI Plot
Number of Points: 200
Sample Rate: 32k
Autoscale: No

QT GUI Constellation Sink
Name: QT GUI Plot
Number of Points: 1.024k
Autoscale: No

QT GUI Range
ID: time_offset
Label: Channel: Timing Offset
Default Value: 1
Start: 999m
Stop: 1.001
Step: 100u

QT GUI Range
ID: freq_offset
Label: Channel:...ency Offset
Default Value: 0
Start: -100m
Stop: 100m
Step: 1m

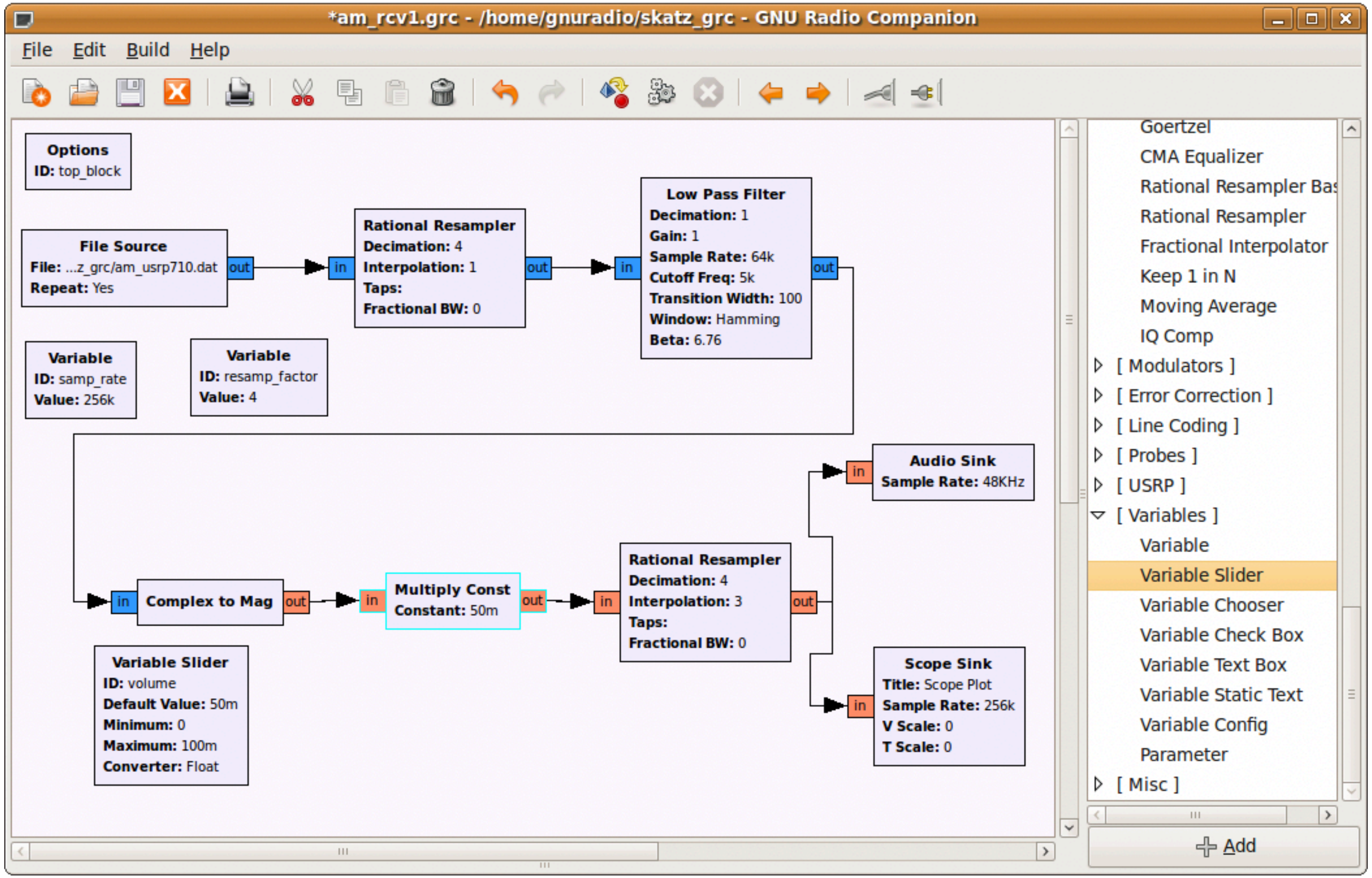
QT GUI Frequency Sink
Name: QT GUI Plot
FFT Size: 1.024k
Center Frequency (Hz): 0
Bandwidth (Hz): 32k

- GRC UI on the right.
- The "panel" on the right is list of categories of components that come with GRC like demodulators, adders, signal sources and sinks, SDR interfaces, FFT's

From Wikipedia

AM Demodulation in an SDR

- Analog AM modulation is accomplished by simple mixing. Digital demodulation mirrors this. The demodulator on the right uses our observations earlier that taking the magnitude of the analytic representation of the AM signal decodes it.



Options
ID: top_block

File Source
File: ...z_grc/am_usrp710.dat
Repeat: Yes

Rational Resampler
Decimation: 4
Interpolation: 1
Taps:
Fractional BW: 0

Low Pass Filter
Decimation: 1
Gain: 1
Sample Rate: 64k
Cutoff Freq: 5k
Transition Width: 100
Window: Hamming
Beta: 6.76

Variable
ID: samp_rate
Value: 256k

Variable
ID: resamp_factor
Value: 4

Audio Sink
Sample Rate: 48KHz

Complex to Mag

Multiply Const
Constant: 50m

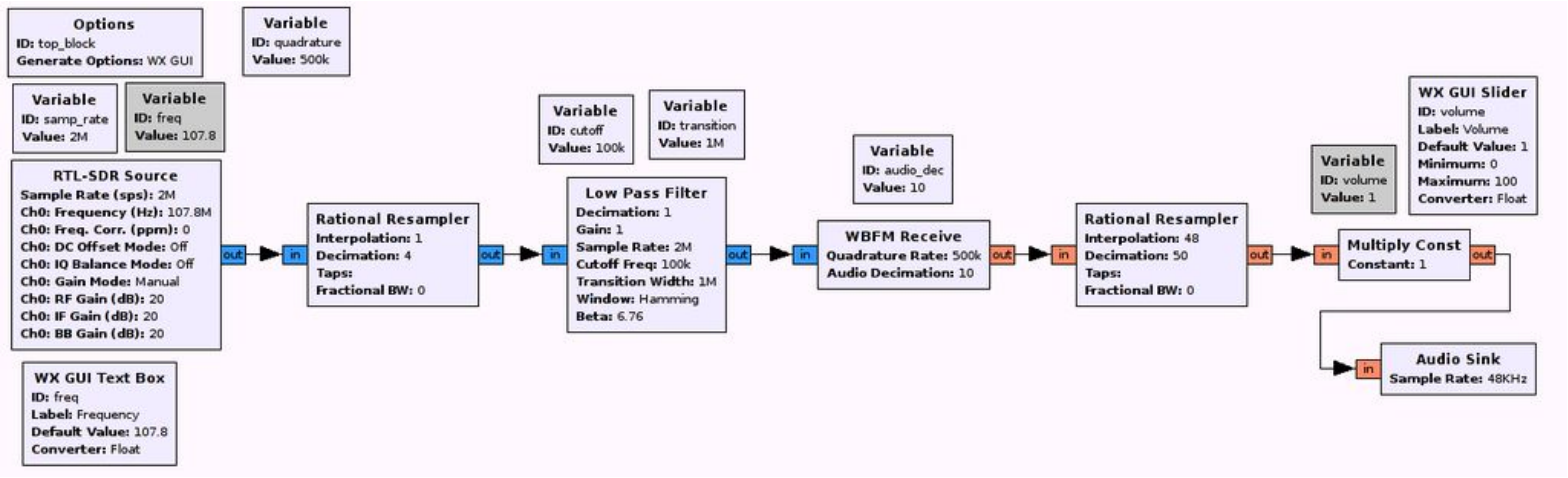
Rational Resampler
Decimation: 4
Interpolation: 3
Taps:
Fractional BW: 0

Scope Sink
Title: Scope Plot
Sample Rate: 256k
V Scale: 0
T Scale: 0

Variable Slider
ID: volume
Default Value: 50m
Minimum: 0
Maximum: 100m
Converter: Float

FM Demodulation in an SDR

- Analog demodulation of FM usually proceeds by turning the FM signal into an AM signal and demodulating that in the usual way. Digital demodulation is a little different.



Options
ID: top_block
Generate Options: WX GUI

Variable
ID: quadrature
Value: 500k

Variable
ID: samp_rate
Value: 2M

Variable
ID: freq
Value: 107.8

RTL-SDR Source
Sample Rate (sps): 2M
Ch0: Frequency (Hz): 107.8M
Ch0: Freq. Corr. (ppm): 0
Ch0: DC Offset Mode: Off
Ch0: IQ Balance Mode: Off
Ch0: Gain Mode: Manual
Ch0: RF Gain (dB): 20
Ch0: IF Gain (dB): 20
Ch0: BB Gain (dB): 20

WX GUI Text Box
ID: freq
Label: Frequency
Default Value: 107.8
Converter: Float

Rational Resampler
Interpolation: 1
Decimation: 4
Taps:
Fractional BW: 0

Variable
ID: cutoff
Value: 100k

Variable
ID: transition
Value: 1M

Low Pass Filter
Decimation: 1
Gain: 1
Sample Rate: 2M
Cutoff Freq: 100k
Transition Width: 1M
Window: Hamming
Beta: 6.76

Variable
ID: audio_dec
Value: 10

WBFM Receive
Quadrature Rate: 500k
Audio Decimation: 10

Rational Resampler
Interpolation: 48
Decimation: 50
Taps:
Fractional BW: 0

Variable
ID: volume
Value: 1

WX GUI Slider
ID: volume
Label: Volume
Default Value: 1
Minimum: 0
Maximum: 100
Converter: Float

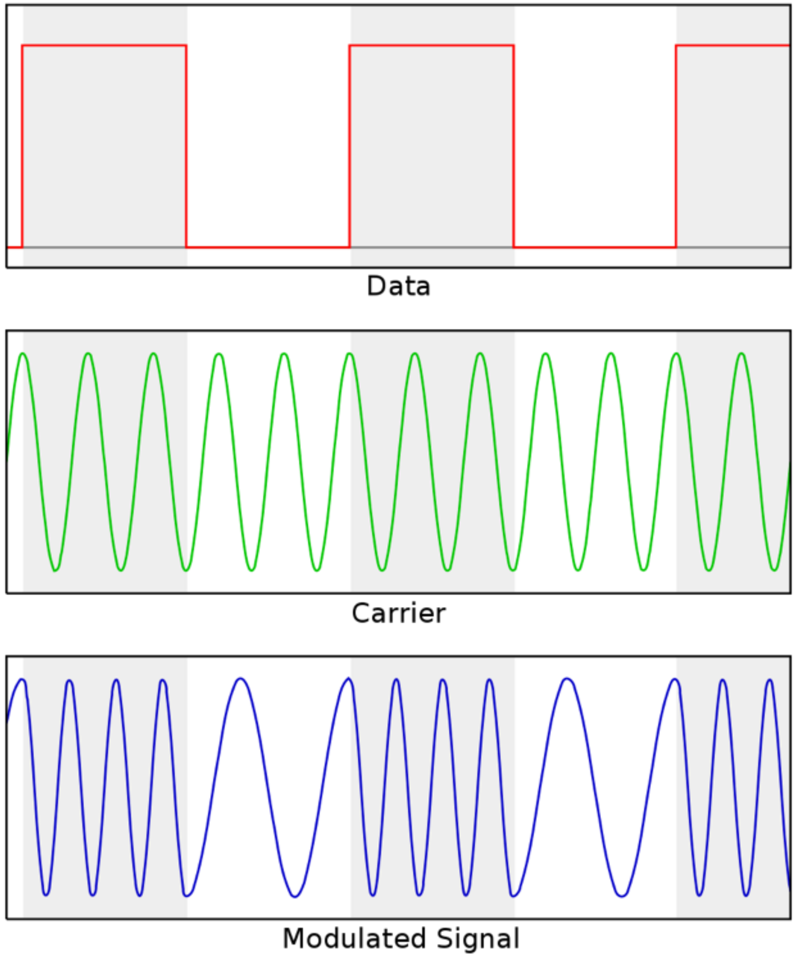
Multiply Const
Constant: 1

Audio Sink
Sample Rate: 48KHz

RTL-SDR.com

FSK Modulation

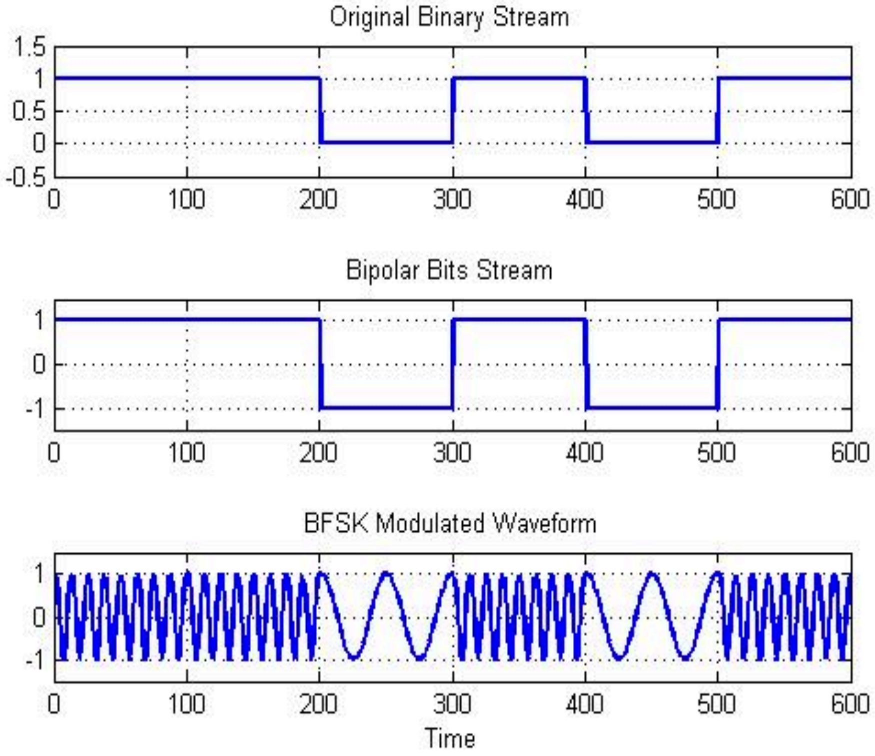
- Frequency Shift Keying is a digital modulation technique.
- 0 and 1 are encoded by different frequencies each in sequential fixed time periods



Wikipedia

BFSK modulation

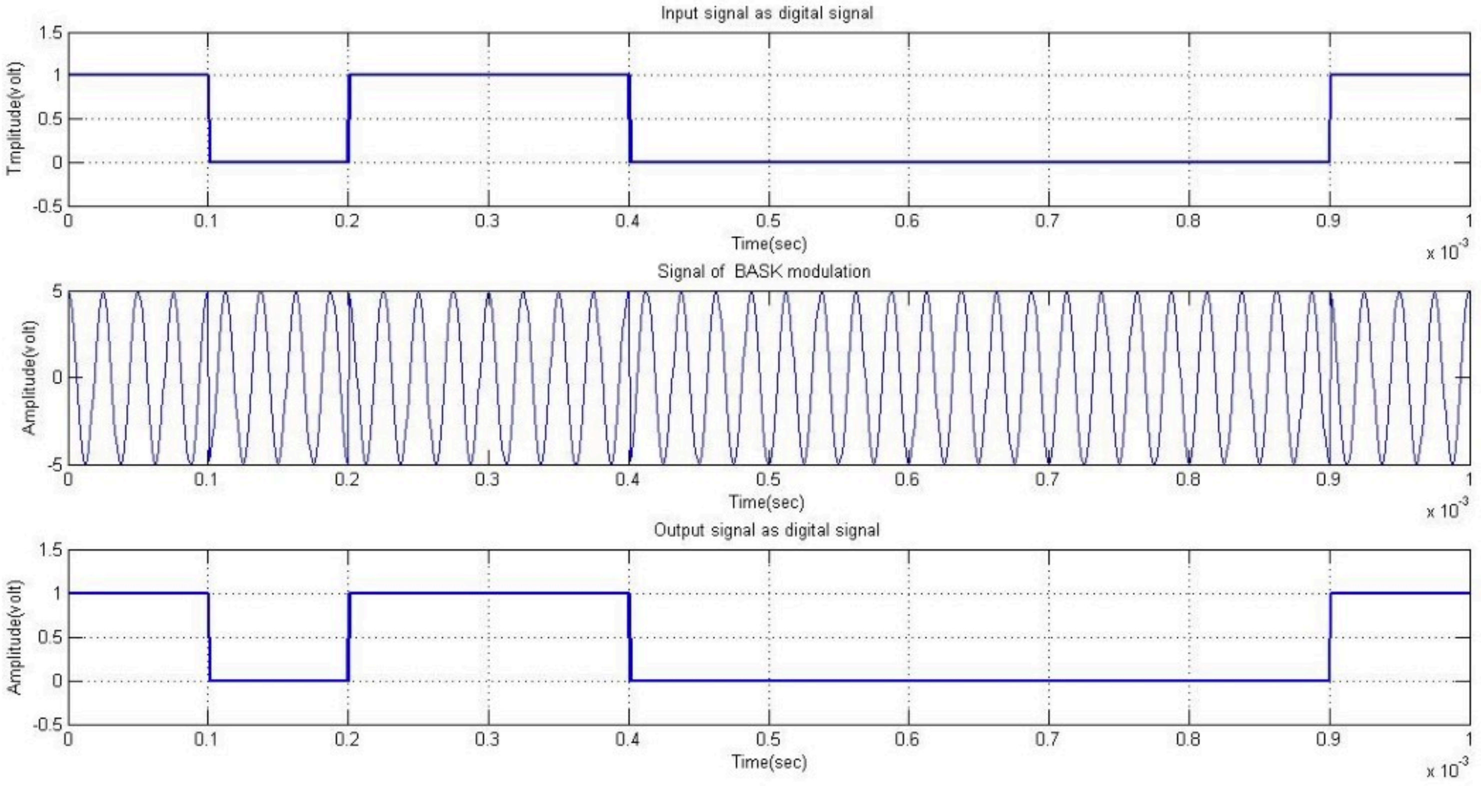
- Binary Frequency Shift Keying (BFSK) modulates digital signals where 0 and 1 are represented by different frequencies.



Mathworks

BPSK modulation

- Binary Phase Shift Keying (BSPK) modulates digital signals where 0 and 1 are represented by different phase states.



Mathworks

QPSK Modulation

- Quadrature Phase Shift Keying (QPSK): Similar to BPSK but there are four subcarrier phases all with the same amplitude. There are 2 bits per symbol.
- If we can choose amplitudes we can get more bits per symbol. This is Quadrature Amplitude Modulation (QAM).

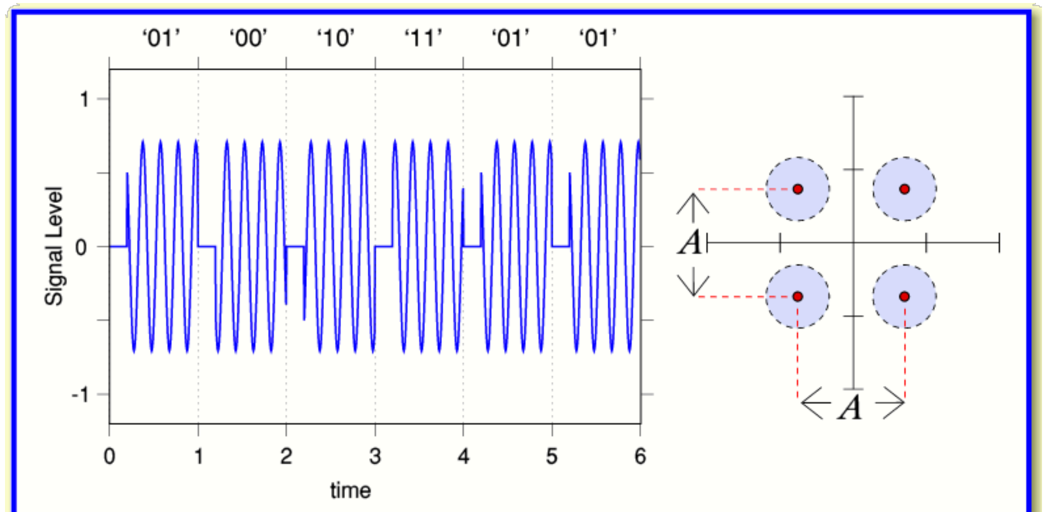


Fig 19.6 Quadrature Phase Shift Keying

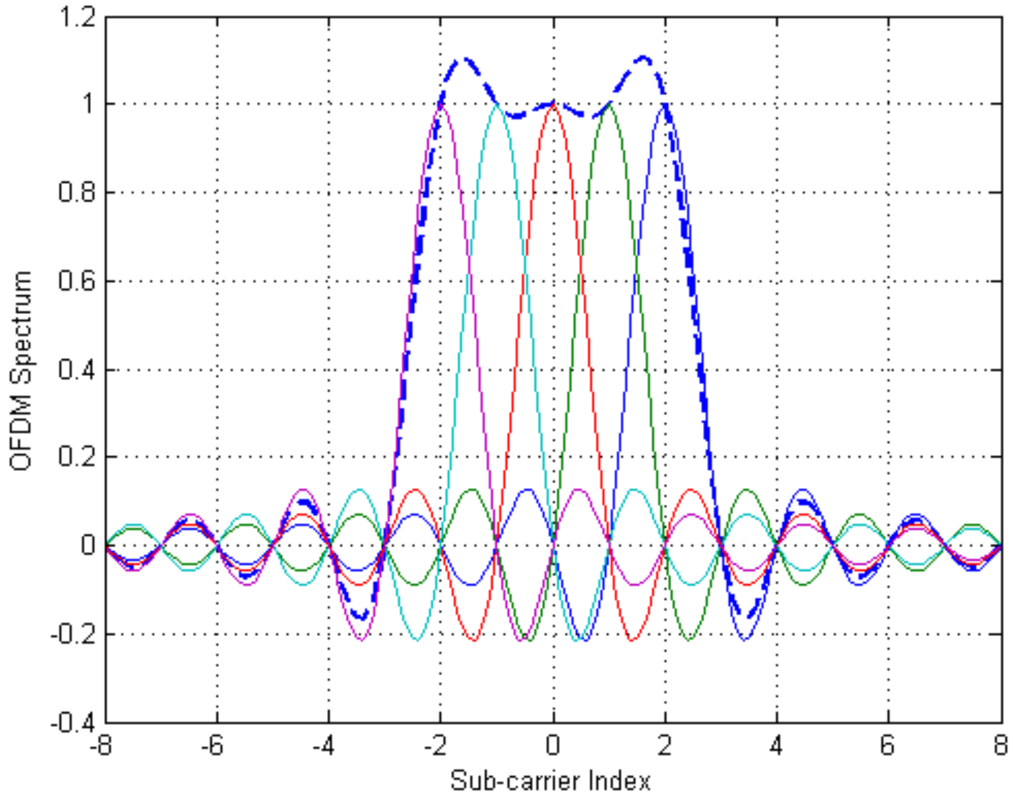
From https://www.st-andrews.ac.uk/~www_pa/Scots_Guide/RadCom/part19/page2.html

Software defined radio, part 4, receiving, transmitting and decoding wireless router signals with SDR

- OFDM
- 802.11

OFDM Modulation

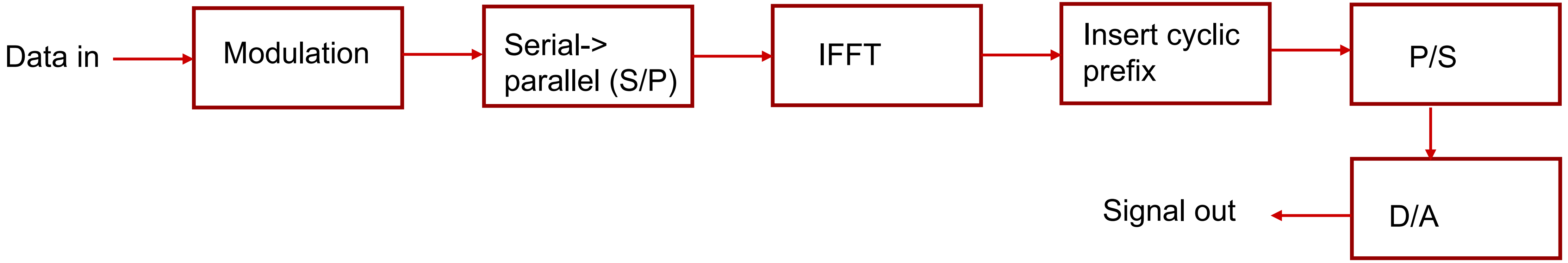
- Orthogonal Frequency Division Modulation encodes sequential digital bits on different carriers.
- It is used in 802.11 and has both high bandwidth and good noise immunity.
- Subcarriers are chosen to be orthogonal to each other (as in the analytic signal representation) and requires accurate frequency synchronization between transmitters and receivers.
- It is significantly more complex than any of the other modulation techniques we've already seen.



Why OFDM?

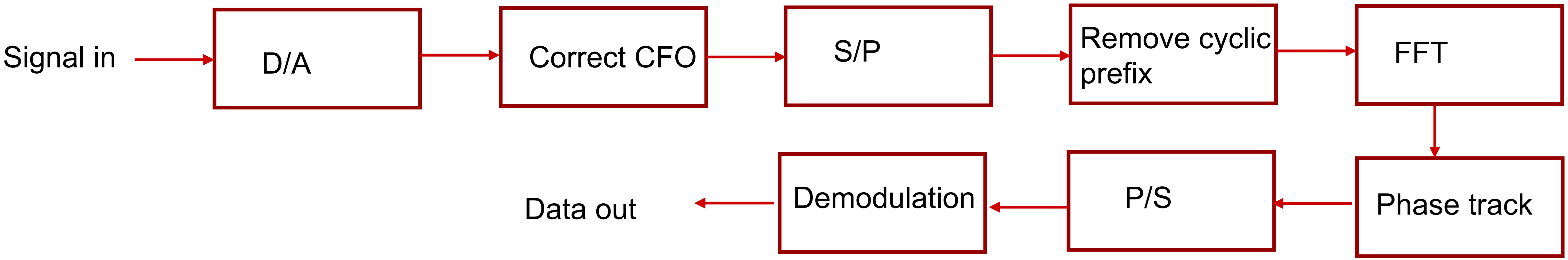
- Wireless signals can suffer from a number of distortions including selective fading (caused by absorption), frequency contention and multi-path distortion caused by the arrival of the same signal from two different paths.
- ODFM is resistant to these distortions and, because it converts the serial data stream into several parallel streams, it is fast.
- Unfortunately, we won’t go into the design details but you can check it out.

OFDM transmitter elements



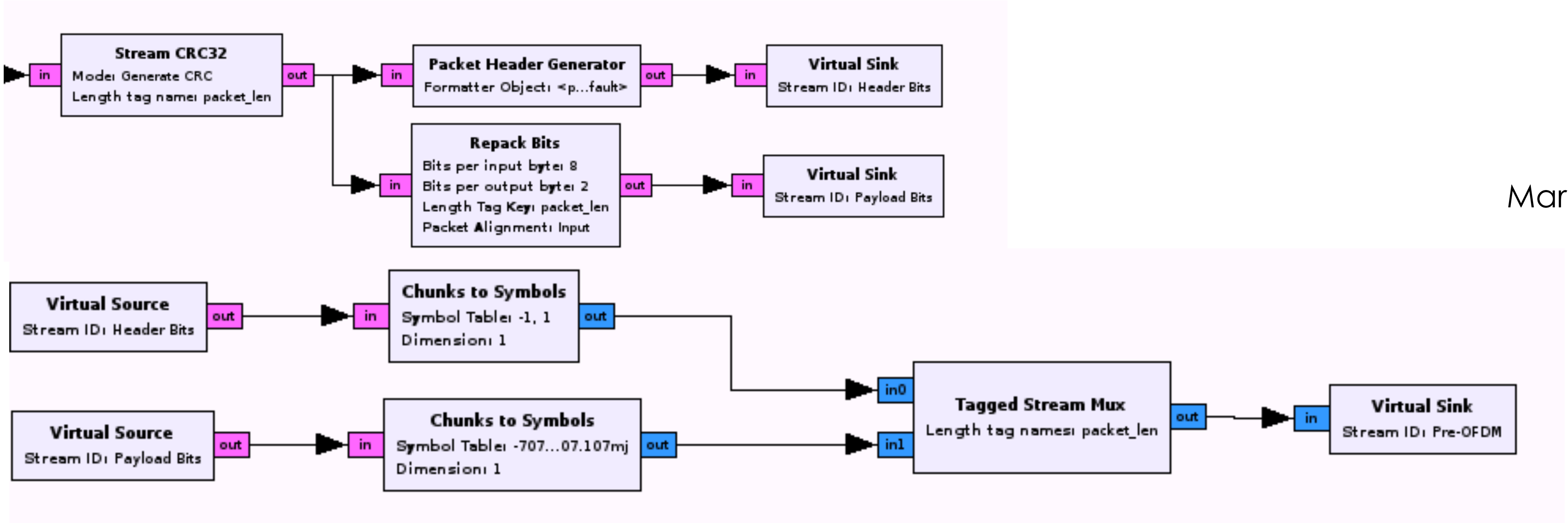
- Code prefix add immunity to crosstalk on different channels. End of symbol is often repeated in front of symbol to protect from latency
- Bits converted into symbols (constellation patterns) in $\mathbb{C}$ then CP added after each symbol
- For packet based transmission several preamble symbols are added to help with packet detection

OFDM – receiver elements



- Receiver needs to correct for carrier drift and timing differences.
- $f_{CFO} = f_c - f_{LO}$

OFDM prefix codes and symbol mapping



Stream CRC32
Mode: Generate CRC
Length tag name: packet_len

Packet Header Generator
Formatter Object: <p...fault>

Virtual Sink
Stream ID: Header Bits

Repack Bits
Bits per input byte: 8
Bits per output byte: 2
Length Tag Key: packet_len
Packet Alignment: Input

Virtual Sink
Stream ID: Payload Bits

Virtual Source
Stream ID: Header Bits

Chunks to Symbols
Symbol Table: -1, 1
Dimension: 1

Virtual Source
Stream ID: Payload Bits

Chunks to Symbols
Symbol Table: -707...07.107mj
Dimension: 1

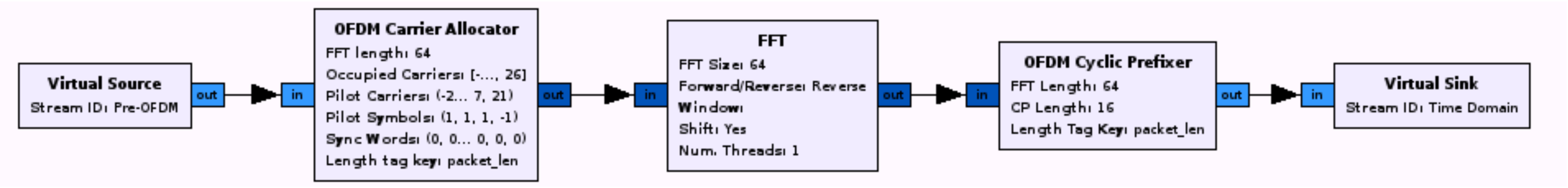
Tagged Stream Mux
Length tag names: packet_len

Virtual Sink
Stream ID: Pre-OFDM

Martin Braun

- Generate cyclic prefix
- Map bits to constellation symbols

OFDM prefix codes



Virtual Source
Stream ID: Pre-OFDM

OFDM Carrier Allocator
FFT length: 64
Occupied Carriers: [-..., 26]
Pilot Carriers: (-2... 7, 21)
Pilot Symbols: (1, 1, 1, -1)
Sync Words: (0, 0... 0, 0, 0)
Length tag key: packet_len

FFT
FFT Size: 64
Forward/Reverse: Reverse
Window:
Shift: Yes
Num. Threads: 1

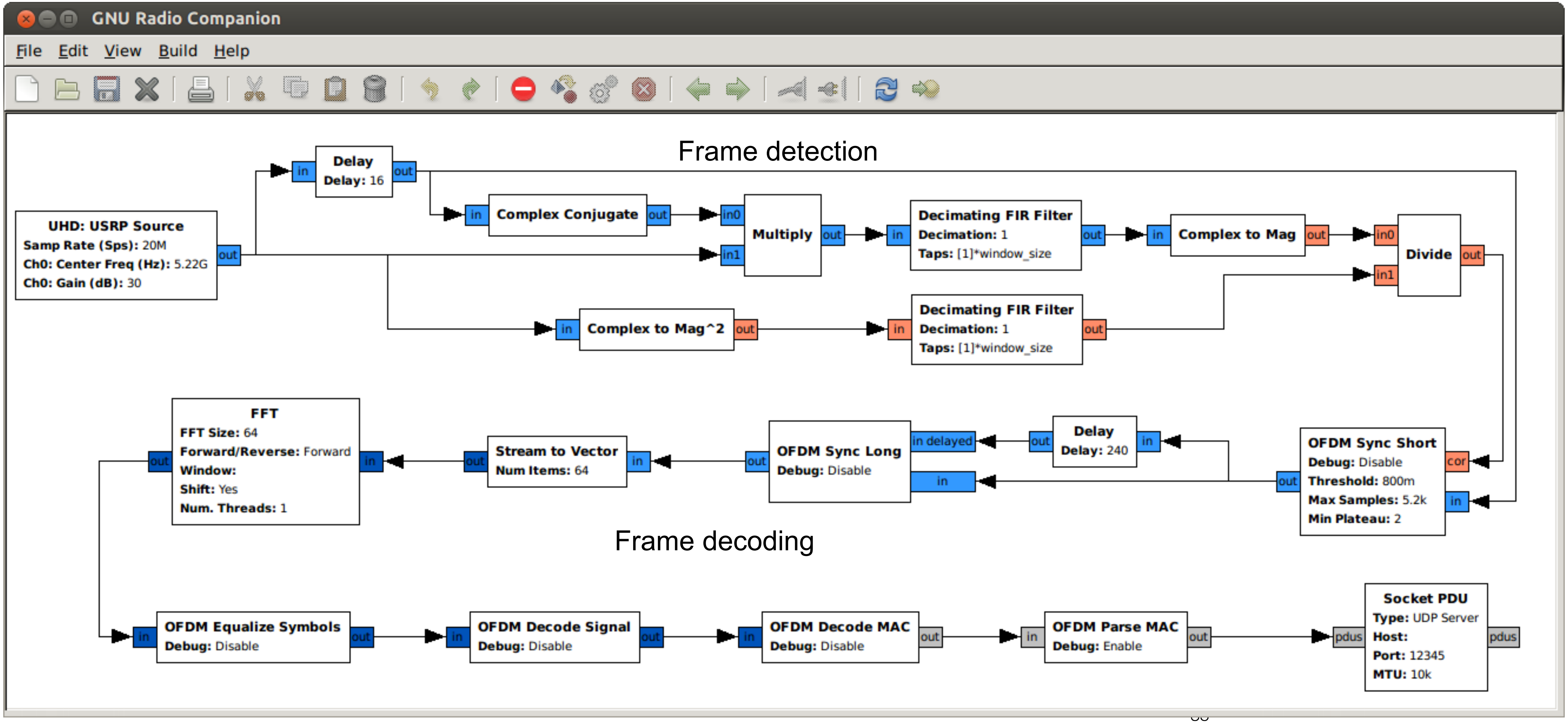
OFDM Cyclic Prefixer
FFT Length: 64
CP Length: 16
Length Tag Key: packet_len

Virtual Sink
Stream ID: Time Domain

Gnu Radio

- Allocate Carriers and add prefixes

Gnu Radio OFDM Receiver



Frame detection

UHD: USRP Source
Samp Rate (Sps): 20M
Ch0: Center Freq (Hz): 5.22G
Ch0: Gain (dB): 30

Delay
Delay: 16

Complex Conjugate

Multiply

Decimating FIR Filter
Decimation: 1
Taps: [1]*window_size

Complex to Mag

Divide

Complex to Mag^2

Decimating FIR Filter
Decimation: 1
Taps: [1]*window_size

OFDM Sync Short
Debug: Disable
Threshold: 800m
Max Samples: 5.2k
Min Plateau: 2

Delay
Delay: 240

OFDM Sync Long
Debug: Disable

Stream to Vector
Num Items: 64

FFT
FFT Size: 64
Forward/Reverse: Forward
Window:
Shift: Yes
Num. Threads: 1

Frame decoding

OFDM Equalize Symbols
Debug: Disable

OFDM Decode Signal
Debug: Disable

OFDM Decode MAC
Debug: Disable

OFDM Parse MAC
Debug: Enable

Socket PDU
Type: UDP Server
Host:
Port: 12345
MTU: 10k

Notes on Gnu Radio OFDM receiver

- Frame detection
  - Detect start of a OFDM frame
  - Calculate autocorrelation: $a[n] = \sum_{k=0}^{N_{win}-1} s[n+k]s^*[n+k+16]$
  - Power $p[n] = \sum_{k=0}^{N_{win}-1} s[n+k]s^*[n+k]$
  - $c[n] = \frac{|a[n]|}{p[n]}$ c[n] jumps on frame start
- Frequency offset correction and symbol alignment done by sync long block.
  - $\Delta f = \frac{1}{16} arg \sum_{n=0}^{N_{short}-1-16} s[n]s^*\ [n+16]$, $s[n] \coloneqq s[n]e^{1n\Delta f}$
- Phase offset correction is started by FFT then Equalize symbols does phase correction and channel estimation.

From Bloess, Segata, Sommer and Dressler

Notes on Gnu Radio OFDM receiver

- Signal Field decoding
  - Determines length and encoding of symbols
- Frame decoding
  - Demodulate, demultiplex, descramble bit stream
- Socket PDU is a UDP server.
  - This is the output sink consisting of UDP datagrams
- Both BPSK and QPSK modulation are supported
- Sample system uses Ettus N210 and VERT2450 (3dBi) antenna

From Bloess, Segata, Sommer and Dressler

Software defined radio, part 5, amplifiers, beam forming, jamming and SDR RF attacks

- Direction finding
- Signal loss
- Jamming
- Beam forming and steering
- Attacking IoT devices

Directionality and steering

- Not all radio transmissions are omnidirectional.
- Dipole antennas transmit preferentially in the direction perpendicular to the surface of a cylinder enclosing the dipole wire and not in the third direction, perpendicular to the “caps” of the cylinder. Ground effects distort this geometry.
- Parabolic antennas are highly directional and have gains around 60 dB. A 2 meter parabolic dish has a beam width of about 2.6 degrees around 10 GHz.
- You can transmit (or receive) over several physically separated antennas (of any type) and use superposition to produce a highly directional transmission or reception patterns as phased array antennas do.
- In many applications, signal characteristics, like resolution, depend on the size or “aperture” of the antenna. That’s why high resolution telescopes are so big. In addition to efficient power use, highly directional antennas can be used, as we’ll see, to locate emitters.
- You can get commercial directional antennas, like here. These give about 10dB of gain.

Signal loss

- The “spreading” loss of a signal as a function of frequency and distance, is, roughly:
  - $L_S = 32.4\ (dB) + 20\log(f_{MHz}) + 20\log(d_{km})$
  - Interaction with the atmosphere adds another 2dB.
  - For 2.4 GHz, $20\log(f_{MHz}) = 67.6\ dB$
- At the receiver,
  - $P_R = P_T + G_T + G_R - L_S$
  - For example, if $P_T = 30dBm$, $G_T\ = 10dB$. $G_R = 3dB$, $P_R = -59\text{dBm}$, so $\frac{P_R}{P_T} =$ $10^{-5.9}\text{dBm}$. dBm is referenced to power in milliwatts. That is, $10\log(\frac{P}{10^{-3}})$ so $30dBm$ means P_T is 1 watt.
  - We see the power of the signal at the receiver is about 1.25 microwatts.
- The gains at the receiver and transmitter are characteristics of their antennas.

Receiver sensitivity

- The receiver sensitivity is the power of the weakest signal the receiver can reliably receive.
- For a crystal receiver with a pre-amp, $S_{max} = -114\ dBm + N_{PA} + 10\log(B_e) + SNR$.
  - B_e is the effective bandwidth
  - N_{PA} is the pre-amp noise (in dB)
  - SNR is the minimum required signal to noise ratio for the receiver per frequency bandwidth. This is about 6 dB for a good receiver.
- There is always background thermal noise (kTB) and often other noise. Thermal noise is about -114dB/MHz at about room temperature. So for example if the receiver bandwidth is about 100KHz, it's best thermal noise is about -124 dBm.
- If the receiver is not "ideal" there is additional noise measured by the "noise figure," N_F.
- Given the receiver receiver sensitivity and the signal loss, we can calculate the required power for a transmitter which can receive the given modulated signal.

Jamming

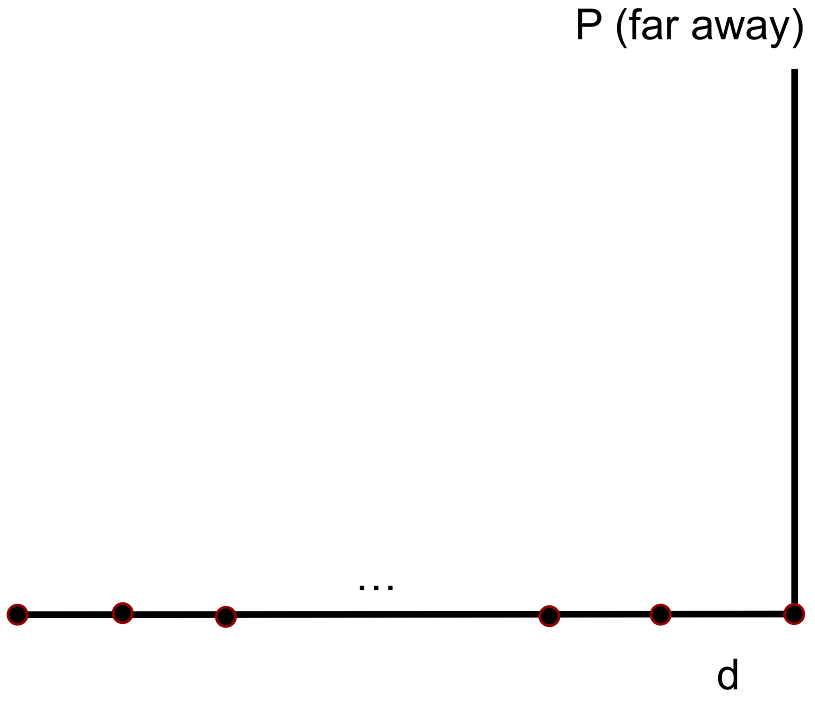
- Jamming seeks to interfere with an intended signal by presenting a higher power signal at the location of the receiver.
- Jammers may try to completely prevent reception or modify received signals. GPS deception is an example of signal modification.
- Power is a critical characteristic in jamming a receiver and efficient steering of available power.
- There are several anti-jamming techniques: spread spectrum and frequency hopping can be employed to prevent an prospective jammer from targeting a frequency range.
- Receivers can try to use antenna directionality to preferentially receive signals from a particular direction.
- With 802.11, use of encryption and cryptographic integrity protection can lessen the impact of eavesdropping; however, routing and destination is still visible in most schemes. If you encrypt, you’d better do key management correctly.

Jamming ODFM

- The critical factor in jamming is the power at the receiver passband(s). This depends on the geometry of the jammer transmission(s), the real transmitters(s) and the receiver as well as their power. These together with the frequency determines the jamming effectiveness.
- Recall ODFM has multiple channels encoding the signal and we need to jam enough of them to interfere with communications.
- By the way, if we present a jamming signal of significantly more power, we may be able to introduce a “false signal” and take over the channel rather than merely blocking it.

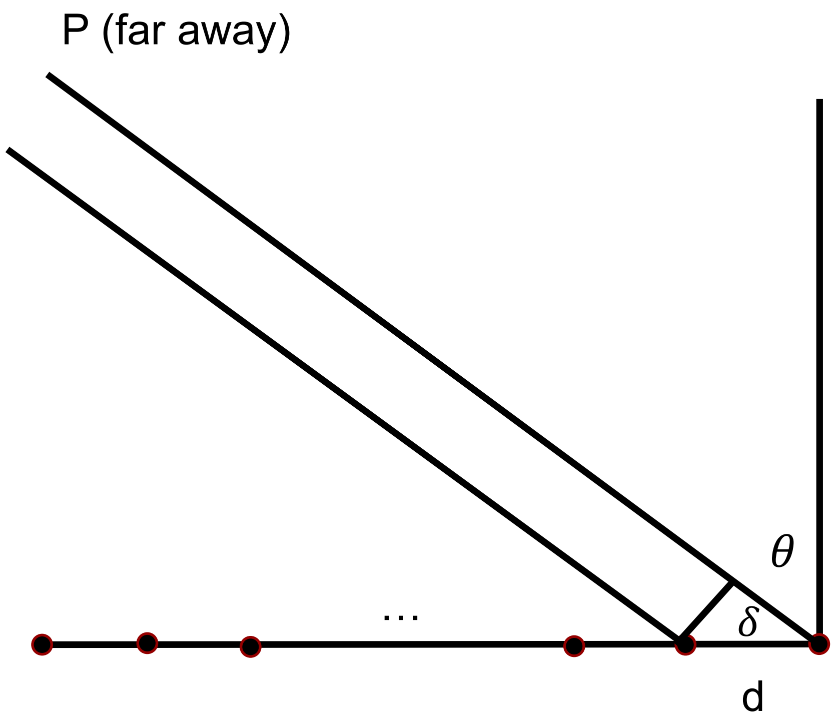
Array of coherent sources

- Suppose we have n equally spaced oscillator with phase delay α between successive emitters as depicted on the right, and a point, P far away. The received signal is:
  - $R = A[\cos(\omega t) + \cos(\omega t + \alpha) + \ \ldots + \cos(\omega t + (n-1)\alpha)]$
- R resolves into an oscillating signal times an amplitude. See, for example, Feynman Lecture son Physics, vol I, chapter 30. That is:
  - $f(t) = \ A_R \cos(\omega t + \delta)$
  - $A_R = A\frac{\sin(\frac{n\alpha}{2})}{\sin(\frac{\alpha}{2})}$
- If α is small, A_R is maximum (in absolute value) when $\frac{n\alpha}{2} = 2\pi k + \frac{\pi}{2}$, and minimum, when $\frac{n\alpha}{2} = k\pi$. Putting $\Delta = nd$, which is the length of the baseline of the array of oscillators, there is a minimum when $\Delta = 2\pi k$ and a maximum when $\Delta = 2\pi k \pm \pi$.



Beam forming and reception with multiple elements

- If P is at an angle from the vertical and if the oscillators still have a phase delay, $\alpha,$ signals from successive emitters have to travel an additional distance, $\delta = d\sin(\theta)$ so have an arrival delay $\frac{\delta}{c}$. So they have an additional phase delay $\phi = \frac{2\pi d\, sin(\theta)}{\lambda}$.
- The total phase delay is $\varphi = \alpha + \phi$.
- Let's pick $\alpha = 0$. From the previous result, the maximums are at $\Delta = \frac{2\pi n d\, sin(\theta)}{\lambda} = 2\pi k,$ or $\sin(\theta) = \frac{k\lambda}{\Delta}$.
- When we receive signals from P, we see the same phase effect so the signal strength is maximum when $\sin(\theta) = \frac{k\lambda}{\Delta}$. This allows us to find the direction of an emitter or tailor the power of a transmission so it is focused at the direction, θ.
- A parabolic antenna achieves the same effect with very narrow beam widths and, correspondingly, smaller angular uncertainty.



Additional information on phased arrays

- The configuration in the previous slide is called a phased array.
- The phased array beam-width $BW_{3dB} = \frac{102}{N}$
- The phased array gain is $G = 10\log(N) + G_e$, where G_e is the gain of an individual element.
- A phased array can only be steered within about 45 degrees of the boresight.

Broad band amplifiers

- Most SDR’s have low output power.
- You can buy broad band amplifiers to boost signal strength, see here and here.
- Remember to make sure emissions remain within regulatory limits or are done in a Faraday cage.

SDR and IoT

- SDR’s have become very cheap and processors (and FPGA’s) that run at high enough clock rates to do real time signal processing are now generally available.
- We will use a HackRF One SDR, which samples signals in the band 1MHz to 6 GHz and sends the digital time series to a computer through a USB connection. It also takes digital time series data and synthesizes transmission waves.
- We will write programs to process and generate signals. For example, we can transmit and receive 802.11 wireless signals, demodulate them and process the encoded data with an SDR.
- We and also “jam” or “spoof” such signals with the SDR.

Follow the rules

- You can receive and decode anything you want but the FCC regulates transmission over most of the radio spectrum. In unlicensed spectrum, transmission power is limited to .25 watts of power.
- If you get a “ham” license, much more spectrum is available and much higher power can be used.
- Hams have community repeaters and operate satellites.
- Toys and remote controllers use frequencies around 433 MHz.
- Radars use much higher frequencies. X-band radar (a common variant) operates at 8-12 GHz.
- You can use a spectrum analyzer to detect frequency emissions of cars, fobs wireless entry systems, etc.
- There’s a lot more out there.

Exercises

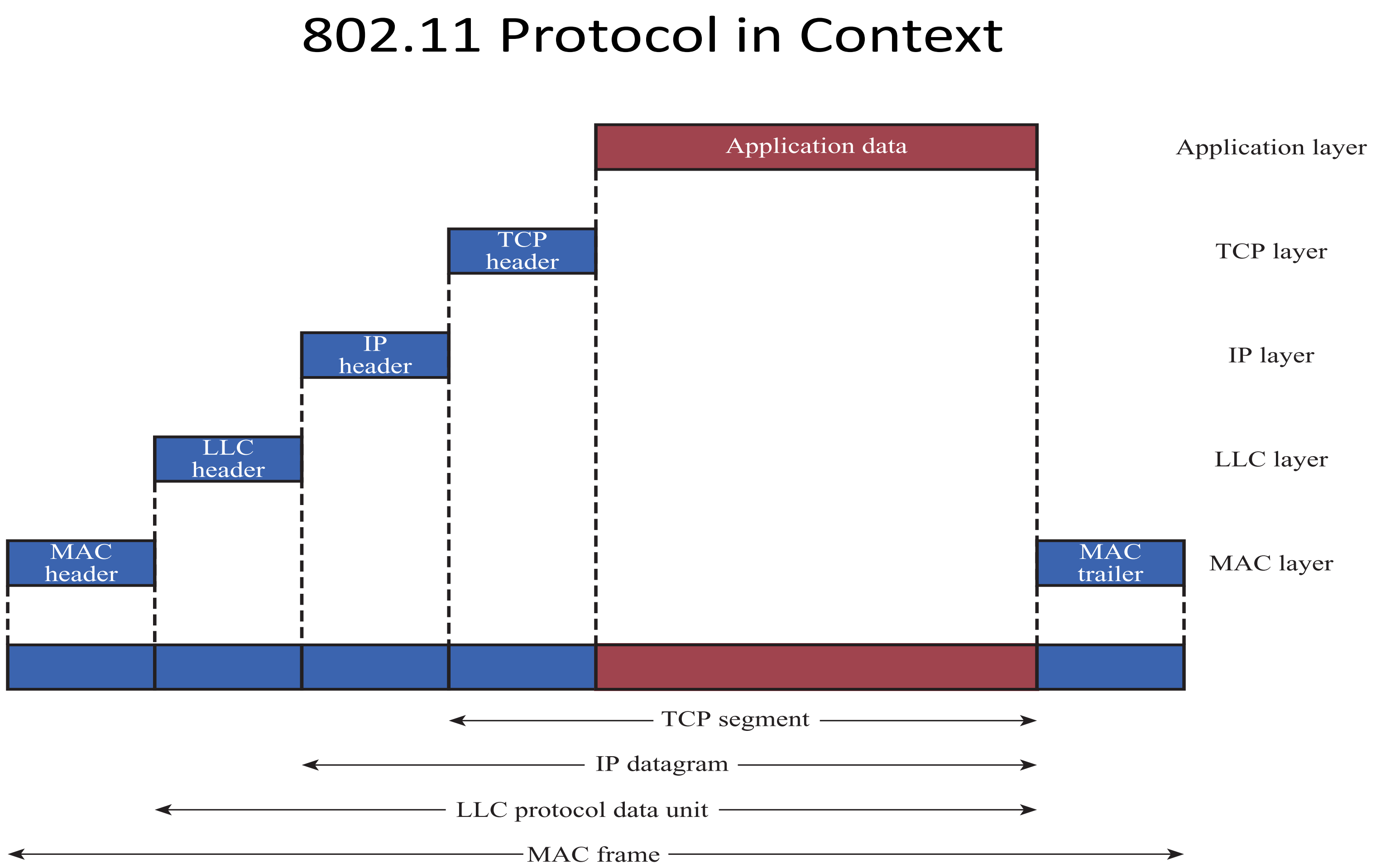
1. Build an FM detector from and SDR using GNU Radio
2. Intercept 802.11, BLE, Zigbee communication and log it.
3. Use 3 SDR's to jam a low power 802.11 transmitter.
4. Develop a GNU based SDR decoder to capture (unencrypted) airplane routing information.
5. Calculate the power needed to effectively jam IoT reception via wireless.
6. How many standard SDR's (or sdr's with standard amps) would you need? What is the area of effect?
7. Can you glitch IoT wireless communications with standard protocols? Are there any protocols that are resilient to this?
8. Can you apply RF to interfere with IoT performance? Is there an analogy with software fuzzing?
9. Reverse engineer and jam/interpose wireless weather station protocol
10. Reverse engineer and jam/interpose wireless car based protocol

References

- Michael Ossmann, https://greatscottgadgets.com/hackrf/ (Tutorial)
- GNU Radio
- ARRL Handbook
- Petrich and MacDermott, Digital signal processing and GNU radio
- Bloess, Segata, Sommer and Dressler, An IEEE 802.11a/g/p OFDM Receiver for GNU Radio
- Adamy, Electronic Warfare 101
- SDR for the masses

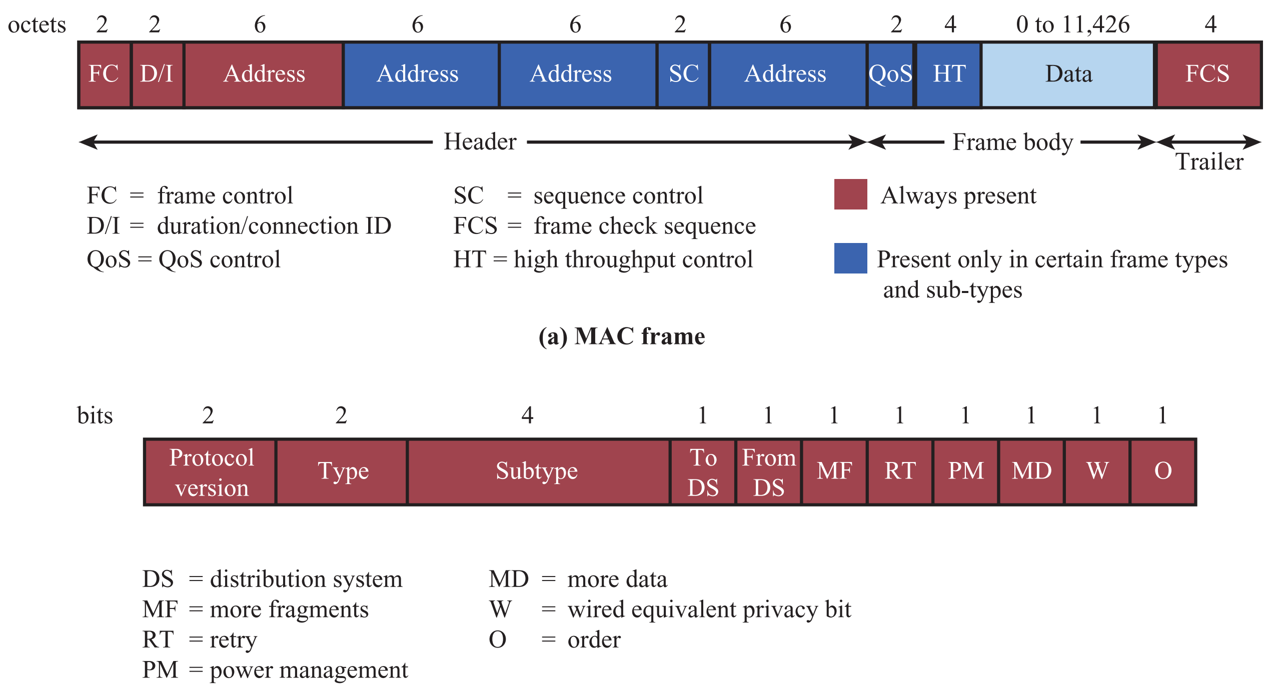
License

Beam forming 802.11 protocol stack



Mac frame format

802.11 MAC Frame Format



octets	2	2	6	6	6	2	6	2	4	0 to 11,426	4
	FC	D/I	Address	Address	Address	SC	Address	QoS	HT	Data	FCS

FC = frame control
D/I = duration/connection ID
QoS = QoS control

SC = sequence control
FCS = frame check sequence
HT = high throughput control

🟥 Always present

🟦 Present only in certain frame types and sub-types

(a) MAC frame

bits	2	2	4	1	1	1	1	1	1	1	1
	Protocol version	Type	Subtype	To DS	From DS	MF	RT	PM	MD	W	O

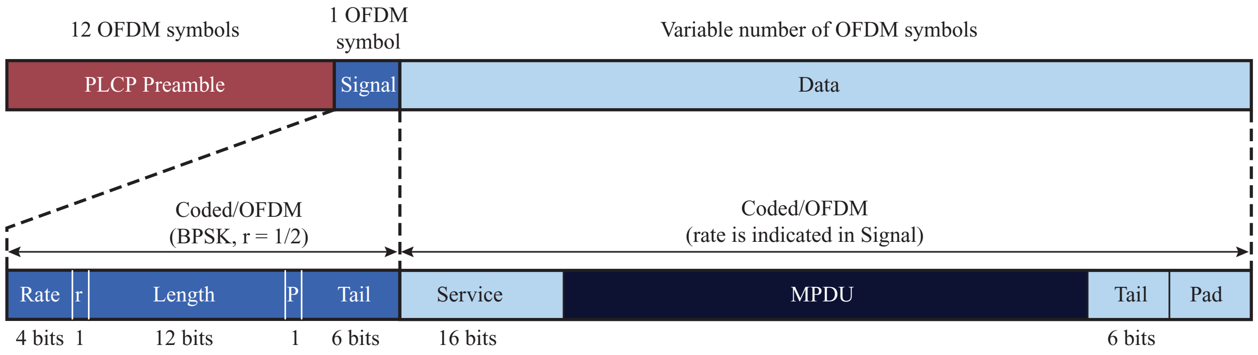
DS = distribution system
MF = more fragments
RT = retry
PM = power management

MD = more data
W = wired equivalent privacy bit
O = order

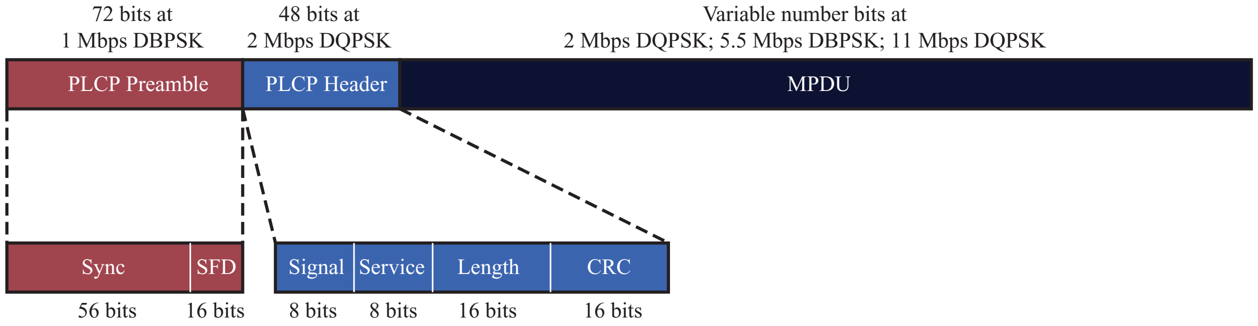
(b) Frame control field

OFDM integration

802.11 Physical Level Protocol Data Units



(a) IEEE 802.11a physical PDU



(b) IEEE 802.11b physical PDU

Radio and SDR

- Radios and transmission of EM waves.
- Propagation
- Modulation (OFDM, BPSK ...)
- Digital signal processing
- SDR’s
- GNU radio
- Electronic warfare: jamming, spread spectrum
- SDR’s: HackRF One, Ettus USRP