Modulation and Coding – labolatory

Digital Modulation

Amplitude Shift Keying (ASK)

The aim of the exercise is to develop algorithms for modulation and decoding for the two types of digital modulation: *Amplitude Shift Keying* (ASK). The first part of the exercise will be to build a simple ASK modulator and decoder. The second part of the exercise will be to build a modulator and decoder ASK modulation with using SDR devices (and transmitting a real signals).

1. ASK (Amplitude Shift Keying) modulation

There are two types of ASK modulation: On Off Keying (OOK) and Amplitude Shift Keying (see Fig.1)

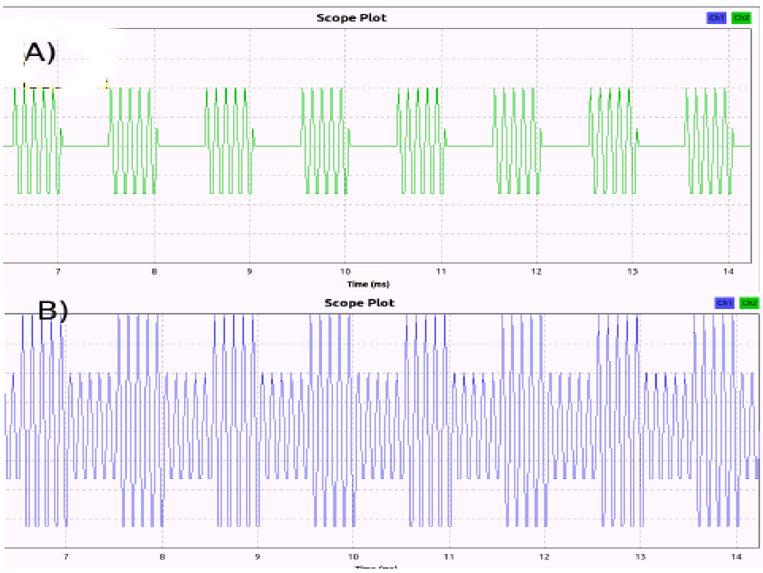

Fig. 1: Examples of digitally modulated signals (A) modulation OOK (On-Off Keying), (B) Modulation ASK (Amplitude Shift Keying)

Figure 1 shows a square wave (*fm = 1 kHz*) modulated by OOK method (Fig. 1a) and ASK (Fig. 1b). The carrier signal is a sinusoidal signal with a frequency *fc = 10KHz*. The main difference between the two methods of modulation signal it that, in the case of the ASK modulated signal is emitted during each symbol (even for '*Zero*'), see fig. 1b. For OOK modulation the carrier signal is not transmitted during for '*Zero'* symbol.

1.2 OOK / ASK Modulator.

The main difference is the form of the modulating signal and the method of decoding (in the case of digitally modulated signals are talking about modulation and decoding). Modulating signal now has a binary form - 0 and 1 (in general case may be *n*-VALUE, where *n>2*, e.g. 0,1,2,3,4 - then is possible to coding in one symbol *n* values).

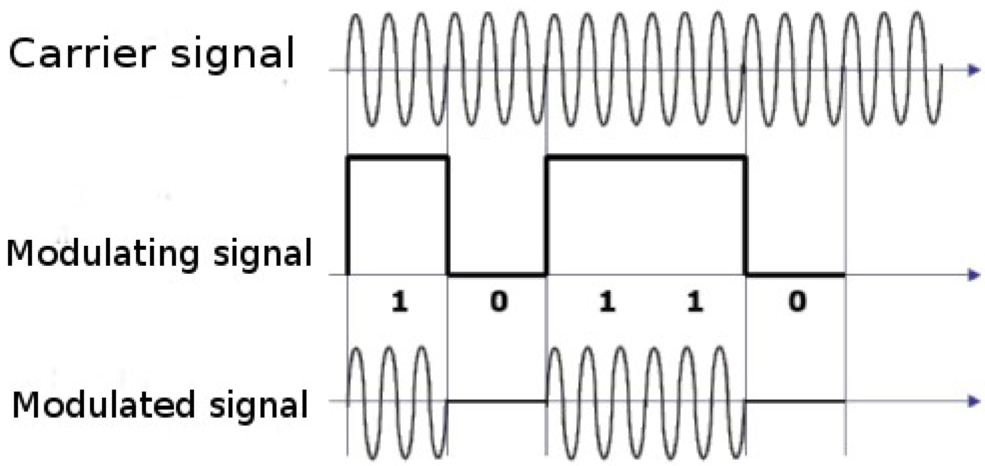

Fig. 2 On-Off Keying Modulation.

In the case of digital modulation the modulating signal is a digital signal. It can take the values 0 and 1 or -1 and 1. In the case of m-values modulators, the input alphabet can be bigger and contain exactly m values. Figure 2 shows modulating a binary signal (values 0 and 1) by OOK modulation. As can be seen as at the symbol time 0 signal is no transmitted.

How to make a ASK/OOK modulation in GnuRadio?

- Add a two signal sources: carrier signal (wave type: **sinus,** frequency = **10KHz**), and modulating signal (wave type: **square**, frequency = **100 Hz**). To adjust the frequency, use the component „*WX Slicer*” (add two “WX Slider” components. One for carrier frequency (*10KHz - 100KHz)*, and second for modulatig frequency (*10-1000Hz*)).
- Multiply these two signals by themselves.
- Show results of signal modulation (WX_SCOPE_SINK). This is OOK modulation.
- For change to ASK modulation, you must change amplitude levels of modulated signal (from 0 and 1, to e.g. 1 and 2). It can be done by add constant value to modulated signal (Component "add const").

- Show results of signal modulation (WX_SCOPE_SINK). This is ASK modulation.

1.3 Channel Model

The Transmission channel model is used to best reproduce the behavior of the signal during transmission, in the form of electromagnetic wave. To generate interference and noise is used AWGN (*Additive White Gaussian Noise*). It is a model of the channel in which the noise is constant and spectral power density (*watts / Hz*) and a Gaussian distribution of amplitude. This model is represented by the component "Channel Model". This parameter allows you to adjust the intensity of noise is called "*Noise voltage*". Should be regulated by the GUI (WX_SLIDER) in the range of *0 ... 1*V (preferably at one *mV*), the default value should be 0V.

1.4 OOK / ASK decoding

This time the aim is to recover (to decode) the symbols 0 and 1. The output signal should be exactly the same form as the input signal (although it may be shift at time). Decoder OOK / ASK might look like this:

- In order to filter out all other signals use the filter frequecny Xlating FIR Filter. It requires an application filter parameters as a string parameter (variable filter_taps). Create a variable called filter_taps, whose value will be as follows:

firdes.low_pass(1, samp_rate, fc, 25000, firdes.WIN_HAMMING, 6.76)

•

(Compare the parameter names of the typical characteristics of the filter with parameters from 'filter_taps' variable, ***fc* is a carrier frequency signal**).

- Next, we compute the squared module of the vector formed by the real part and the imaginary signal component via "*Complex Mag ^ 2*" component.
- The next step is decode the symbol. Each symbol has a fixed duration, amounting to tk. In our case it will be equal to twice the frequency of the modulating signal (in one period of symbol are contain two symbols 0 and 1). To decode the symbols use the component "*Clock Recovery MM*". The "*Omega*", which is a factor number of samples per symbol, set to *1* (bit rate equal to the velocity of symbols).
- Show results of signal modulation (WX_SCOPE_SINK). Compare with original modulated signal (see Fig 4).

- Check how behaves when the input signal is increasing the noise level.
- In order to improve the decoding can use a low pass filter to filter the noise. Place it between the "*Complex Mag ^ 2*" and "*Clock Recovery MM*". Adjust filter parameters to get the best results (To adjust the filter cutoff frequency, use the component „*WX Slider*", *Transition With* set to *100 Hz*).

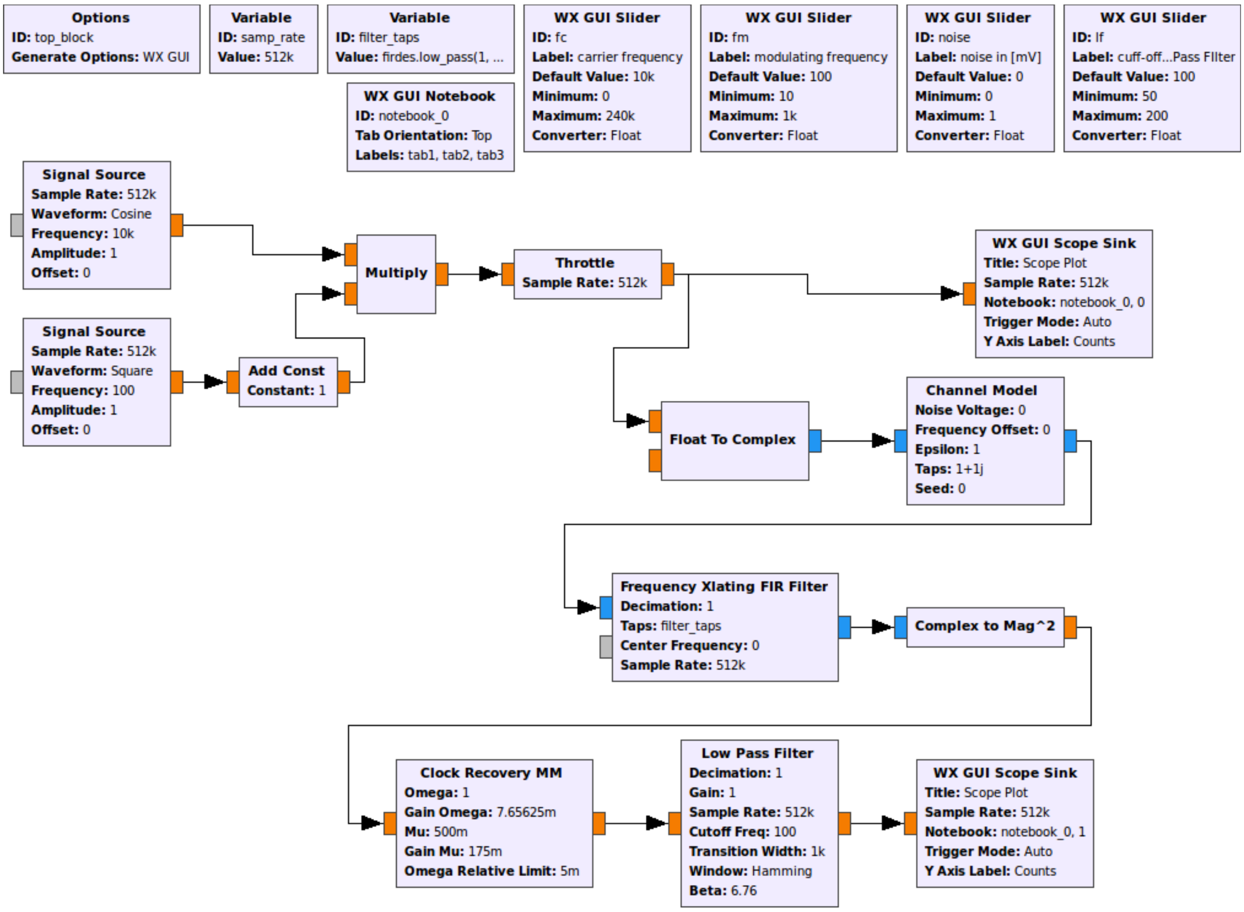

Fig. 3 Final flow diagram of ASK modulation and demodulation (with channel model).

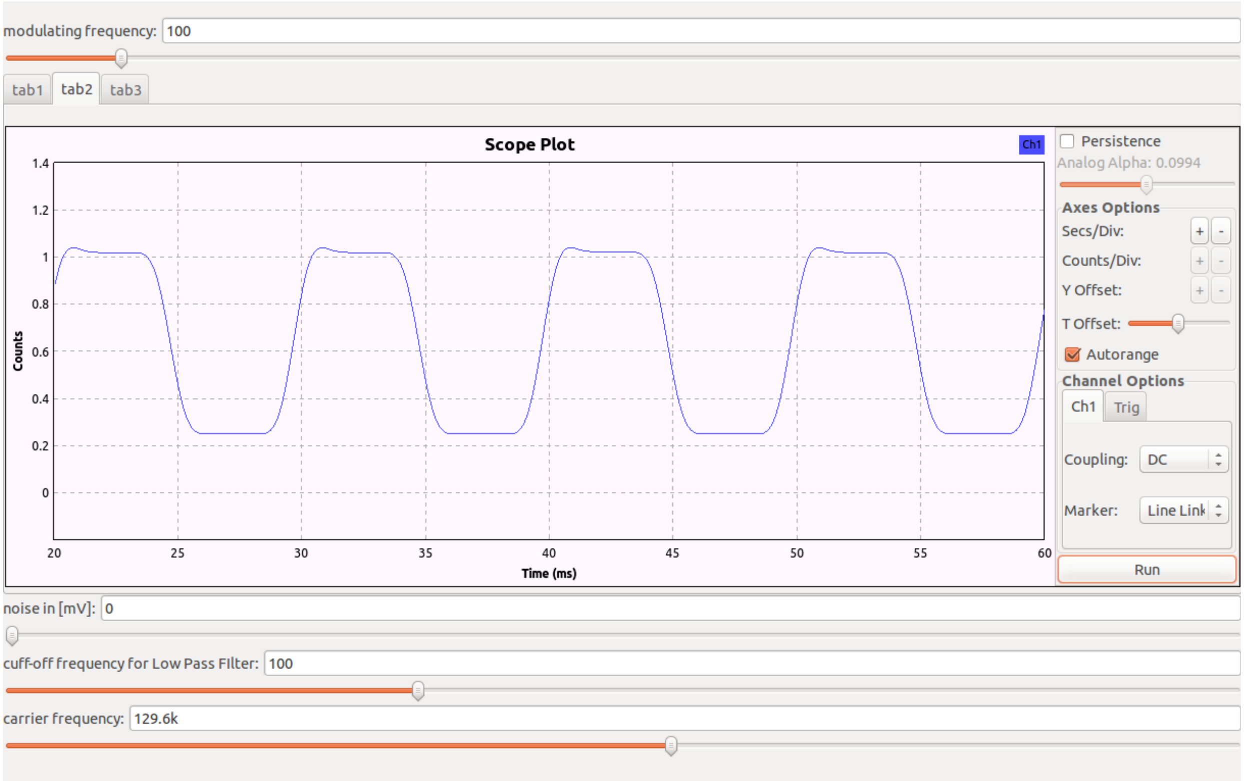

Fig. 4: Recovered ASK signal

1.5 M-ASK

Amplitude modulation keying (OOK, ASK) in the previous section allowed for two amplitude levels: the first level represented the first symbol of the alphabet, and the second level of amplitude - the second symbol alphabet. This is the version of the binary modulation ASK / OOK, sometimes called B-ASK (B-OOK). The alphabet contains only two symbols are, - for 0 and 1. However, you can increase the size of the alphabet. For example, the alphabet can contains 3, 4 or more symbols. In this case, by one symbol you can send more information.

There are two main parameters determining the transmission speed: *Symbol Rate* and *Bit Rate*. *Symbol Rate* mean the number of changes in the transmission medium per second (eg. how many times signal amplitude was changed (in one second)) and is measured in *Bauds*. *Bite Rate* is measured in *bits per second.* It can be calculated as below:

$$K = V * \log_2(n)$$

where:

V – symbol rate (in *Bauds*)

n - valence signal (binary - 2, octal - 8, etc.).

Example:

If symbol rate = *1000* baunds, and valence signal = *2* (binary signal), then Bite rate = *1000 bit/s*.
If symbol rate = *1000* baunds, and valence signal = *4* (binary signal), then Bite rate = *2000 bit/s*.

- To generate a signal modulated in M-ASK, enough to change a modulated signal. Now it should have at least a 3 different amplitudes. You can this by add two square signals, eg.
    - first signal: frequency = *100 Hz*, amplitude = *1*,
    - second Signal: frequency = *200 Hz*, amplitude = *2*.
- Show results of signal modulation (*WX_SCOPE_SINK*). Compare with original modulated signal.
- Is this the signal is more sensitive to noise? (You can use the constellation diagram - "*Constellation diagram / sink*").

2. Text transfer using ASK modulation and SDR devices.

Exercises require two posts (two computers and two SDR devices), so you can work at team. First team will be make a transmitter (section 2.2) and second one will make a receiver (section 2.3).

2.2 Transmitter

1. Prepare a file text (any text, not less than 10-15 kb).
2. Build the diagram as shown below:

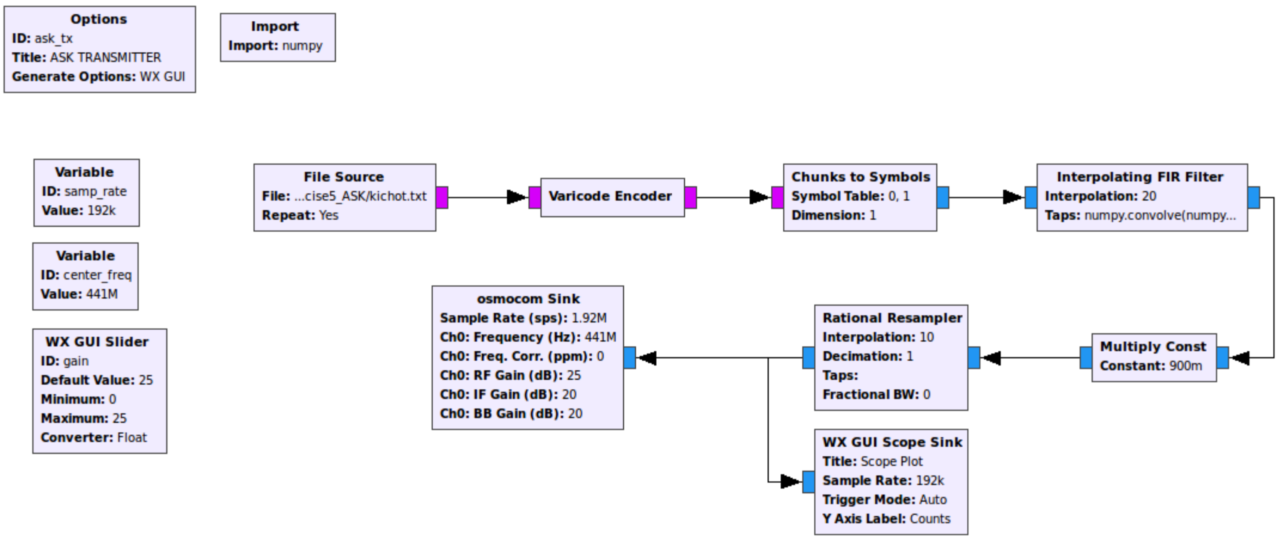

fig 5: Diagram for ASK trasmitter

- *Varicode encoder* - it's used for text compression, also covert byte stream to binary stream [0 and 1]
- *Chunks to symbol* – convert signal value to 0 and 1 and cover signal form *byte* to *complex*
- *Interpolatig FIR Filter* – a convolution FIR filter. In this case it's work as a g*aussian filter.*

  To set a filter characteristic you must type *(*in field *"Taps"):*

*numpy.convolve(numpy.array(filter.firdes.gaussian(1, 20, 1.0, 4*20)),numpy.array((1,) * 20))*

- *Rational Resampler* - changing a sample rate frequency

Proper modulation takes place inside the SDR device (mixing the signal generated in Gnuradio with the carrier signal – field "*CH0 Frequecny*" in *OsmocomSink*)

<u>2.3 Receiver.</u>

1. Build the diagram as shown below:

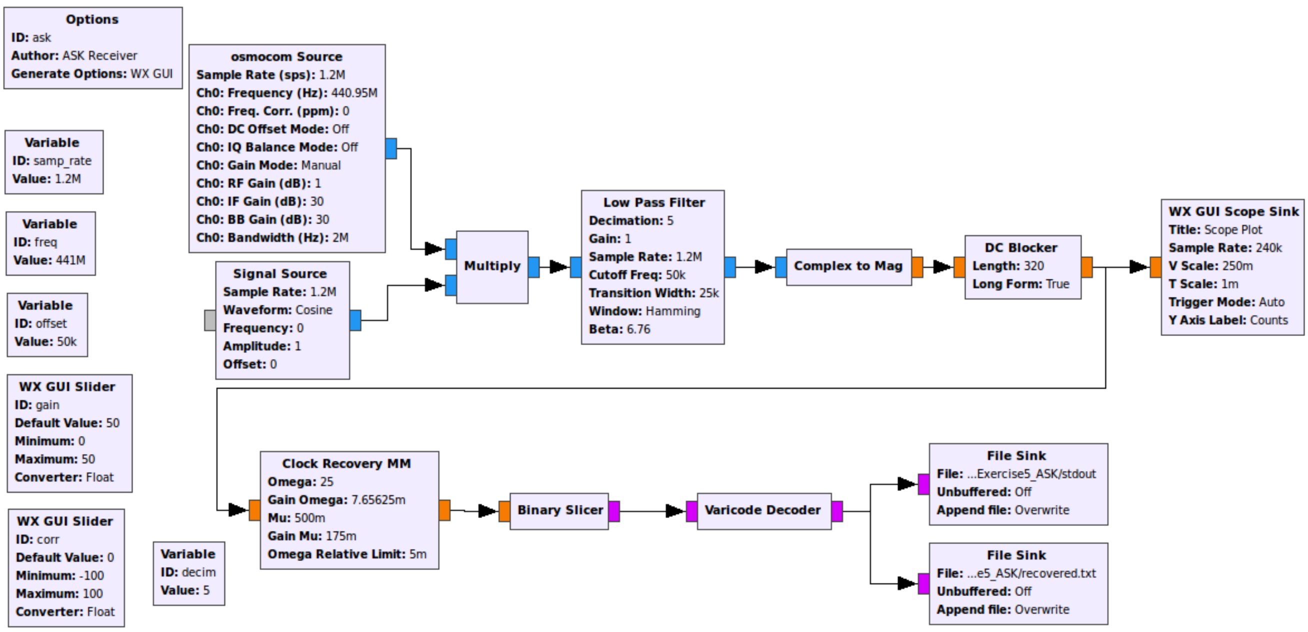

fig 6: Diagram for ASK Receiver

- *DC Blocker* - Blocks the DC component of a signal. This can be useful when working with AM signals, as taking the magnitude of the envelope will always be positive and thus introduce a DC bias on the signal.
- Clock Recovery - this block is need to recover a symbol form input signal (a must know what is symbol duration):
  - in field *"Omega"* type *(samp_rate/5/9600)*
  - Leave the remaining fields unchanged

- *Binary Slice* – Change float signal to binary stream
- *Varicode decoder* – decoded and return text into byte stream (ASCII code)
- Set the output File (component with “stdout” is use for displaying recoverer text into console), type (instead a file path)

/dev/stdout

2.4 Check trasmission

- Can you receive a correct text?