Modulation and Coding – labolatory

Digital Modulation

Gaussian Minimal Shift Keyng (GMSK)

The aim of the exercise is to develop algorithms for modulation and decoding for digital modulation named *Gaussian Minimal Shift Keyng (GMSK)*.

1. <u>GMSK moludation.</u>

GMSK modulation is based on MSK (Minimal Shift Keying), which is itself a form of continuous-phase frequency-shift keying. Here there are no phase discontinuities because the frequency changes occur at the carrier zero crossing points (see Fig 1.). This corresponds to a modulation index of *0.5.* One of the problems with standard forms of PSK is that sidebands extend out from the carrier. To overcome this, MSK and its derivative GMSK can be used. For example the GMSK is wide used on GSM communication.

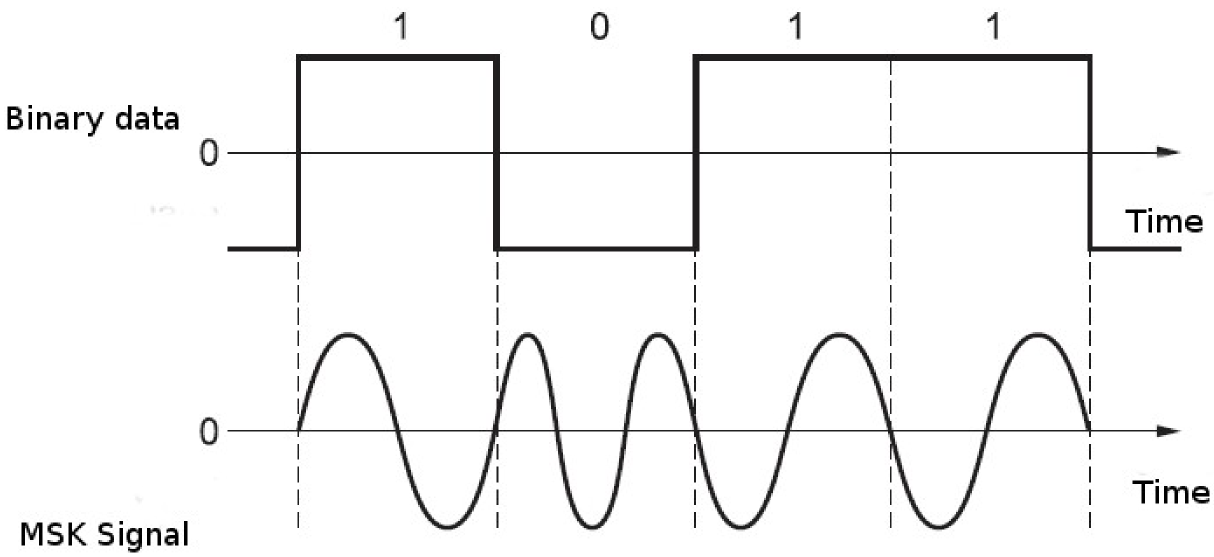

Fig. 1: MSK modulation

A fundamental difference between the MSK and GMSK modulation is that the square-wave signal (modulation signal) is replaced by a Gaussian pulse. The modulating signal is formed by a Gaussian filter, and after then is modulated by MSK method. The example of GSMK modulating signal are shown on Fig. 2.

There are two main ways in which GMSK modulation can be generated. The most obvious way is to filter the modulating signal using a Gaussian filter and then apply this to a frequency modulator where the modulation index is set to *0.5*.

A second method is based on Quadrature modulation. The term quadrature means that the phase of a signal is in quadrature or *90* degrees to another one. The quadrature modulator uses one signal that is said to be in-phase and another that is in quadrature to this. Using this type of modulator the modulation index can be maintained at exactly *0.5* without the need for any settings or adjustments.

In this exercise we will use a first method of GMSK modulation. This method is also used on GnuRadio (into GMSK modulator component).

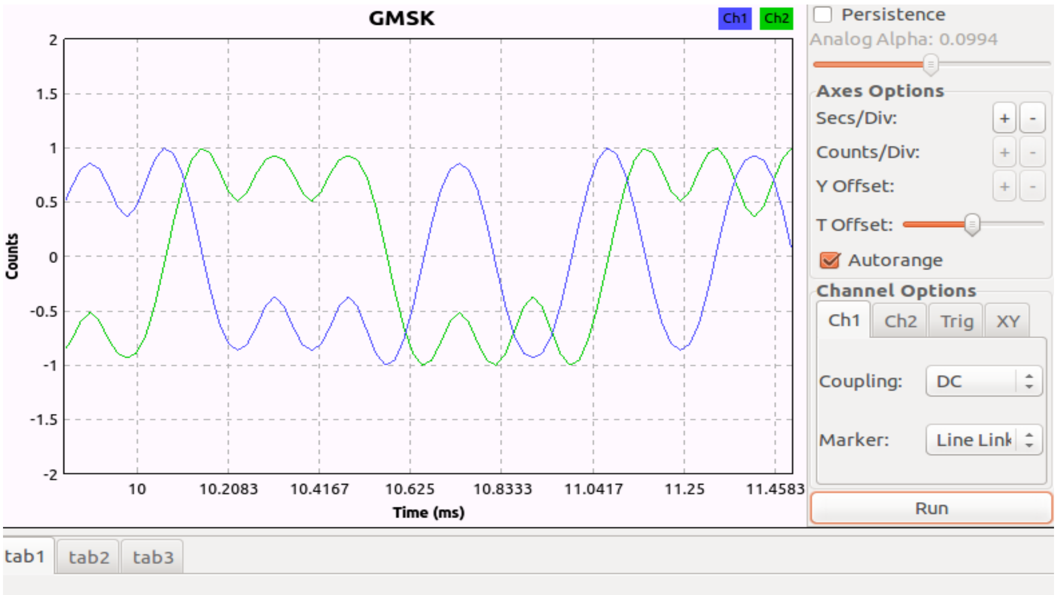

Fig. 2: Example of GMSK modulating signal.

1.1 GMSK modulator based on frequency modulation

- Prepare audio file. Read the audio file by "Wave file source" component. Generate a digital signal (via the codec G.723.40 - speech codec used in VoIP, bandwith 5-6 kbit/s), build the diagram as shown below (fig. 3):

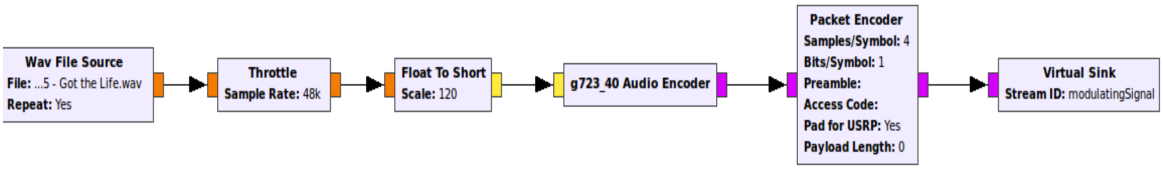

Fig. 3: Audio signal compresing by G.723.40 codek.

- Set *samp_rate* to 32...48KHz (Should be math to wave file sampling rate).

- Build GMSK modulator according to the following diagram (Figure 4 and 5):

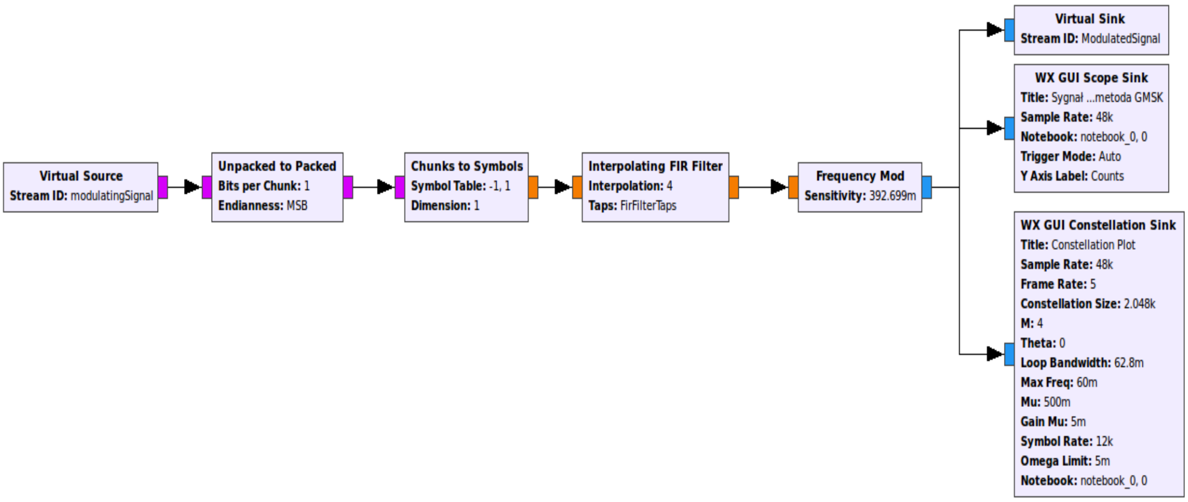

Fig 4: Block diagram of the GMSK modulator

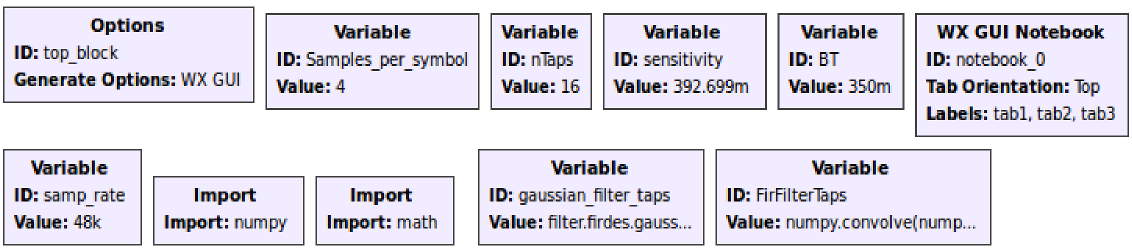

Fig. 5: Diagram of the GMSK modulator (variables)

- For importing additional libraries we can use a *Import* Component.

  In "value" field enter "*import math*" (for *math* library)

  In "value" field enter "*import munpy*" (for *numpy* library)

- Set value of “BT” variable to *0.35*.
- The variable "Sensitivity" determines the sensitivity of the modulator frequency. The variable "Sensitivity" is determined as follows:

  Sensitivity = (math.pi /2) / Sample_per_symbol

  enter only (in “*value*” field): *(math.pi /2) / Sample_per_symbol*

- Variable *nTaps (*which defines the size of the mask filter convolution*)* is calculated on the basis of *variable Samples_per_symbol:*

  *nTaps = Samples_per_symbol * 4*

- Parameters of Gaussian filter is set by variable *"gaussian_filter_taps":*

  filter.firdes.gaussian(1,Samples_per_symbol,BT,nTaps)

- The definition of Gaussian filter is shown as follows:

  *numpy.convolve(numpy.array(gaussian_filter_taps),numpy.array(numpy.array((1 ,)*Samples_per_symbol)))*

- Set parameters of *„Interpolatig FIR filter”* Component as below*:*

  In field *„Interpolation”* enter *"Samples_per_symbol" (*this is name (ID) of variable *“Samples_per_symbol”).*

  In field *„Taps”* enter *"FirFilterTaps" (*this is name (ID) of variable *“FirFilterTaps”)*

- Set parameters of *"Packet Encoder"* component as below:

  field *„Sample per symbol”* set to *4.*
  field *„Bits per symbol”* set to 1.

- Run the script. Check the result of modulation. It should be similar to that shown in Figure 2.

1.2 GMSK demodulator

- Build demodulation based on the block diagram shown in Figure 6 (in the same project as the modulator).

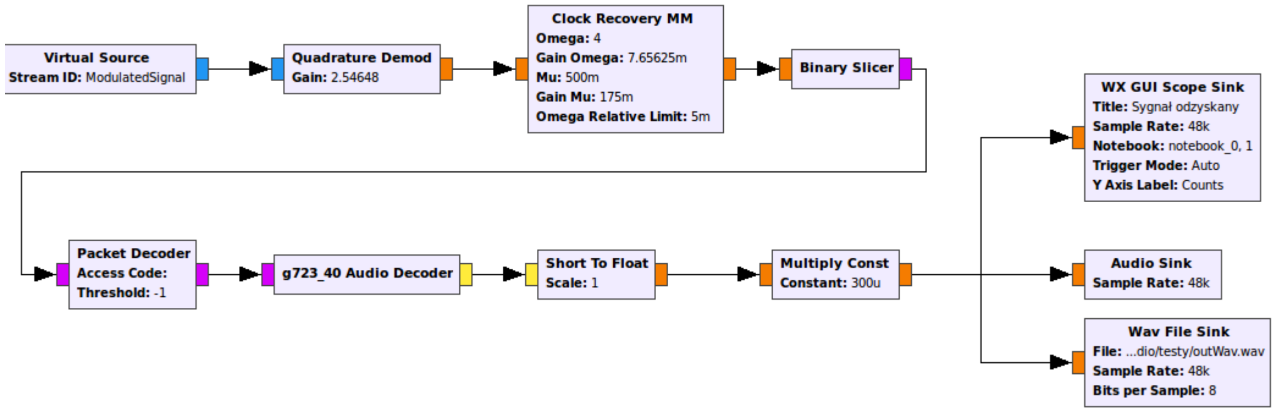

Fig. 6: GMSK demodulator (with audio decompression blocks)

Quadrature Demod component:

- Set value of "*Sensitivity*" variable to *1.0 / Sensitivity*

  (where "Sensitivity" is a name of variable – see Fig. 5 and paragraph 1.1).

Clock Recovery CC component:

- The „*Omega*" field component „*Clock Recovery CC*" fill in by variable „*Samples_per_symbol*".
- Set output wav file.

- Run the script. Is demodulated signal is correct?

1.3 File transmission using GMSK modulation (not obligatory exercise part):

- Try to send a (small) file, for example.: txt file, graphics file, pdf file (in this case you should remove all blocks which are using to audio compression / decompression).