**Modulation and Coding – labolatory**

# Digital Modulation

## BER – Bit error Rate

The **bit error rate** (**BER**) is the number of bit errors per unit time. The **bit error ratio** (also **BER**) is the number of bit errors divided by the total number of transferred bits during a studied time interval. Bit error ratio is a unitless performance measure.

The **bit error probability** $p_e$ is the expectation value of the bit error ratio. The bit error ratio can be considered as an approximate estimate of the bit error probability. This estimate is accurate for a long time interval and a high number of bit errors.

Example

As an example, assume this transmitted bit sequence:

0 1 1 0 0 0 1 0 1 1

and the following received bit sequence:

0 1 1 1 0 1 1 0 0 0,

The number of bit errors (the underlined bits) is, in this case, 3. The BER is 4 incorrect bits divided by 10 transferred bits, resulting in a BER of 0.4 or 40%.

*question 1: in binary case maximum of BER is 0,5 (50%).*

*Why ?*

## BER measured on GnuRadio

# I. Transmitter.

### 1. Constalletion object.

- Add new variable „const", set value to:

  *(digital.constellation_bpsk(), digital.constellation_qpsk(), digital.constellation_8psk())*

*(This a simple array, numbered from 0 to 2)*

- *Add new variable „const_type", set value to:1*

GnuRadio provides two constellation representations that we can use to more easily define and interact with constellation objects. We define these constellations with a set of constellation points in complex space and the symbol mappings to those points.  Example: For a constellation that has 4 symbols, it then has log2(4) = 2 bits/symbol. We define this constellation with:

```
constel_points = [c0, c1, c2, c3]
symbols = [s0, s1, s2, s3]
```

**2. Data sources:**

- On the right side, find "Random Source" from "Sources" category. Bring it to the  main window. Double click "Random Source" block and change the output type to  "Byte", the maximum to

const[const_type].*arity()*

*where*

*constellation* is name of GnuRadio constellatio object

and number of samples to 10M (try use *int(10e6)* instead of 1000000), and repeat to "No".

- Add Slicer, min values set to -100, max value set to 100, default values set to 10. „*EbN0*" Name this slider as "*EbN0*" (object ID), give label as "Energy Bit / Noise Power"
- Import a „math" liblary (compoment „*Import*", value: „*import math*")
- Add "Noise Source" from "Sources" category and change the amplitude as

*1.0 / math.sqrt(2.0 *const[const_type].bits_per_symbol() * 10**(EbN0/10))*

(amplitude can be adjusted to change SNR).

- Add "Chunks to Symbols" from "Misc Conversions" category and change the input type to "Byte", symbol table to "const[const_type].*points()*", the dimension to 1.

- Add "Add" from "Operators" category and make connections between "Chunks to

Symbols" and "Add" blocks, "Noise Source" and "Add" blocks.

- Add "QT Constellation Sink" from "QT" category  (if you use a QT liblary)  and make connections between this compoment and „Add" compoment.
- Click "Execute the flow graph" icon to execute it. Now you should to be able a constellation diagram for you choose modulation.

## II. Error measure

- Add "*Constellation Decoder*" from „*Symbol coding*". Set „Constellation obj" to

  *const[const_type].base()*

- Add "Error Rate" from "Misc" category and change the window size to 10M ( *int(1e7)* ), the bits per symbol to "*const[const_type].bits_per_symbol()*)".
- Add „*QT Number Sink*". Set input to float, *min* to *0*, and *max* to *1*. Make a connection between "Error Rate" and "Number Sink" blocks.
- Make a connection between "Error Rate" and "*Random Source*" via *Throtlle* block.
- Click "Execute the flow graph" icon to execute it. Now you should to be able a constellation diagram for you choose modulation and measured BER.

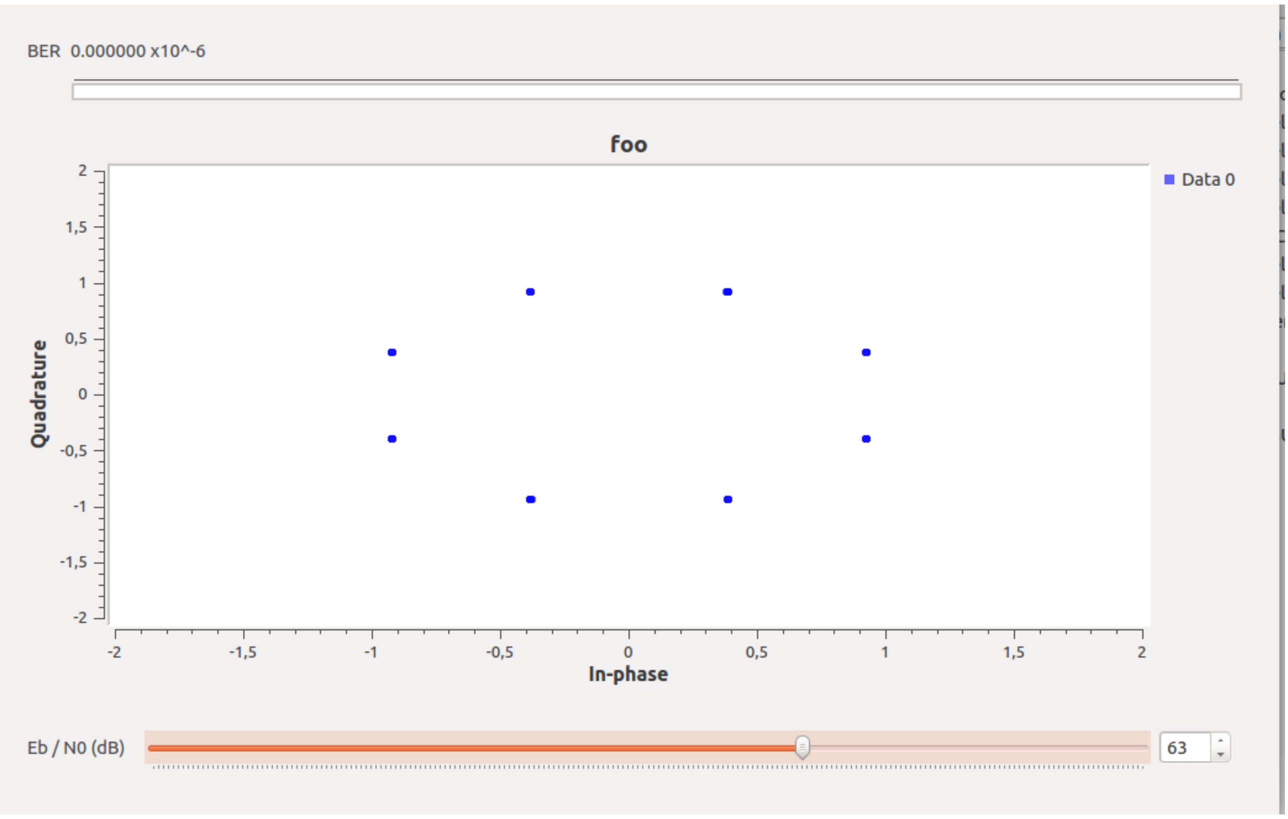


Fig 1: Example of BER measure with 8-PSK modulation

### III. BER measure – Exercises

1. Measure BER for all available modulations.

a) How does the size of the test block affect the BER measurement?

b) Which one is more less sensitive to noise when E0/ N0 = 9 ?

c) Draw a chart like on fig. 2 for all available modulations.

d) How does the number of possible symbols affect the sensitivity of modulation to noise (measured as a BER value)?

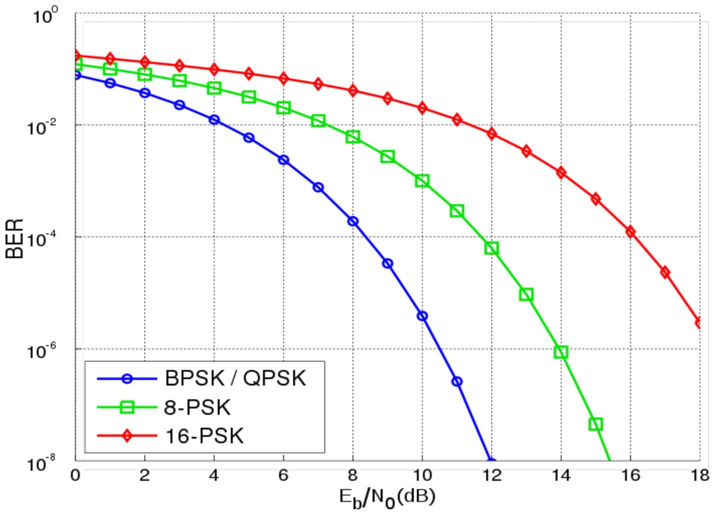


Fig.2: Bit-error rate (BER) vs $E_b/N_0$ curves for different digital modulation methods is a common application example of $E_b/N_0$. Here an AWGN channel is assumed [Get from wikipedia].

2. Measure BER for QAM 16:

- make a new copy of your project.
- Add a C*onstelltion* block. Choose a QAM16 modulation.
- Change the "*const[const_type]*" in your project (in all places) to a *constellation* (now you use a single instance of constellation object).
- Measured a BER like a case in previous modulations

3. Measure BER with SDR devices (not obligatory Part)

- Prepare a text file
- Prepare a transmission system (with ASK or FSK modulations, change "repeat" in "file source" to "false"), try send text file.
- Measure a phisical distance between transmitter and receiver antenna.
- Compare original and received file in Total commander (or another software), calculate BER value.
- Changle distance  between transmitter and receiver antenna and measure BER value again.
- Try draw a chart with distance transmitter and receiver antenna (on x axis) and BER (on Y axis).

  Is this any correlation between this distance and E0/N0 ?

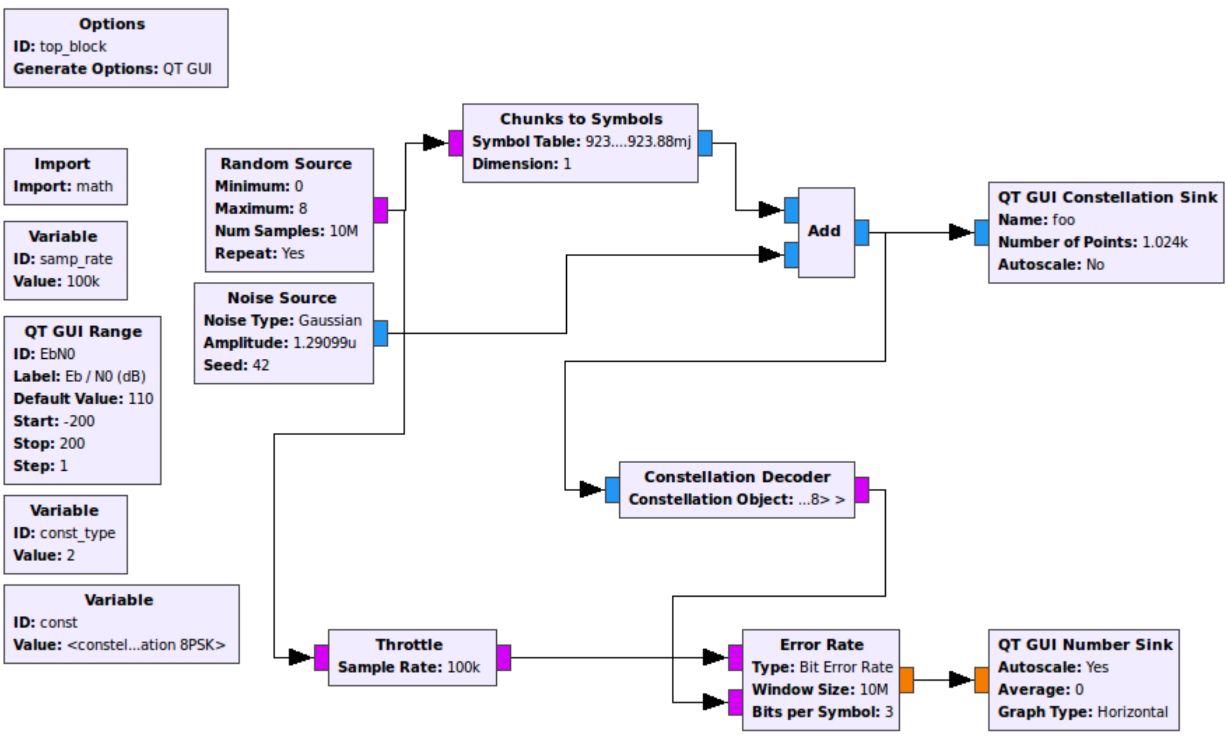


Fig.3: flowchart for exercise