Lab 10

Wadhwani Electronics Lab

Department of Electrical Engineering
Indian Institute of Technology, Bombay.

Legends

- Question/Observation: Show it to the TA and explain (carries marks)
- Recall/think about something
- Caution
- Additional information - weblink

Aim of the experiment

- **Part 1: Carrier Frequency and Phase Synchronisation in Communication Links**
  1. To study carrier frequency and phase offset problem in communication links.
  2. To design a Costas loop for achieving carrier frequency/phase synchronisation for QPSK signals.
  3. To get familiar with the dynamics of a phase-locked loop using the Costas loop (which is also a PLL).

- **Part 2: End To End Communication**
  The goal is to have you perform end to end data transmission. We will send a text file using three digital modulation techniques -
  1. BPSK
  2. Differential BPSK
  3. DIfferential QPSK

Pre-lab Work

- Go through the prelab study material.
- Make sure that you revise the previous lab experiments.
- Revise your concepts of control systems – Bode plots, stability criterion (gain margin, phase margin), pole zero compensation for stability. uploaded along with this document.
- Dynamics of a second order system: calculation of the natural frequency of oscillation of the feedback loop, damping factor and settling time.

Part 1:
Carrier Frequency and Phase Synchronisation in Communication Links

PART 1A: QPSK signal with frequency offset

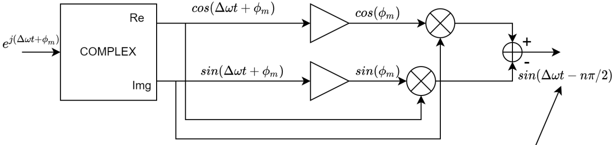
- Generate QPSK signals with sps=4(samples-per-symbol) and sample rate of 320 kHz.
- Use Random Source, followed by Chunks to Symbols, followed by Polyphase Arbitrary Resampler for this.
- Now multiply the incoming QPSK signal with 0.1 Hz complex sine wave i.e. with $e^{j2\pi ft}$.

✓ Use "Polyphase Clock Sync" with output SPS = 1 and plot the constellation in XY mode on a Scope Sink.

Observe that the constellation is rotating (Why?)

The signal from the multiplier output will be used to effectively represent "a down-converted base-band signal with 0.1 Hz carrier frequency offset".

Part 1B: Make a QPSK carrier phase detector



- ✓ Make the above Phase Detector. The above phase detector will have phase ambiguity of $\pm n\pi/2$
- A saw-tooth wave with frequency 40 Hz should be observed if you apply a frequency offset of 10 Hz in the Part 1 (Why?)
- Use Threshold Detector blocks for the above comparators (threshold levels +0.001,-0.001), followed by a subtraction of 0.5 (Why?)
  - The Threshold Detector block implements a Schmitt Trigger (comparator with hystersis with the two threshold levels).
  - Use a low pass IIR filter to suppress noise (*FF coefficients*: 0.01; *FB coefficients*: [-1,0.99]; *Old style of taps*: True)

PART 1C: Feedback signal through udp port

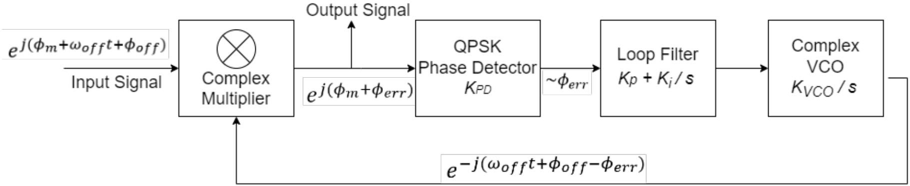
- GNU-Radio software doesn't allow blocks connected together in a feedback loop (however, a Python or C++ code canbe written to have feedback within the block).
- However, Costas loop requires a feedback. To overcome this imitation, we will give our phase detector output to a udp sink port and get it back through a udp source port.
  The Destination address and Destination port of both the udp sink and source is 127.0.0.1(refers to the IP of the localhost) and 12345 (this can be any unused udp port) respectively. Let the rest of the options remain in their default state.

⚠ However, there can be a problem that the udp source sends more number of packets leading to dropping of packets. To avoid this we introduce a delay of 10 samples after the udp source block.

⚠ Make sure that the output from the udp source block was not saturating, by multiplying with an appropriate constant.

PART 1D: Complete the Costas Loop

- Now give the resulting signal to the Loop filter (discussed in the prelab material) implemented using an IIR filter with the following parameters: FF taps: [1.0001,-1]; FB taps: [-1,1]; Old Style of taps: True
- Give this output to a Complex VCO with sensitivity of about -5 (why negative sign?).Multiply the VCO output to the signal with carrier offset that was generated in Part 1.
- ✓ Now observe the constellation after Polyphase Clock Sync (make sure the carrier offset was 0.1 Hz or less). It should stop rotating and settle to the desired constellation diagram. Vary the phase offset and determine upto what frequencies we can get back the original constellation.
- ✓ Observe roughly the time taken for the constellation to settle. Is it roughly equal to the settling time constant calculated by you?



PART 1E: 8-PSK signal with frequency offset

- Now instead of a QPSK, generate a 8-PSK signal and introduce a frequency offset of 10kHz as done in Part1.
- Implement the Viterbi-Viterbi algorithm after the polyphase clock sync.
- As we have to estimate the frequency offset we have to use differential decoding of the argument(arg(s[n]s*[n-1]) as done in the FM demodulation).

⚠ The differential decoding is followed immediately after raising the signal to its 8th power.

- The argument is to be given to a VCO(complex) block to generate the appropriate error signal.

⚠ The sensitivity of the VCO should approximately be the sample rate. Why??

✓ Observe the output of the VCO using the FFT sink and check if you are getting back the frequency offset that was given.

✓ Multiply the output of the VCO with the signal obtained from the polyphase clock sync. Observe the output and show it to your TA. Are you getting back your original 8-PSK constellation? If not, why??

PART 1F: Correcting the phase offset

- Now we need to correct for the phase offset obtained in Part 5. Just by changing the phase detector, we can use the same costas loop setup to achieve this.
- Implement the phase detector for the 8-PSK signals using the Viterbi-Viterbi algorithm and give the output of Part 5 as the input to the phase detector.

✓ Complete the costas loop as done in part 3 and part 4.

- Observe the output and show it to your TA. Are you getting back your original constellation?
- Compare the performance of the costas loop setup and the one using the Viterbi-Viterbi method. Which one is better?

Part 2:
End to End Communication

- This task requires you to perform end-to-end transmission as well as reception of a given text file with the help of several built-in GNU radio blocks that you have used in the previous labs.
- You must perform the transmission of the file using the following digital modulation schemes:
    1. BPSK
    2. Differential BPSK
    3. DIfferential QPSK
- Remember that every QPSK symbol comprises of two bits of information. So you might have to carry out packing of bits after unpacking. Same goes for demodulation of the signal.

Why Differential modulation?

- In case of transmission using BPSK, it is assumed that you have a perfect phase synchronization between the transmitter and the receiver. But what if you had a 180° phase shift between the transmitter and the receiver? Your constellation diagram would look perfect but your demodulated signal would be wrong!
- One solution to this problem is to not encode the information in the phase value of the output symbol, but in the phase differences.
- Repeat the whole task 1 (BPSK transmission) with minor differences i.e. introducing the differential blocks for DBPSK transmission. Information about additional blocks is given in the next slide.

Differential Encoder and Decoder

These blocks are to be used when doing Differential BPSK and Differential QPSK. The modulus value changes correspondingly, 2 for BPSK and 4 for QPSK.