Laboratory exercise 2
Quadrature Amplitude Mixer

1 Goal:

The goal is to achieve knowledge and practical experience in design of Quadrature Amplitude Mixer (QAM) for modern application specific computers, to get programming and debugging experience in VHDL language.

2 Theoretical information

QAMs are widely used in DSP applications for the frequency conversion in digital transceivers, filters, modems, radars, sonars, mobile phones, satellite receivers, etc.

Signal mixing means using a nonlinear operation, usually multiplying the input signal and a reference oscillator signal, to produce spectral images at the sum and difference frequencies. For example, consider a mixer of the radio receiver: if we “mix” an RF signal at 90 MHz with an oscillator at 89 MHz, the output of the mixer will have energy at 169 MHz (sum of frequencies) and 1 MHz (their difference). The 1-MHz signal becomes the signal of interest at the 1-MHz *intermediate frequency* (IF), while the sum frequency is easily filtered out. If the oscillator frequency is increased to 89.1MHz, it will translate an RF signal at 90.1 MHz to the IF; hence, channel selection, or tuning, can be realized by varying the oscillator frequency and tuning the output to the IF, using a fixed-frequency bandpass filter.

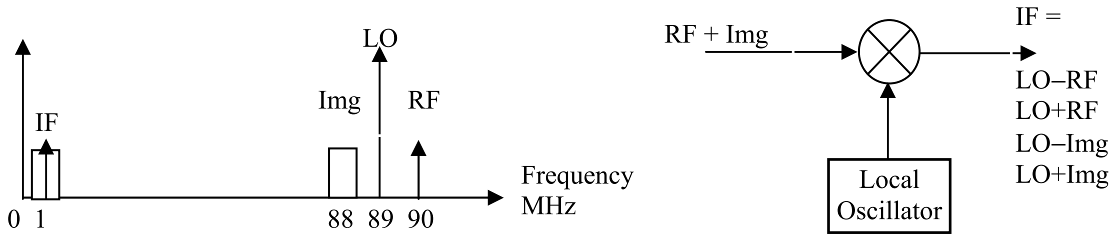


Fig.1.

However, when mixing the 90-MHz RF with an 89-MHz local oscillator (LO), any 88-MHz interference present on the RF signal will also be translated to a difference frequency of 1 MHz. Clearly, any RF signal at the “image” frequency of 88 MHz must be suppressed well below the level of the desired signal before it enters the mixer. This suggests the need for a filter that passes 90 MHz and stops 88 MHz, with a transition band of twice the intermediate frequency. This illustrates one of the trade-offs for IF selection, lower Ifs are easier to process, but the RF image-reject filter design becomes more difficult.

One worth mentioning because of its widespread use is *quadrature downconversion*. In-phase and quadrature representations (shortly – I and Q parts) of the input signal are mixed separately and combined in a way to produce constructive interference on the signal of interest and destructive interference on the unwanted image frequency. Quadrature mixing requires two (or more) signal processing channels.

It worth to be mentioned that the couple I and Q is the special signal representation which is named as the analytical signal. Mixing the input RF signal with the complex output of the LO gives the analytical signal in which the “image” signal is separated,

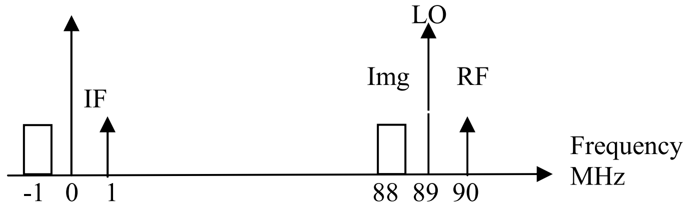


Fig.2.

Let the input signal is

$x(i)=A\cos(2\pi f_X\, iT+\phi)$,

where f_X is the LO frequency, T is quantization period, ϕ is some angle. The multiplication of this signal to the sine and cosine waves of LO results in the following terms:

$$I(i)=A\cos(2\pi f_X\, iT+\phi)\cdot\cos(2\pi f_X\, iT) = 0.5A(\cos(4\pi f_X\, iT+\phi)+\cos\phi), \qquad (1)$$
$$Q(i)=A\cos(2\pi f_X\, iT+\phi)\cdot\sin(2\pi f_X\, iT) = 0.5A(\sin(4\pi f_X\, iT+\phi)-\sin\phi).$$

We see that the resulting signal consists of the sine and cosine waves of the doubled frequencies and some DC parts which depend on the input signal phase. Usually the resulting signal of interest is the phase parameters $\cos\phi$, and $\sin\phi$. They are usually derived from the quadratures I and Q by their low pass filtering. The calculations (1) are implemented by the following schema:

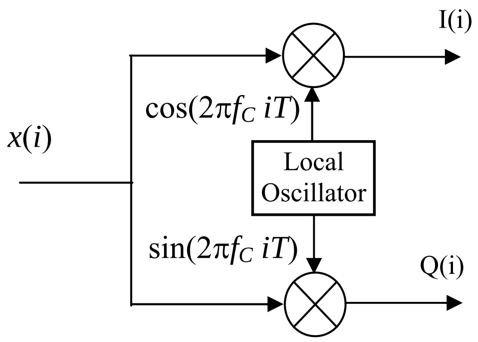


Fig.3.

3. QAM design example

QAM has the structure shown in the Fig.3. It is based on the LO model, which was designed in the previous laboratory exercise. Its parameters are: frequency f_X = 0,01f_S , input data width 16, output data width – 16, sine wave width – 16 bit.

The resulting QAM is described by the following VHDL code.

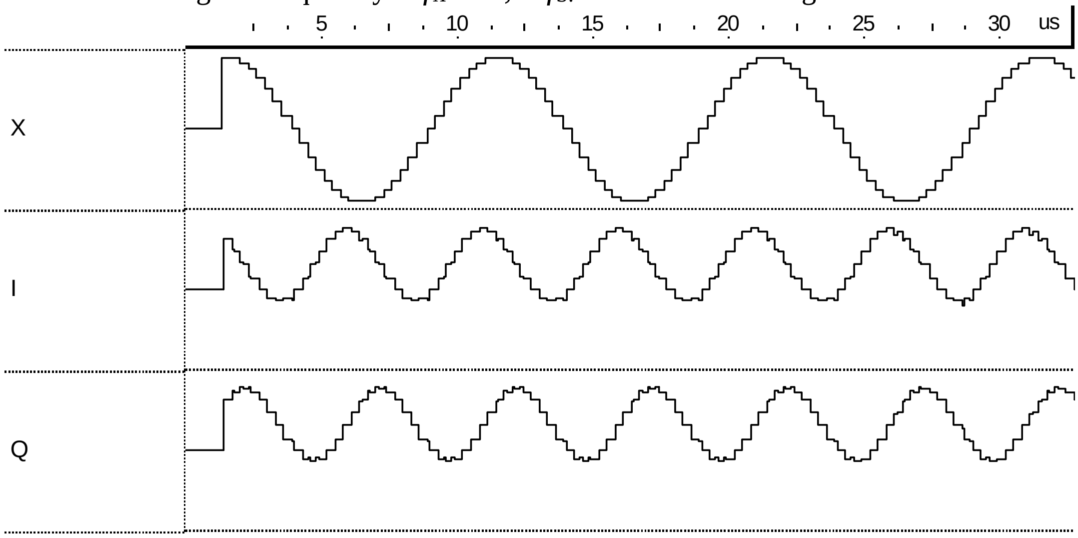
```
library IEEE;
use IEEE.STD_LOGIC_1164.all;
use IEEE.STD_LOGIC_SIGNED.all;
entity QAM is
  port( CLK : in STD_LOGIC;
        RST : in STD_LOGIC;
        X : in STD_LOGIC_VECTOR(15 downto 0);
        F : in STD_LOGIC_VECTOR(15 downto 0);
        Q : out STD_LOGIC_VECTOR(15 downto 0);
        I : out STD_LOGIC_VECTOR(15 downto 0) );
end QAM;

architecture QAM of QAM is

component SIN_GEN is
 port(CLK : in STD_LOGIC;
      RST : in STD_LOGIC;
      F : in STD_LOGIC_VECTOR(15 downto 0);
      SIN_O : out STD_LOGIC_VECTOR(15 downto 0);
      COS_O : out STD_LOGIC_VECTOR(15 downto 0));
end     component    ;
signal sini,cosi: STD_LOGIC_VECTOR(15 downto 0);
signal ii,qi: STD_LOGIC_VECTOR(31 downto 0);
```

```
begin
  U_LO:SIN_GEN port map(CLK, RST, F,
        SIN_O=>sini,
        COS_O=>cosi);

R_IQ:process(CLK,RST) begin  --
              if RST='1' then
                    ii<=(others=>'0');
                    qi<=(others=>'0');
              elsif CLK='1' and CLK'event then
                    ii<=X*cosi;
                    qi<=X*sini;
              end if;
        end process;
 I<=ii(30 downto 15);
 Q<=qi(30 downto 15);

end QAM;
```

Here the process R_IQ infers the multipliers and respective result registers. These registers are needed for the pipelined calculations and for the proper porting this QAM to another components of the system where it would be implemented.

To prove the QAM project the proper testbench has to be designed. This testbench consists of the QAM instantiation and some sine wave source. Such source can be the another instantiation of the sine wave generator which was designed in the 1-st laboratory exercise. Note, that the reset signals to this generator and to QAM must be held on the different time moments to make some phase shift ϕ.

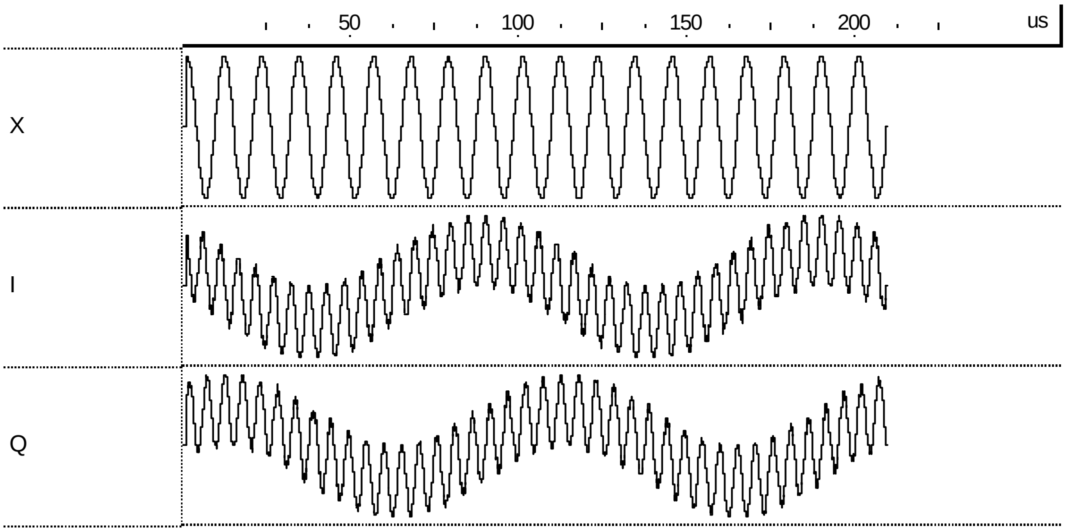
Consider the signal frequency is $f_X = 0{,}01 f_S$. Then the resulting waveforms are as the following:



From these waveforms we can see that

1) the resulting signal frequency is the twofold frequency of the input signal;
2) the resulting wave magnitude A is equal to a half of the input signal magnitude (taking into account the scale factor of the multiplication) ;
3) the DC components are estimated as I_D =24890; Q_D =23168; that means the phase shift which is equal to arctg(Q_D/ I_D) = 43,5°. This figure is near to the phase shift ϕ = 43,2°, which represents the given delay of 12 clock cycles between the reset signals (360° *12/100 =43,2°).

Consider the signal frequency is $f_X = 0{,}009 f_S$. Then the resulting waveforms are as the following:



The waveform analysis shows that the DC component is exchanged to the sine wave component with the frequency of $0{,}001f_S = 0{,}01f_S - 0{,}009f_S$. This proves the QAM proper operation.

4. **Laboratory exercise implementation**

The QAM has to be built and tested as in the previous example.

Each exercise variant has a set of parameters, which are numbered by natural numbers. A set of them is derived from the record-book number of the student. Consider 3 last figures a_2,a_1,a_0, of the record-book number. Then the variant number is

$N = 100a_2 + 10a_1 + a_0 = 2^9b_9+2^8b_8+2^7b_7+2^6b_6+2^5b_5+2^4b_4+2^3b_3+2^2b_2+2^1b_9+b_0$,

where b_i are the bits of the number N in the binary representation.

The QAM parameters are input signal bit width Ni and output signal bit width No is selected from the Table 1.

Table 1

$b_2,b_1, b_0,$	Ni	No
000	10	12
001	10	13
010	12	14
011	12	15
100	14	14
101	14	15
110	16	15
111	16	16

5. Laboratory exercise report

The laboratory exercise report must contain:

- Goal of the work,
- QAM description,
- VHDL texts,
- Waveforms of testing,
- Conclusions.

Literature

1. Отнес Р., Эноксон Л. Прикладной анализ временных рядов. –М.:Мир. –1982. – 428 с.
2. Рабинер Л., Гоулд Б. Теория и применение цифровой обработки сигналов. –М.:Мир. –1978. – 848 с.