A Programmable Signal generator

Uroš Minoski

***Abstract*—This paper represents a programmable signal generator as a project in undergraduate studies. Generator is composed of interpolator, complex mixer, NCO, filter for sinc compensation, DAC and analog NF filter. This paper does not deal with DAC and analog NF filter. Interpolator is designed as multistage interpolator in which every stage is polyphase structure of HB filter with attenuation of spectral replica of 60 dB . Complex mixer and NCO are designed as one element with CORDIC algorithm. Filter for sinc compensation is FIR filter so that reconstructed signal's amplitude deviates no more then 0.025 dB. DAC has 10-bit resolution. Analog NF filter is there to get rid of unwanted high spectral components that might be result of DA conversion**

***Index Terms*—Complex, interpolation, CORDIC, FIR, HB filter, DAC, mixer, NCO**

I. Introduction

Fig. 1 shows the block diagram of programmable signal generator. Input complex signal is needed for efficient use of spectra. Real time signals have Hermitian symmetry, so half of the spectra is wasted. On the other hand, complex time signals do not have Hermitian symmetry so there are not redundant information in spectra. Real time signals are not inherently complex, so we need two channels. One channel contains real part of complex signal, and the other channal contains imaginary part of complex signal.

Both channels need to be upsampled by a factor of 16 in order to ensure generated analog signals of greater frequencies then of FPGA source. That means that DAC must be able to work with maximum sampling frequency of 983.04 MHz. If reconstruction is done in higher Nyquist zones, generator can produce signals of even higher frequencies.

Complex mixer is needed for translating input signal to desired frequency. It is equivalent to multiplying signal by $e^{-j\Omega_0 t}$. NCO ensures desired digital frequency Ω_0.

Inherent property of zero-order hold reconstruction is scalloping, so it is necessary to have FIR filter for sinc compensation. FIR must be implemented so that reconstructed signal's amplitude deviates no more then 0.025 dB.

II. Design and Resoults

A. Interpolator

Fig. 2 show multistage implementation of an interpolator. This is the best scenario because equivalent filter order is minimum and in every stage interpolation is done by factor of 2 which allows implementation of half-band filters that more simplifies hardware. Benefits of multistage implementation are in less complex filters. Order of filter in direct implementation can be calculated using eq. (1) and is $N \approx 224$. On the other hand, using eq. (2) order of equivalent filter of multistage implementation is $N \approx 52$. Tab. I shows parameters, estimates

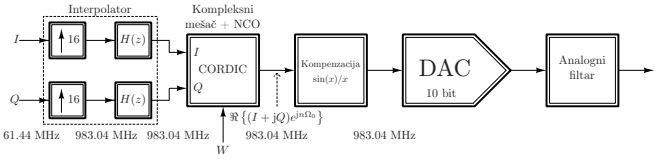

Fig. 1. Block diagram of programmable signal generator.

and actual filter orders and number of different coefficients in every stage.

$$N \approx \frac{1}{\Delta F}\frac{A_{dB}}{22}. \tag{1}$$

$$N_{eq} \approx \frac{A_{dB}}{22}\sum_{i=1}^{4}\frac{1}{dF_i}. \tag{2}$$

TABLE I
Parameters of filters in ever stage

	F_{pass}	F_{stop}	ΔF	N_{est}	N	# coeff
$A(z)$	0.203	0.297	0.09	30	34	9
$B(z)$	0.102	0.398	0.296	9	10	3
$C(z)$	0.051	0.449	0.398	7	6	2
$D(z)$	0.025	0.474	0.449	6	6	2

Advantage of polyphase decomposition of half-band filter can be seen in Fig. 3. One polyphase structure has only one nonzero element and the other has same coefficients as FIR type II used in half-band filter generation. Because the other structure has coefficients of FIR type II, they are symmetric so less hardware multipliers, for that structure, are needed. Number of multipliers is the seme to the number of different coefficients that can be calculated using eq. (3).

$$N_{coeff} = \frac{N+2}{4}. \tag{3}$$

Benefits of multistage and polyphase realization with half-band filters can be seen in number of operations. Number of operations in direct implementation is $3584f_s$, in one stage polyphase implementation is $224f_s$ which is 16 times better, in four stage polyphase interpolation is $52f_s$ which is 4.3 times better and if filter is half-band the number is $28f_s$ which is 1.86 times better then previous implementation and 128 times better then direct implementation. It is important to mention that number of operations is dimensionless quantity and all expresions must be devided by 1 Hz to ensure that.

Fig. 4 shows diagrams of spectra of signals, in blue, and amplitude response of filters, in red, in every stage of multistage implementation. As can be seen, most strict parameters are for filter in first stage and most relaxed are for a filter in the last stage. That means that the order of equivalent filter

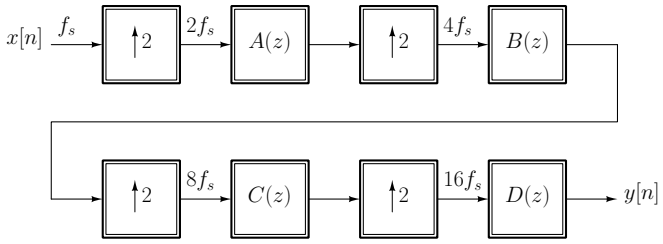

Fig. 2. Block diagram of multistage implementation of interpolator.

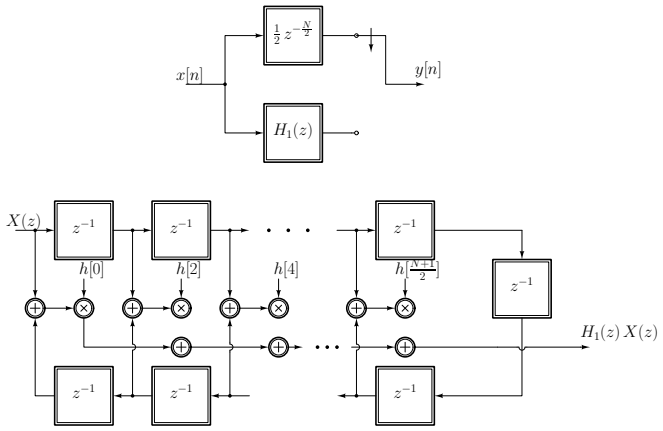

Fig. 3. (a) Block diagram of polyphase implementation of HB filter. (b) Block diagram of hardware realization of $H_1(z)$.

of multistage implementation will be sure less then of direct implementation. Output of the stage will be signal multiplied by filter's amplitude response.

B. Complex Mixer and NCO

Fig. 5 shows NCO and complex mixer. The property of CORDIC algorithm allows simple implementation of complex mixer. In order to find trigonometry functions $\sin(x)$ and $\cos(x)$ initial conditions should be $(x_0, y_0)^T = (1, 0)^T$ but for complex mixer they should be $(x_0, y_0)^T = (I, Q)^T$ where I and Q are real and imaginary part of complex time signal respectively.

Number of control word bits, to ensure minimal frequency resolution of 1 Hz, can be calculated, using eq. (4), to be 26 bits.

$$L = \log_2 \frac{f_s}{f_{min}}. \tag{4}$$

Output of NCO represents anlge of rotation for CORDIC. Inside CORDIC block, angle acumulator must be truncated to N + 2 bits. Aditional 2 bits conclude in which quadrant vector is located. Truncation can cause degradation in SNR.

Resolution of CORDIC is tightly linked to resolution of DAC so it is 10-bit resolution. In order to achieve that, algorithm needs to have 10 iterations. Guard bits are added to LSB in order to have error less then one LSB while doing shift calculations in algorithm. Number of guard bits can be calculated using eq. (5) to be 4. In total, number of bits for x and y is 14 and number of bits for z is 12. Output of CORDIC must be truncated to 10 bits by removing guard bits.

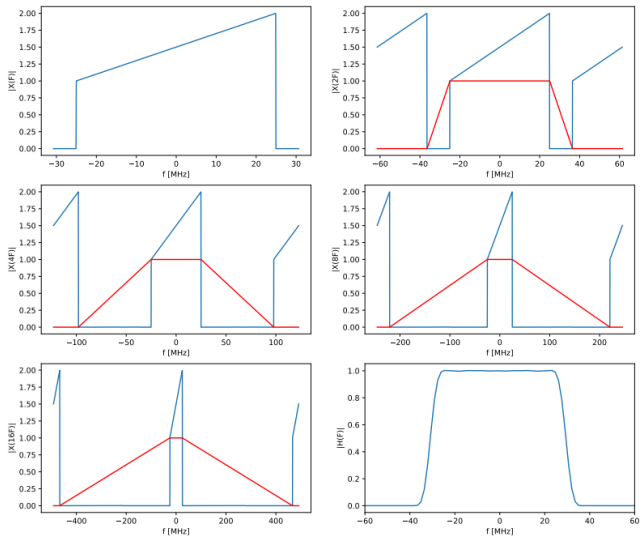

Fig. 4. Spectre of output signals, in blue, and amplitude response of filters, in red, in every stage. (a) Spectra of input complex signal. (b) Output spectra of complex signal in first stage, in blue, and amplitude response of filter. (c) Second stage. (d) Third stage. (e) Fourth stage. (f) Amplitude response of equivalent filter in multistage implementation.

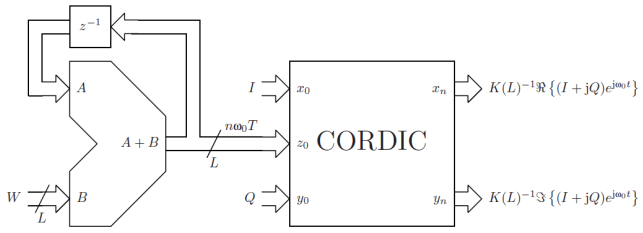

Fig. 5. Block diagram of NCO and complex mixer using CORDIC algorithm.

$$n_{guard} = \log_2(n). \tag{5}$$

C. FIR Filter for sinc Compensation

FIR filter for sinc compensation is designed with Remez algorithm. Order of filter is 9 in order to ensure no more then 0.025 dB amplitude deviation of reconstructed signal. Coeficients of filter are h = [0.002, -0.008, 0.025, -0.097, 1.156, -0.097, 0.025, -0.008, 0.002].

D. Jitter Losses

Noise in clock signal can cause uncertainty in acquisition time. Jitter depend on frequency of a signal. Eq. (6) shows SNR due to jitter and eq. (7) SQNR, where N is the number of bits.

$$SNR = 20\log_{10}\frac{1}{2\pi f t_j}. \tag{6}$$

$$SQNR = 6.02N + 1.76. \tag{7}$$

In order to have same SNR due to jitter as SQNR on frequency 884,74 MHz, maximum jitter time t_j should be 0.144 ps. The total SNR will in that case be 65 dB, for 3 dB more then SQNR.