XVII IMEKO World Congress
Metrology in the 3rd Millennium
June 22–27, 2003, Dubrovnik, Croatia

COUNTER BASED FREQUENCY SYNTHESIZER

Milan Stork

Faculty of Electrical Engineering, University of West Bohemia, Plzen, Czech Republic

Abstract - This paper describes architecture a new pure digital frequency synthesizer based on generators, counters and a register. The technique described here is much simpler then other method. Presented synthesizer is the most suitable for the design of VLSI architectures or for programmable Large Scale Integration. On the other hand, this synthesizer has a disadvantage in low output frequency, but this can be overcome by using this synthesizer together with phase locked lop.

Keywords: frequency synthesizer, phase locked loop, delay-locked loop.

1. INTRODUCTION

The aim of frequency synthesis is to generate arbitrary frequency f_X, from a given standard frequency f_S, it means to solve the equation (1):

$$f_X = k_X * f_S \quad (1)$$

where k_X in the simplest case is a fraction formed by small, relatively prime integers. That is,

$$k_X = X_1 / Y_1 \quad (2)$$

and the synthesizer is reduced merely to chain of one frequency divider and one multiplier. If X_1 and Y_1 in (2*)* are products of small prime numbers, the synhthesizer may be realized by chain of frequency multipliers and dividers. However, there are difficulties with hardware solutions, mainly generation of spurious signals and frequent enhancement of the phase noise level.

Reduction of the weight and size was provided by phase locked loop in frequency synthesizers. The most recent achievement in frequency synthesis design is the creation of direct digital frequency synthesizers (DDS) [1].

In this paper a new simple architecture of digital frequency synthesizers with square wave output is presented.

2. FUNCTIONAL DESCRIPTION

In the Fig. 1, there is a block diagram of the digital frequency synthesizer [2]. It consist of Counter 1, which count up frequency f_{C1} gated by input frequency f_X. Parallel output from Counter 1 is connected to Register input and Register output is connected to preset inputs of Counter 2 which counts down frequency f_{C2} . On the output of this Counter 2 there is frequency f_Y. It is expected, that $f_{C1} > f_X$. Number *Nc1* which is stored in Counter 1 during the period of the f_X is given by (3):

$$Nc1 = f_{C1} / f_X \quad (3)$$

This number is written in the Register, where his value can be changed by the Control to *Nc2*:

$$Nc2 = g(Nc1) \quad (4)$$

where *g(.)* denote some function of *Nc1*.

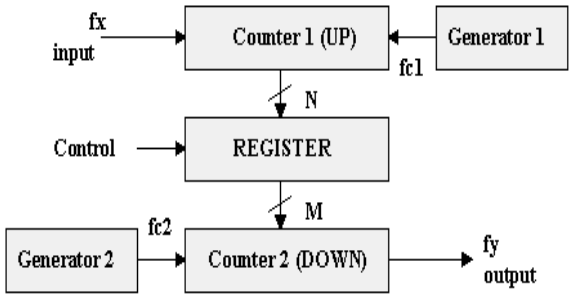

Fig. 1: Block diagram of the proposed architecture

Number *Nc2* is given by (5):

$$Nc2 = f_{C2} / f_Y = g(Nc1) = g(f_{C1} / f_X) \quad (5)$$

Output frequency f_Y can be expressed from (5) by (6):

$$f_Y = f_{C2} / g(f_{C1} / f_X) \quad (6)$$

When, for example function $g(.) = 1/k_1$ (which can be simply realized by shift binary number in the Register) output frequency f_Y is given by (7):

$$f_Y = f_{C2} * k_1 * f_X / f_{C1} \quad (7)$$

Equation (7) shows, that output frequency f_Y is a products of frequency f_{C2},, k_1 and input frequency f_X divided by frequency f_{C1}. All of these parameters can be individually set. The length of the counters and registers must be sufficient to prevent overrun. If the binary counter is expected, then minimal length *L* of the Counter 1 [bit] is given by (8):

$$L => Ceil(\log_2 (f_{C1MAX} / f_{XMIN})) \quad [\text{bit}] \quad (8)$$

where f_{C1MAX} and f_{XMIN} are maximal clock and minimal input frequency and *Ceil* function converts numeric value to an integer by returning the smallest integer greater than

or equal to its argument. In Fig. 2, the synthesizer is shown as a building block.

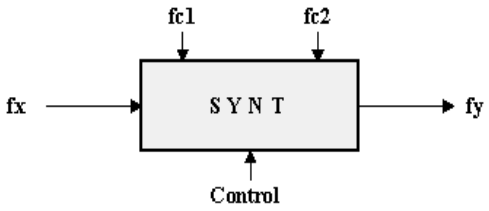

Fig. 2. The digital frequency synthesizer as a building block

3. USING THE NEW FREQUENCY SYNTHESIZER IN THE PHASE LOCKED LOOP

The phase locked loop (PLL) [3], [4] works as a feedback system. The task of PLL is to maintain coherence between input (reference) signal frequency, f_i, and the respective output frequency, f_O, via phase detector (PD) [5]. When PLL locks onto a reference signal the output frequency is given by (9):

$$f_O = N * f_i \tag{9}$$

where N is an integer divide number of divider.

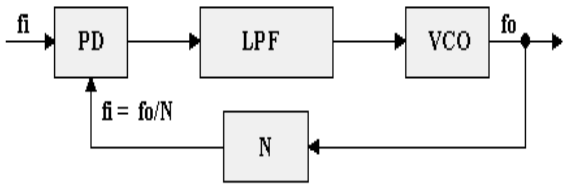

Fig. 3. Block diagram of the basic Phase-Locked Loop. PD - phase detector, LPF - low pass filter, VCO -voltage controlled oscillator, N - frequency divider by N (or multiplier by M). M,N are integer numbers

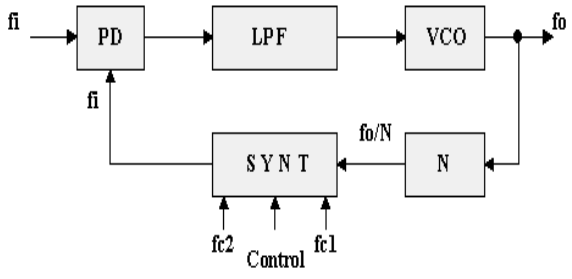

Fig. 4. PLL with the digital frequency synthesizer

Normally, frequency dividers can only produce integer divide ratios (N is integer). Fractional division is accomplished by alternating the instantaneous divide number between *N* a *N+1*, but this causes phase modulation on the VCO [6]. Therefore a different complicated technique is used for correction of this error [7]. In Fig. 3, the SYNT circuit is used in PLL [8].

Frequency on the SYNT input is f_O/N and frequency on the SYNT output is given by (10):

$$f_i = f_{C2} * k_1 * f_O / (f_{C1} * N) \tag{10}$$

From (10) we can derive the frequency of voltage controlled oscillator which is shown in (11):

$$f_O = f_{C1} * N * f_i / (f_{C2} * k_1) \tag{11}$$

In case that number *Nc2* in register is given by (12)

$$Nc2 = m_1 * Nc1 \tag{12}$$

(binary number *Nc1* is multiplied by m_1, e.g. register is shifted to left, instead of divided by k_1), the frequency of voltage controlled oscillator is given by (13):

$$f_O = f_{C1} * m_1 * N * f_i / f_{C2} \tag{13}$$

From the (13) we can see, that output frequency f_O is a function of integer m_1 , N and clock frequencies f_{C1}, f_{C2}.

4. ERROR REDUCTION IN SYNTHESIZER

The digital synthesizer in Fig. 1, has a following disadvantage. When the numbers in counters are small (integer numbers), the output frequency is not accurate. This error can be improved by adaptive control shown in Fig. 5.

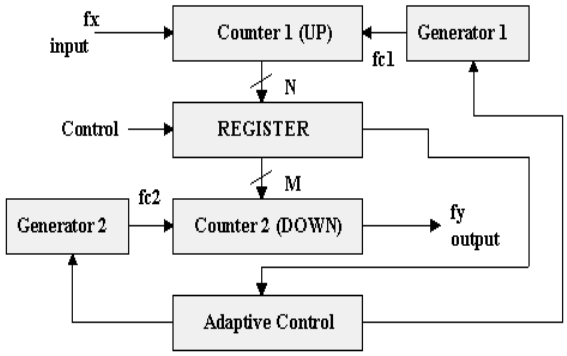

Fig. 5. Adaptive control used in the digital frequency synthesizer

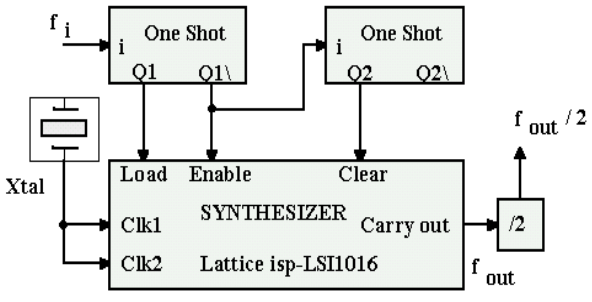

Fig. 6. The digital synthesizer realized by using Lattice isp-LSI1016 IC's, oscillator and 2 one shot devices. Clk1 and Clk2 are connected together.

Fig. 5 is the almost the same as Fig. 1, only adaptive control is added. Adaptive control block reads the contents of the Register. When the number is too small, the

frequency of Generator 1 is multiplied and also frequency in Generator 2 is multiplied, so that f_{C1}/f_{C2} = *constant.* On the other hand, if number in Register is too big, the frequencies of booth generators are divided by same number.

5. EXPERIMENTAL RESULTS

The digital synthesizer was designed and built according to the above discussion requires. It consists of one Lattice ispLsi 1016 device (in-system programmable Large Scale Integration circuit), X-tal oscillator and two peripheral one shot devices (Fig. 6).

The detailed internal block diagram of the 16-bit synthesizer is shown in Fig. 7. The connection of two, one shot devices is shown in Fig. 8. For device testing, $f_{C1}= f_{C2}$ = 31.111 MHz and k_1 =1, so according to the relation (7), the ideal output frequency is:

$$f_Y = f_X \tag{14}$$

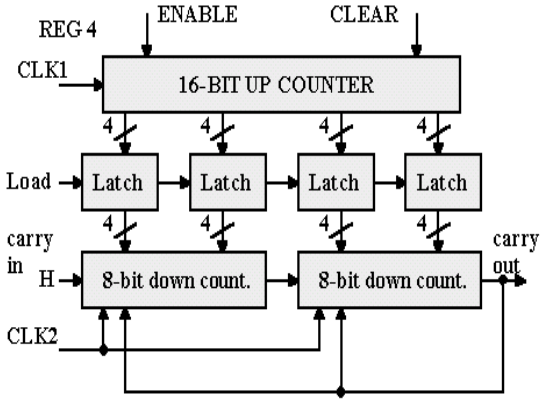

Fig. 7. The detailed block diagram of the digital frequency synthesizer based on programmable logic Lattice isp-LSI1016

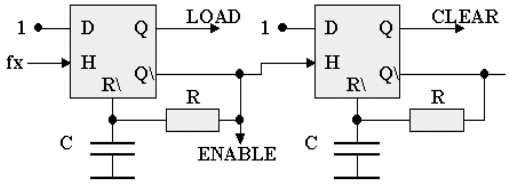

Fig. 8. Dual one shots

Photography of PC board of synthesizer is shown in Fig. 9.

Two, different delays one shot were used. For delay of 0.6 μs (load + clear) the input frequency f_x, output frequency f_y were measured and *C1* number in up-counter and *C2* number in down-counter were computed and number difference *dif* which is given by:

$$dif = C1 - C2 \tag{15}$$

was also computed. The results are shown in Table 1. From Table 1 it can be seen, that differences are constant and error in frequency can be easily corrected. For delay of 60 ns (load + clear) input and output frequencies are the same to 1 MHz. For frequency 1 MHz to 3 MHz, the results are in Table 2.

TABLE 1. Input and output frequencies for 600 ns delay (load + clear)

fx [Hz]	***fy*** [Hz]	***C1*** [-]	***C2*** [-]	***dif*** [-]
1502	1502	20713	20713	0
2010	2014	15478	15460	18
4008	4016	7762	7745	17
6004	6026	5181	5162	19
10008	10068	3108	3090	18
20004	20258	1555	1535	20
40000	40980	777	759	18
100000	106600	311	291	20

TABLE 2. Input and output frequencies for 60 ns delay (load + clear). It is important to note, that input and output frequencies are the same to 1 MHz

f_x	1082.1	2020	3275	[kHz]
f_y	1083.0	2032	3276	[kHz]

For f_{C1} = f_{C2} = 31.111 MHz, the maximal input frequency is approx. 3.5 MHz for good function. Minimal input frequency (to avoid an overflow of 16-bit counter) is 476 Hz.

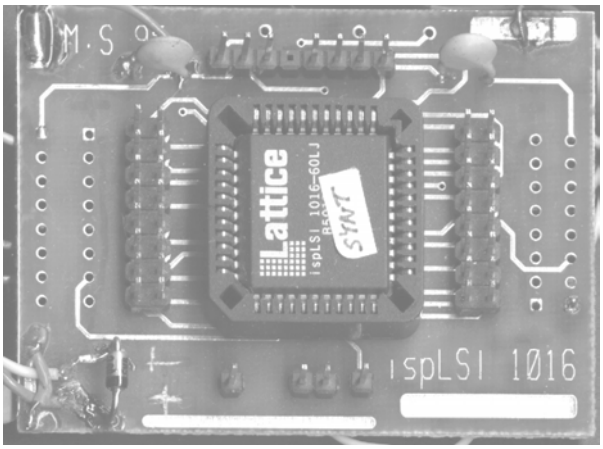

Fig. 9. PC board of realized synthesizer. PC board size: 60 x 45 mm

The digital synthesiszer, which was described in Fig. 1 was constructed and measured. has following disadvantages:

The synthesiszer has following disadvantages:

a) Accuracy depends on integer number in Counters
b) Not suitable for high output frequency
c) Square wave output

The advantages are:

a) Pure digital architecture
b) Wide range of frequency changing
c) Can be used as building block for fractional PLL frequency synthesizer
d) No setting problems

e) Stable
f) Fast response
g) Easily realized by programmable logic array
h) Adaptive control can be simply added for quality improving.

6. SUMMARY

The frequency synthesizers form, which is the basic of most radio system designs and their performance is often key to the overall operation. They are also an important building block in almost, all digital and mixed signal integrated circuits as a clock multiplier. Apart from the usual integer-N PLL implementation of the clock multiplier, where a voltage controlled oscillator is locked to a clean reference clock, architectures based on a (DLL) have been successfully used recently as a clock multipliers [9], [10]. The main disadvantage of conventional DLL's, however, is their limited phase capture range.

A new design technique of the frequency synthesizer has been presented in this paper. The presented digital frequency synthesizer was patented in the Czech Republic. Schemes for direct and indirect synthesizers were shown and basic equations and block diagram were also described. The digital frequency synthesizer was realized as 16 bit device, by using Lattice ispLsi 1016 IC's (Counter max. frequency 80 MHz), and experimental results were introduced. It is important to note, that delay, caused LOAD and CLEAR can be easily corrected. The synthesizer can be best of all realized simply by using FPGAs or another types of programmable logic. The synthesizer is suitable for fractional frequency multiply, divide or for another frequency processing. Main advantage is that synthesizer has a fully digital structure and also, there are no stability problems. Also possibilities of wide range input frequency is important. The digital synthesizer can be used with phase-locked loop for simple production of the fractional PLL. In near future, adaptive control will be add in the digital frequency synthesizer for better function on higher frequency.

ACKNOWLEDGMENT

This research work has been supported by *New Technologies - Research Centre in West Bohemian Region LN00 B084.*

REFERENCES

[1] Kroupa, V. F.: Direct Digital Frequency Synthesizers, New York, IEEE Reprint Press Book, 1998.
[2] Stork, M.: A Digital Frequency Synthesizer, Czech Republic Patent, AO 256872, October 30, 1989.
[3] Kroupa, V. F.: Theory of Phase-Locked Loops and Their Applications in Electronics (in Czech), Academia Praha 1995.
[4] Rode, Ulrich L.: Microwave and Wireless Synthesizers, Theory and Design, John Willey & Sons, ISBN 0-471-52019-5, 1997
[5] Curtin, M., O'Brien, P.: Phase-Locked Loops for High-Frequency Receivers and Transmitters - Part 1, Analog Dialogue 33-3, 1999, pp. 1-4, Part 2, Analog Dialogue 33-5, 1999, pp. 1-5, Part 3, Analog Dialogue 33-7, 1999, pp. 1-5, Analog Devices, http://www.analog.com/
[6] Danielson, D.D, Froseth, S.E.: A Synthesized Signal Source with Function Generator Capabilities, Hewlett-Packard Journal, Vol. 30, no. 1, January 1979, pp. 18-26.
[7] Chodora, J.: A Digitally Corected Fractional-N Synthesizer, Hewlett-Packard Journal, Vol. 44, no. 2, April 1993, pp. 44.
[8] Surber, J., McHugh, L.: Single-Chip Direct Digital Synthesis vs. the Analog PLL, Analog Dialogue, Vol. 30, no. 3, 1996, pp.12-13, Analog Devices, http://www.analog.com/
[9] G. Chien and P. R. Gray: A 900 MHz local oscillator using DLL-based frequency multiplier technique for PCS applications. *IEEE J. Solid - state Circuits, vol, 35.,*
[10] Remco, C.H. at all.: Low-Jitter Clock Multiplication: A Comparison Between PLLs and DLLs. IEEE Transaction on Circuits and Systems - II: Analog and Digital Signal Processing, Vol. 49, No. 8, August 2002, pp. 555-566.

Author:Milan Stork, Associate Professor, University of West Bohemia, Department of Applied Electroncs, P.O.Box 314, 30614 Plzen, Czech Republic, Tel, fax: +420377634202, e-mail: stork@kae.zcu.cz