An All Digital Fractional-N Synthesizer

Designing a high-performance frequency synthesizer is not a trivial task, especially when modern components such as direct digital synthesizers (DDS) ICs are not available. This synthesizer uses cheap off-the-shelf components, but modern all-digital fractional-N synthesis techniques, that allow it to meet or beat the performance of much more complex and costly designs.

By Oleg Skydan, UR3IQO

During the design of a home-made DSP HF transceiver, I faced the challenge of the first LO synthesizer design. I studied several solutions to improve the performance of synthesizer—to obtain small step size while keeping the other parameters state-of-the-art. A multiple-loop design was too complex and expensive for home building.[1] The hybrid DDS-driven PLL synthesizer discussed in *QST* and *QEX* in recent years[2, 3] would be a suitable solution, but I was unable to get DDS chips. So I decided to try all-digital fractional-N techniques.

[1]Notes appear on page 33.

Litvinova Str 31/8
c. Donetsk-14 Ukraine 83014
skidan@mail.ints.net

After some experiments, I have reached success with a simple all-digital, fractional-N, single-loop synthesizer with fractional spur compensation using a fourth-order sigma-delta modulator.

Fractional-N Divider Basics

Fractional-N synthesizers have been used for many years to improve the performance of indirect frequency synthesizers. Fig 1 shows the principle of a fractional-N divider. The division ratio of the divider is made to have a fractional component by changing the division ratio of the divider periodically, so the average value contains a fractional element. If, for instance, a fractional value of 0.1 is required then the division ratio is changed by one every tenth cycle. If a fractional value of 0.01 is required then the division ratio is changed by one every hundredth cycle. This offers much finer frequency control than integer-based systems.

An accumulator whose digital output is incremented for each cycle of the divider by the fractional frequency requirement is a convenient method of controlling the division ratio. The accumulator uses an adder latch to add the contents of its input to its current output on each cycle of the clock. It behaves as the digital equivalent of an integrator and since the integral of the frequency is phase, its output represents the relative phase of the fractional component. Every time the accumulator reaches its capacity, it produces an overflow, which changes the divider division ratio.

There is a price to be paid for the improvement in frequency resolution

these systems provide. The manipulation of the divider ratio generates phase perturbations, and hence, spurious signals that must be eliminated in a useful synthesizer design. The nature of the phase perturbations is predictable and can be cancelled using an analog correction system. Such a system was used in the first LO of the RA1792/RA6790 receivers. Solutions that use various techniques to improve the performance of PLL-based fractional-N systems have been patented.[4, 5] Unfortunately, the fractional-spur suppression was limited by analog circuits.

The Digital Revolution

In 1984, John Wells of Marconi Instruments invented a new solution to the problem of fractional-N synthesizers that did not require analog components to correct noise and spurious emissions normally introduced by fractional-divider schemes.[6] An implementation of such a system is shown in Fig 2.

This digital fractional-N system is based on the principle of noise shaping. Instead of trying to cancel the fractional-N spurs, this system spreads them out over a wide spectrum, and modifies the resulting noise shape so as to minimize low-frequency spectral content (see Fig 3). The technique actively reduces the generation of low-frequency noise and exchanges it for increased levels of higher-frequency noise. This is a very good arrangement because the PLL itself acts as a low-pass filter.

In the example of Fig 2, the single accumulator of the earliest forms of fractional-N synthesizers is replaced with three or more accumulators, the output of each being connected to the input of the next. The overflow from each of the accumulators manipulates the division ratio of the divider.

The first accumulator overflow acts in the same way as the accumulator in the simplest fractional-N systems. It changes the division ratio divider from N to $N + 1$ for one cycle when the accumulator overflows. The remainder output from the first accumulator represents the phase error that would result if no other correction were applied. The second accumulator digitally integrates the output of the first accumulator and subsequent accumulators repeat this process. The overflow from the second accumulator causes, and so forth.

The sequences that are generated correspond to the terms of a plurality of sequences, each of which represents successive rows in a Pascal triangle (see Fig 4). The sum of each row is zero, with the exception of the first, which is required to correct the division ratio of the divider overall to obtain the required fractional frequency. Because the second and subsequent rows of the Pascal triangles introduce an average divider change of zero, these accumulator overflows have no long-term effect on the division ratio of the divider. However, they are used to remove low-frequency components from the divider's output spectrum and to transfer the energy to higher frequencies where the PLL loop filter can successfully filter them. As shown in Fig 2, the accumulator overflows are fed via delay networks (implemented by D-type flip-flops) to the adder with weighted inputs, so that they generate the required division-ratio changes.

The ability to suppress fractional spurs depends on the initial accumulators' contents and input to the first accumulator. If we want good spur suppression, we should avoid initiation cyclic sequences in the accumulators (like 1000->0000->1000-> and such in a four-bit accumulator), which would shorten the generated sequences.[7] To ensure "longevity," the LSB of the first accumulator input is always set (only odd fractional numbers are used). The effect is the creation of a very long pseudorandom sequence, eliminating initial-condition effects with high-pass characteristics and the desired average

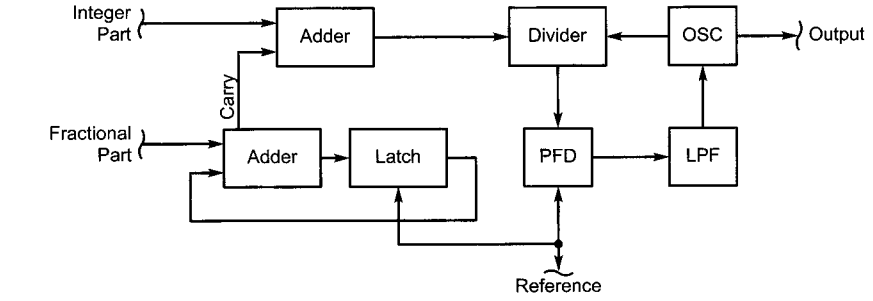


Fig 1—A block diagram of the simple (first order) fractional-N synthesizer.

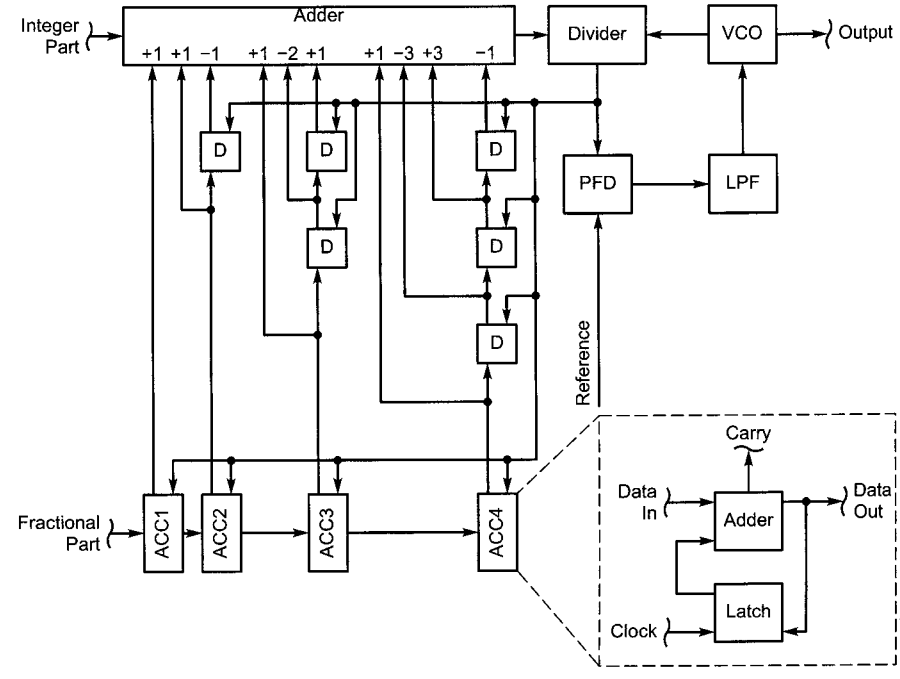


Fig 2—A block diagram of the multiple accumulator fractional-N synthesizer.

division ratio. This can create a frequency error of:

$$\frac{F_r}{2^L} \qquad \text{(Eq 1)}$$

where L is the size of the accumulator and F_r is the reference frequency.

Circuit Details

The synthesizer, shown in Figs 5 and 6, consists of the VCO, two isolation amplifiers, an output frequency divider and a PLL circuit. The VCO operates in three bands (see Table 1). Its output is divided by 4 to 14, depending of the band, before feeding it to the first mixer. This improves phase-noise performance.[8] A KP307G junction FET was selected for the oscillator circuit because of its low noise figure. The KP307G is a Russian transistor. A J310 is a suitable substitution for it. A short-circuited line W1 with an unloaded Q of 300 is used as a resonator. The two high-quality PIN-diodes D7 and D8 are used for band switching. (Unfortunately, I do not know a suitable substitution for these Russian parts.) The series-parallel combination of six Varactors is used for better phase-noise performance.

The two amplifiers Q3 and Q4 isolate the output divider and PLL circuit from the VCO. I used a BF998 with a very low reverse-transfer capacitance.

The U7 divides the output of the synthesizer by any value from 1 through 9. Only division ratios from 2 through 7 are used in my transceiver, with an additional divide-by-two stage (74AC74) located on the first mixer board. Thus, the overall division ratio varies among even values from 4 through 14, to obtain square waves for the mixer. A bit unusual: The divider schematic allows avoiding of the additional inverter use.

The heart of the fractional-N synthesizer consists of two ICs (see Figs 5 and 6). The complex algorithm of fractional-N divider and interface with the control processor (via standard SPI bus) are realized by microcontroller U5, while fast logic functions are realized by complex programmable logic device (CPLD) U4. This makes the design simple and low-cost, but the microcontroller's performance (8 MIPS) limits reference frequency to 150 kHz (this

Table 1

VCO Band	*VCO Frequency (MHz)*
1	90.0-98.8
2	85.0-88.0
3	78.3-79.6

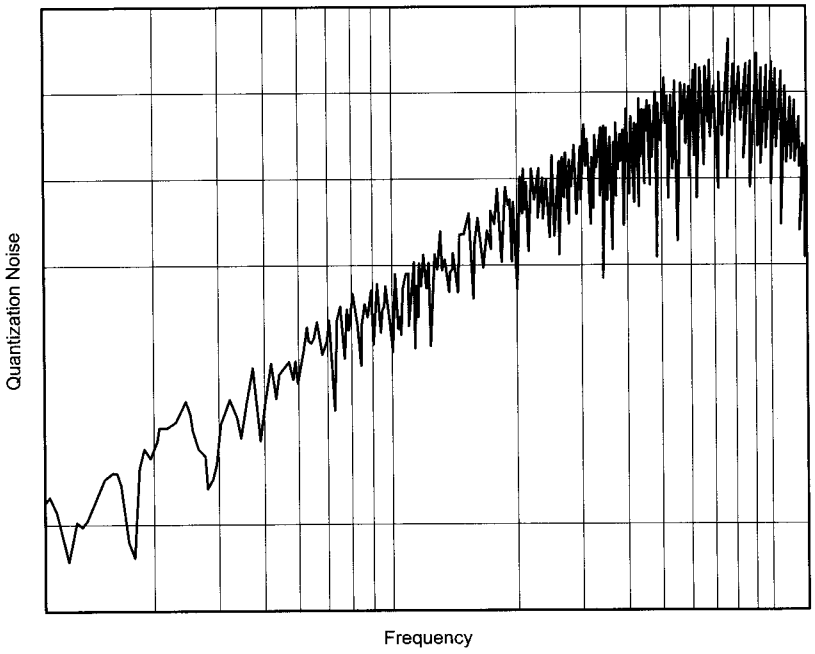


Fig 3—Typical output spectrum shape of the multiple accumulator fractional-N divider.

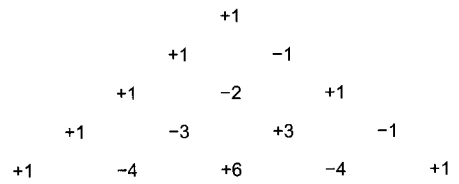


Fig 4—Pascal's triangle used for division-ratio calculating.

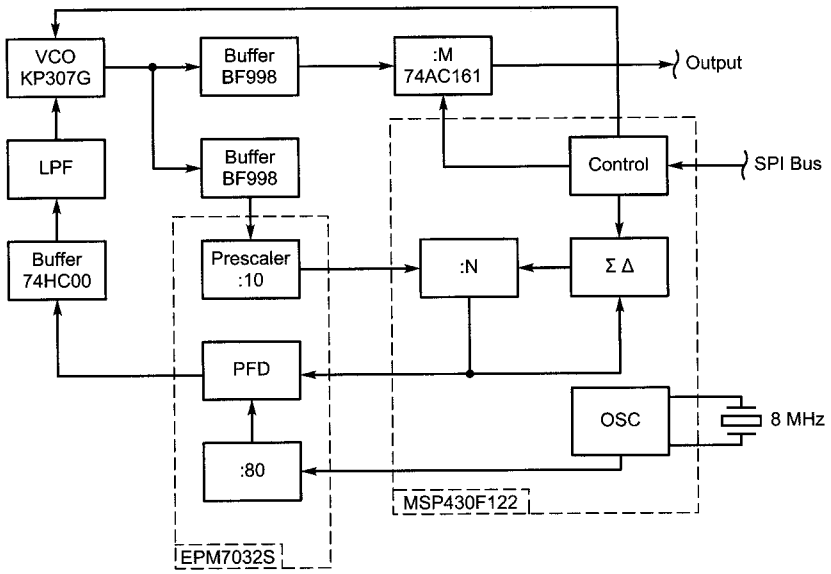
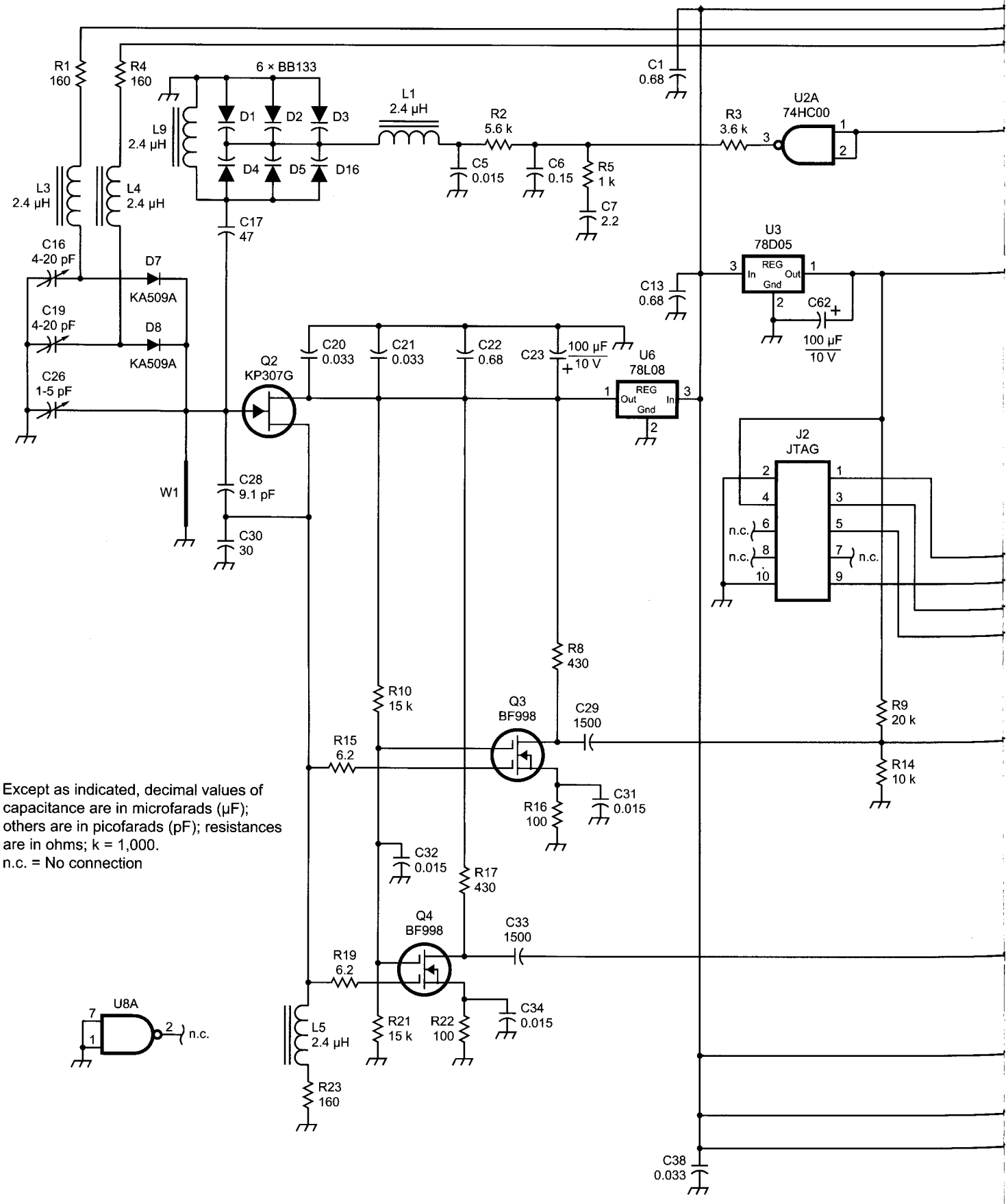
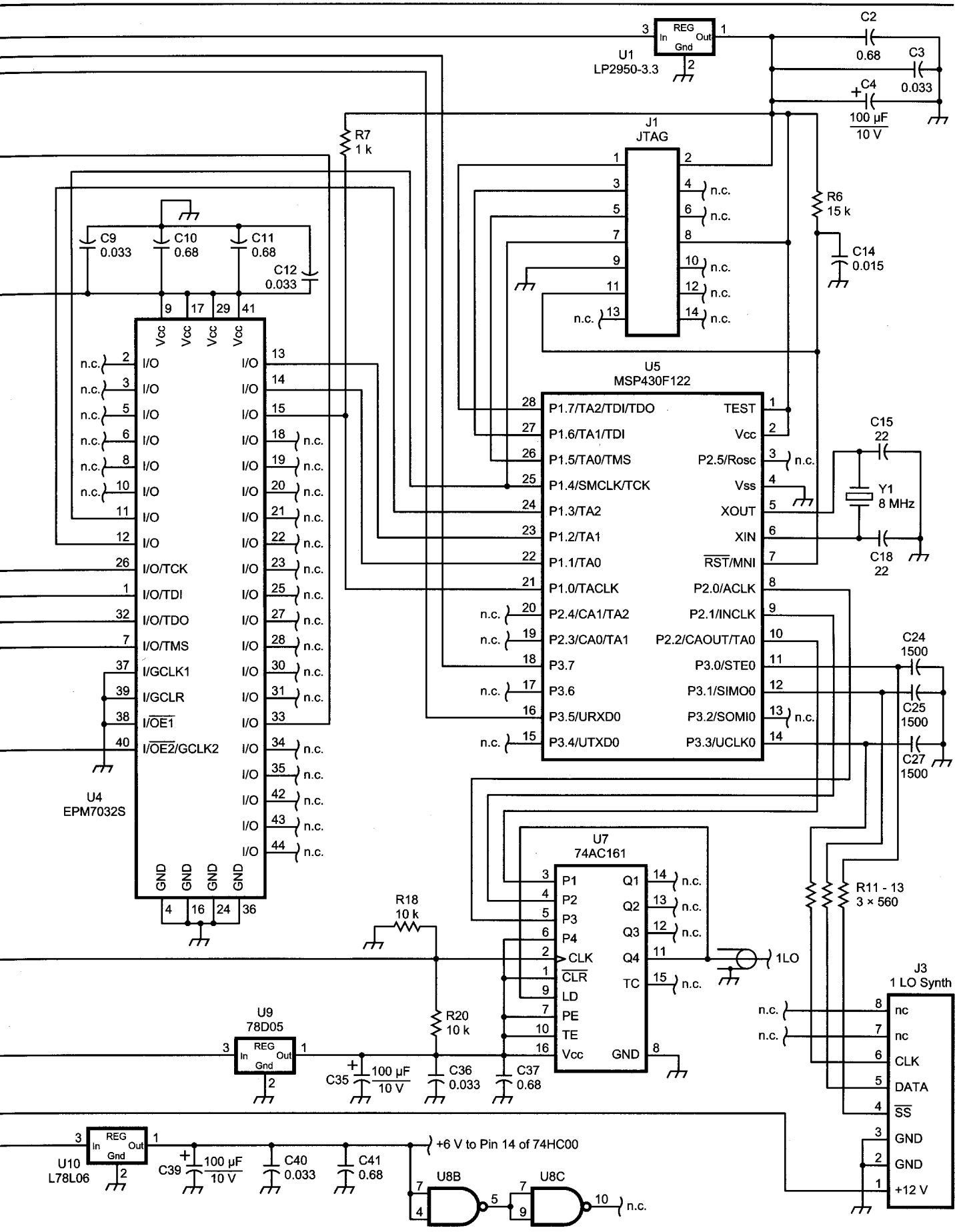


Fig 5—The synthesizer's block diagram.



Except as indicated, decimal values of capacitance are in microfarads (µF); others are in picofarads (pF); resistances are in ohms; k = 1,000.
n.c. = No connection

Fig 6—The synthesizer's schematic diagram.


U1
LP2950-3.3
REG
In
Out
Gnd
C2
0.68
C3
0.033
C4
100 µF
10 V
R7
1 k
J1
JTAG
R6
15 k
C14
0.015
C9
0.033
C10
0.68
C11
0.68
C12
0.033
U5
MSP430F122
P1.7/TA2/TDI/TDO
P1.6/TA1/TDI
P1.5/TA0/TMS
P1.4/SMCLK/TCK
P1.3/TA2
P1.2/TA1
P1.1/TA0
P1.0/TACLK
P2.4/CA1/TA2
P2.3/CA0/TA1
P3.7
P3.6
P3.5/URXD0
P3.4/UTXD0
TEST
Vcc
P2.5/Rosc
Vss
XOUT
XIN
RST/MNI
P2.0/ACLK
P2.1/INCLK
P2.2/CAOUT/TA0
P3.0/STE0
P3.1/SIMO0
P3.2/SOMI0
P3.3/UCLK0
C15
22
Y1
8 MHz
C18
22
C24
1500
C25
1500
C27
1500
U4
EPM7032S
I/O/TCK
I/O/TDI
I/O/TDO
I/O/TMS
I/GCLK1
I/GCLR
I/OE1
I/OE2/GCLK2
U7
74AC161
P1
P2
P3
P4
CLK
CLR
LD
PE
TE
Vcc
GND
Q1
Q2
Q3
Q4
TC
1LO
R18
10 k
R11 - 13
3 × 560
J3
1 LO Synth
nc
CLK
DATA
SS
GND
+12 V
U9
78D05
R20
10 k
C35
100 µF
10 V
C36
0.033
C37
0.68
U10
L78L06
C39
100 µF
10 V
C40
0.033
C41
0.68
+6 V to Pin 14 of 74HC00
U8B
U8C

design uses 100 kHz).

The CPLD U4 is the smallest device in EPM7000S family. It is configured to contain a prescaler, a reference divider and a dead-zone-free phase frequency detector (PFD). The schematic of the CPLD is shown in Figs 7 and 8. Those diagrams were designed with the E+MAX CAD tools.[9]

The voltage divider (R9, R14) is used to set up the necessary bias for EPM7032S clock input. It has LVCMOS compatible inputs, so it needs approximately 1.7 V.

The prescaler is built using a Johnson counter to satisfy the 100-MHz operating frequency requirements and to obtain square-wave output. Five

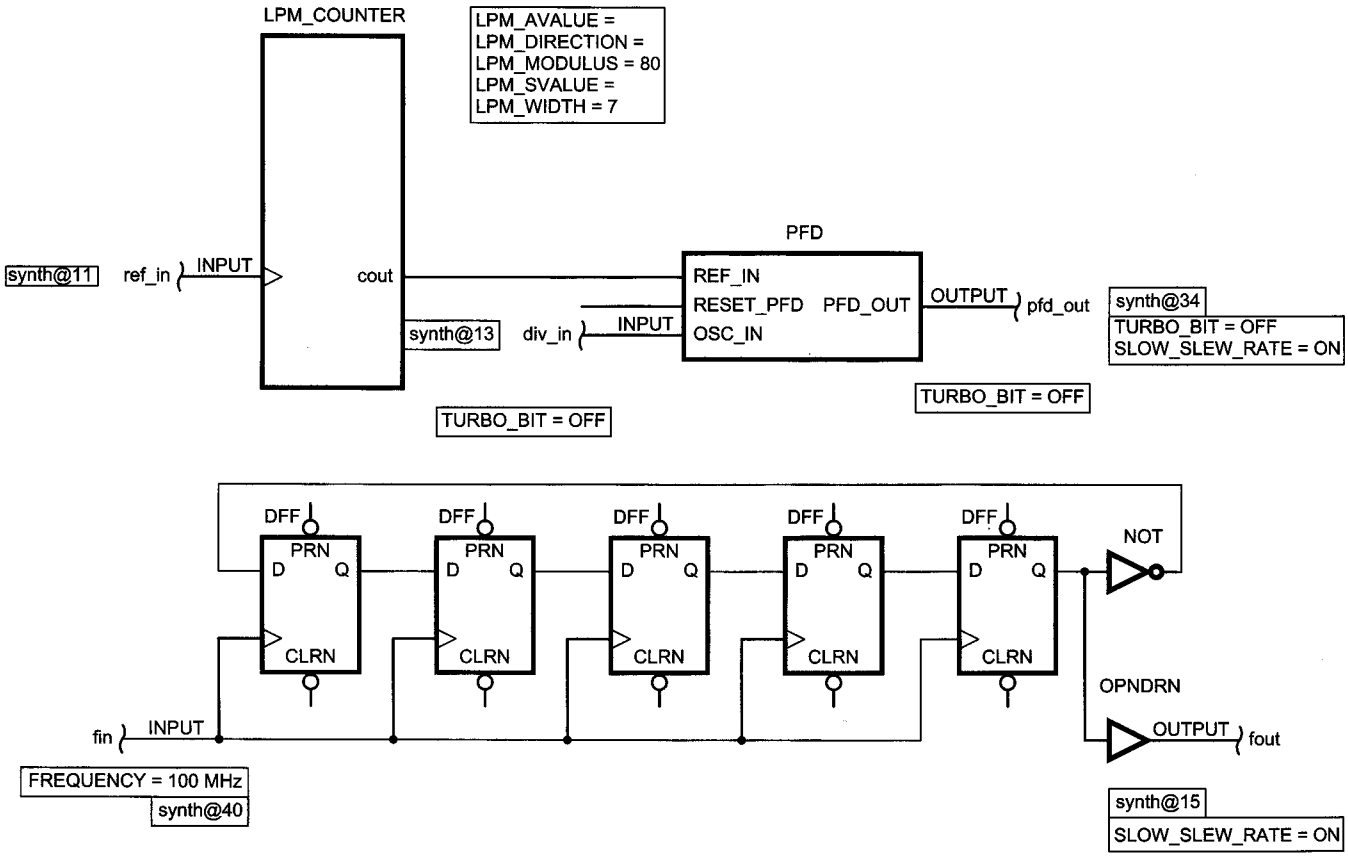


Fig 7—A CPLD functional diagram.

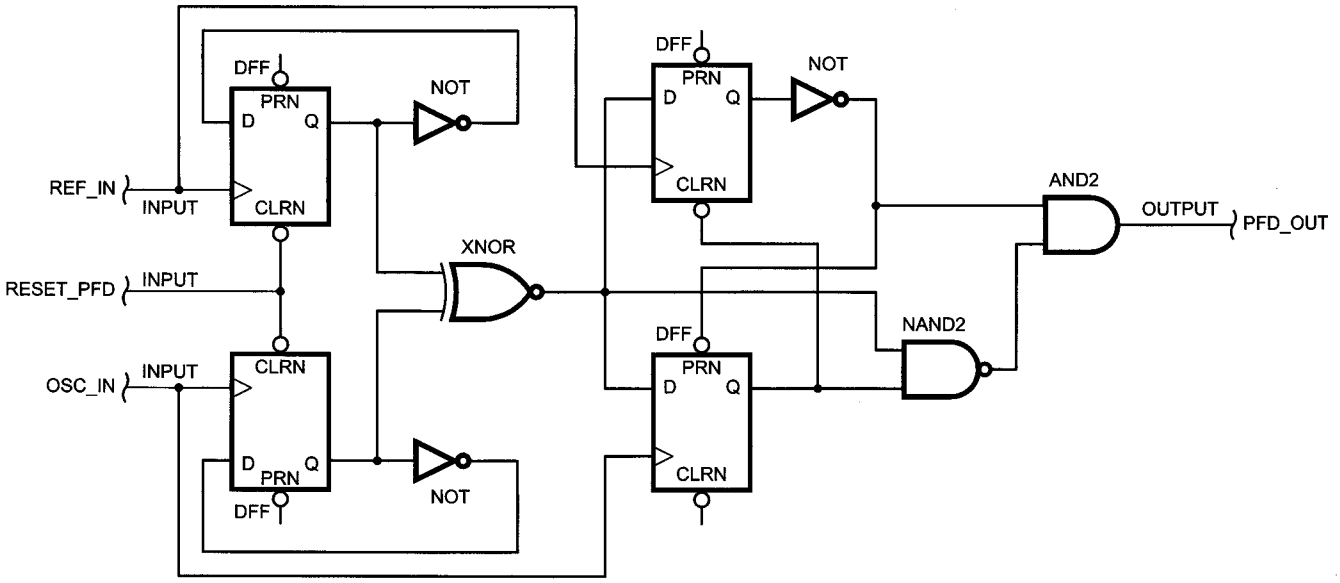


Fig 8—Functional diagram of the dead-zone free PFD implemented in the CPLD.

stages are used for division by 10. The output is fed to the fractional-N divider, implemented in the U5 microcontroller's software. Since U5 is a 3.3 V device, the open-drain output and 1 kΩ pull-up resistor connected to 3.3 V are used.

The reference divider operates at 8 MHz and divides the input frequency by 80 to get a 100-kHz reference, which is fed to the PFD reference input. The microcontroller's oscillator is used as the reference oscillator. To improve its frequency stability, a high-quality vacuum crystal is used here (see Fig 9).

The PFD design is a very important part of the fractional-N synthesizer. The traditional dual D-type flip-flop (DFF) PFD will fail here. As mentioned before, the noise at the output of the fractional divider has a high-pass shape; if we want to have the same noise at the VCO output, we need to have a highly linear PFD. Otherwise, high-frequency noise will be intermodulated down to the low-frequency spectrum. The quad DFF and XOR gate design gives us such an opportunity.

The PFD is composed of two sections (see Fig 8). The first section is the phase detector composed of a dual DFF and the XOR gate, which is activated when the two signals to be compared are close in frequency. The second section is a frequency discriminator composed of a dual DFF and two NAND gates. It overrides the phase-detector section when the inputs have two frequencies far from each other to drive the oscillator frequency toward the reference frequency and put it within range of the phase detector. The detailed description of such a design can be found in the AD9901 datasheet.

The output of the EPM7032S has a 3.8 V logic-one level. This prevents us from connecting it directly to the loop filter. So, I used a 74HC00 gate as an amplifier. It operates at the highest permissible supply voltage (6 V), so I have a 1 to 5.6 V swing at the Varactors. One could say that it would be better to use an op-amp integrator with higher supply voltage here, but this design is simple and does not suffer from the noise and non-linearity of an op amp. By the way, all models of IC-746PRO/756PRO use passive loop filters and even smaller tuning-voltage ranges.

The synthesizer uses a fourth-order, passive loop filter. I have calculated initial values using *MathCAD* and tweaked the values a bit during phase-noise measurements. The corner frequency was chosen to be 400 Hz. The choice was directed by high-frequency noise suppression requirements. The

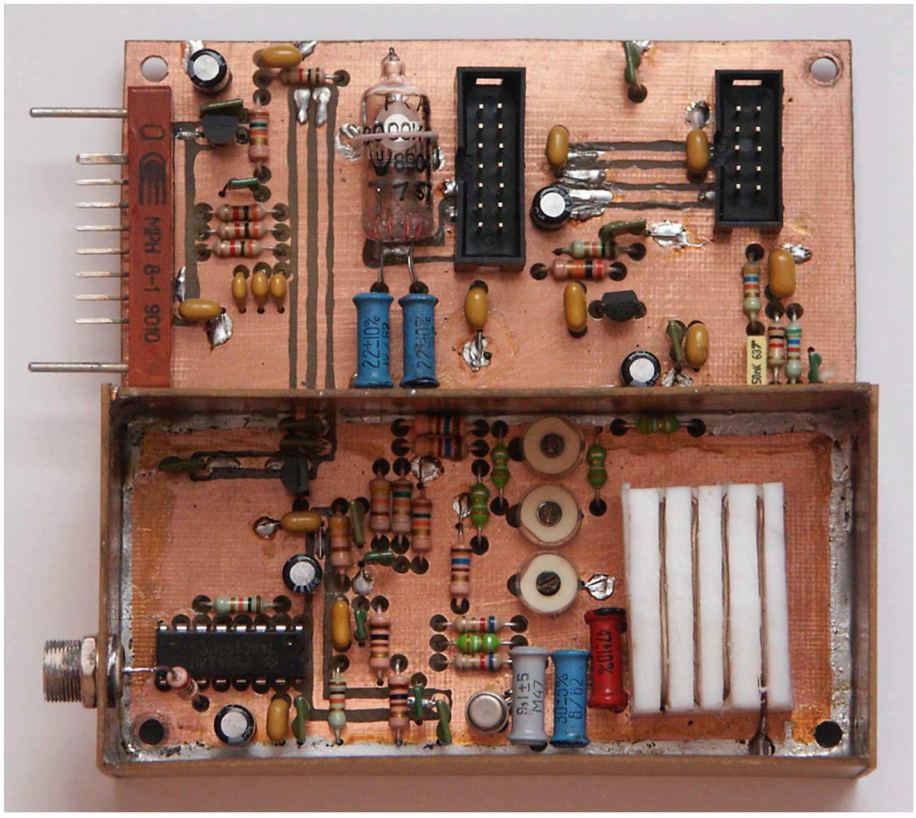


Fig 9—A top view of the synthesizer PC board.

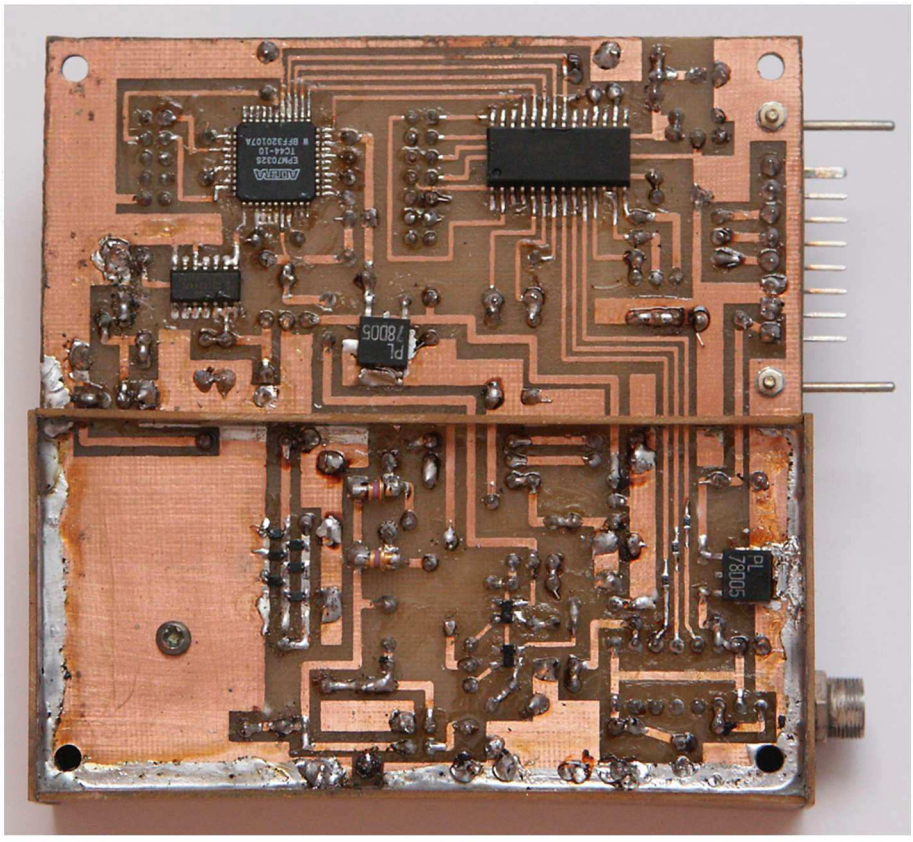


Fig 10—A bottom view of the synthesizer PC board.

Byte number	1	2	3	4	5	6
Description	N lowest byte	N second byte	N third byte	N fourth byte	N highest byte	Control byte

Bit number	7	6	5	4	3	2	1	0
Description	VCO1	0	VCO2	0	0	D2	D1	D0

Fig 11—The control-word format.

simulation of the switching time showed less than 8 ms switching time.

The low-power 16-bit microcontroller, U5, acts as fractional divider and control unit. The software uses the timer module as a divider.[10] When the timer generates a pulse at the output, its interrupt procedure is invoked to calculate the division ratio for the next cycle. This value is loaded into the timer compare register for the next pulse generation. This is repeated 100,000 times per second.

I initially used three 25-bit accumulators (with the 25th bit set to one); but after assembly and testing the synthesizer noise performance, I found some spurs, especially close in. So I decided to try four 32-bit accumulators—they were just what I needed! Fortunately, I need only to make minor changes to software and upload it to microcontroller to try different fractional-N divider configurations.

U5 is also used to control VCO band switching by injecting a 15 mA current through the PIN diodes, setting division ratio of the output divider, U7, and control synthesizer via the SPI bus. The format of the control word is shown in Fig 11.

The next formula gives the relationships of value for the divider (N), output-division ratio (M), output-divider control bits (D) and output frequency (F_{out})[11]:

$$M = 9 - D \qquad \text{(Eq 2)}$$

$$F_{out} = 10 \bullet F_{ref} \bullet \frac{256N+1}{2^{32} \bullet M} \approx 10 \cdot F_{ref} \bullet \frac{N}{2^{24} \bullet M} \qquad \text{(Eq 3)}$$

$$N = \frac{2^{24} \bullet M \bullet F_{out}}{10 \bullet F_{ref}} \qquad \text{(Eq 4)}$$

I think that $\frac{10 \bullet 10^5}{2^{24}} \approx 0.06Hz$ VCO step size is more than sufficient, so I decided to discard the low byte of the 32-bit first accumulator input word and set it to 00000001b to ensure longevity (see above). Notice that this value is divided at least by four in the worst-case of my design. The synthesizer uses five voltage regulators to satisfy ICs power supply requirements and get the necessary decoupling between different synthesizer blocks.

Construction

The synthesizer is made on a 95 mm by 95 mm homemade PC board (see Figs 9 and 10). The VCO, isolating amplifiers and output divider are shielded from the other circuitry. The VCO resonator is made from a piece of Teflon (used for mechanical stability) and silver-plated wire (0.5 mm diameter).

Results

The phase noise of the synthesizer was measured using a signal generator (the manufacturer claims it has

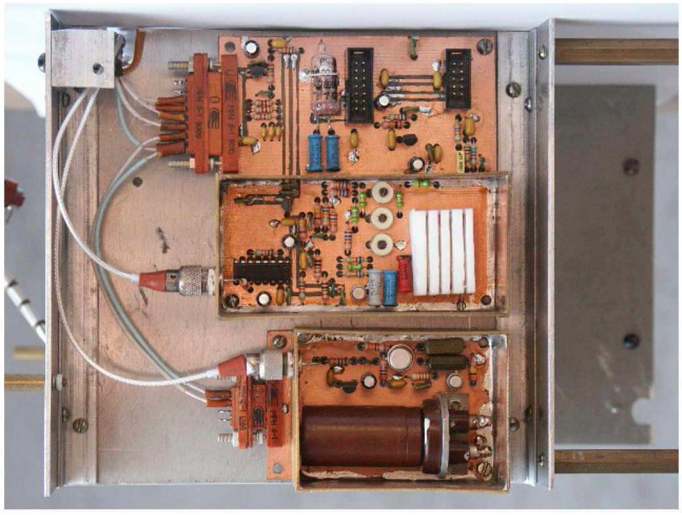


Fig 12—The synthesizer within the transceiver under construction.

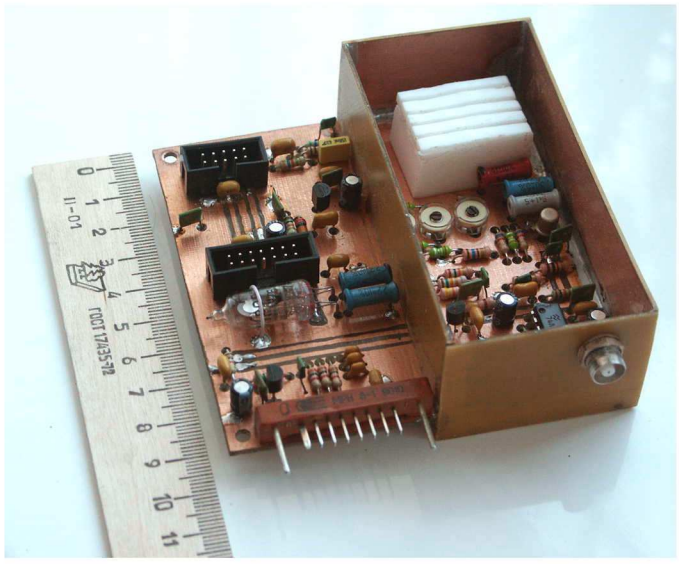


Fig 13—The synthesizer is about 9 cm square.

–140 dBc/Hz phase-noise performance at 100 kHz offset) followed by a two-pole crystal filter. The synthesizer was installed in my homemade DSP transceiver. The test signal was applied to the antenna jack and the DSP block was used

to measure the noise power in 1000-Hz or 2400-Hz bandwidths (the DSP software is capable of precise power measurements) at the different offsets. Then I corrected results by $-10\log_{10}$ (*Bandwidth*) to get results in dBc/Hz. The results are shown in Table 2. These numbers were obtained with the division by 10 at the output of the synthesizer at 20 m, so I expect degradation of the phase noise by 8 dB at 10 m and improvement by 2.9 dB at 160 m.

I have no spectrum analyzer, but I could not find any significant spurs receiving a test signal. At this time, this synthesizer is installed in my transceiver. I have found superior stability and a much simpler tune-up procedure in comparison with the previous design based on a single-loop synthesizer with interpolation provided by "pulling" the master crystal (a system like the one in the Elecraft K2).[12]

Table 2

Offset (kHz)	*Noise (dBc/Hz)*
2	–120
5	–128
10	–133
20	–136
30	–139
40	–142

Future Work

I have found an interesting patent in the US patent database.[13] It is possible to construct a frequency synthesizer with the reference multiplication factor of 1 using that patent and the fractional-N technique described in this article. So, if we used a low-noise reference oscillator, highly linear PFD and wide loop bandwidth, we would be able to get very good VCO noise suppression. I think that it is possible to build a super synthesizer using such a design. Of course it will not be as simple as the one I have described. It will need some additional analog circuitry and fast, high-density programmable logic ICs. Currently, I am satisfied with the described design and will concentrate on finishing the other software and hardware of my DSP transceiver. You can view the progress of my transceiver at **users.ints.net/skidan/T03DSP**.

Conclusions

I have presented a simple all-digital fractional-N synthesizer. Despite its simplicity and low cost, it has satisfied the first-LO requirements for my HF DSP transceiver. The synthesizer contains no analog circuits, except the VCO, so it is very easy to build and tune up (actually you need just to set up the necessary VCO bands by tuning C26, C16 and C19).

I expect digital fractional-N synthesizers to be widely adopted in amateur and commercial equipment. They offer some advantages over widely used direct digital synthesizers because of their manufacturability, high resolution and the predictability of the resulting noise.

Notes

[1]V. Manassewich, *Frequency Synthesizers Theory and Design* (New York: John Wiley and Sons, 1976).

[2]U. Rohde, KA2WEU, "A High-Performance Hybrid Frequency Synthesizer," *QST* Mar 1995, pp 30-38.

[3]C. Drentea, "Beyond Fractional-N," *QEX* Mar/Apr 2001, pp 18-25; May/Jun 2001, pp 3-9.

[4] O. Golovin, "Professionalnie radiopriemnie ustrojstva dekametrovogo diapazona" (Professional receivers for decameter band) in Russian, Moscow, USSR, 1985 (pp 280-282)

[5]R. Cox, Hewlett-Packard Company, "Frequency Synthesizer," US Patent 3,976, 945, 24, Aug 1976.

[6]J. Wells, Marconi Instruments, "Frequency Synthesizers," US Patent 4,609,881, 2 Sep 1986.

[7]C. Hill, "All Digital Fractional-N Synthesizer for High Resolution Phase Locked Loops. Part 2," *Applied Microwave & Wireless* Jan/Feb 1998, pp 38-42.

[8]V. Drozdov, "Lyubitel'skie KV Transiveri" (Amateur HF Transceivers), in Russian, Moscow, USSR, 1988, pp 26-31.

[9]E+MAX and other CAD tools for MAX7000S are available from Altera Web-site (**www.altera.com**) free of charge. Also you can find there an easy to build ByteBlaster download cable.

[10]The software is written in GNU assembler and compiled using GCC compiler. GCC compiler for the msp430 microcontroller series is available free of charge from (**mspgcc.sourceforge.net**).

[11]In my transceiver, the synthesizer signal is additionally divided by two on the mixer board. Here I give formulas for synthesizer board output.

[12]You can learn more about Elecraft K2 transceiver (and download manuals with full schematics) on the Elecraft Web-site (**www.elecraft.com**).

[13]Alexander Roth, Rohde & Schwarz GmbH & Co KG, "Frequency Synthesizer Operating According to the Principle of Fractional Frequency Synthesis," US Patent 5,847,615, 8 Dec 1998. □□