# HDL based Simulation of digital RF Frequency Synthesizers

Andreas Neyer
RWTH Aachen
Integrated Analog Circuits
Sommerfeldstrasse 24
52074 Aachen, Germany
Neyer@ias.rwth-aachen.de

Sören Sappok
RWTH Aachen
Integrated Analog Circuits
Sommerfeldstrasse 24
52074 Aachen, Germany
Sappok@ias.rwth-aachen.de

Stefan Heinen
RWTH Aachen
Integrated Analog Circuits
Sommerfeldstrasse 24
52074 Aachen, Germany
mailbox@ias.rwth-aachen.de

## ABSTRACT

This paper proposes modelling and simulation requirements for mixed-signal RF transceivers enabling a straight forward top-down design process. A fully digital RF frequency synthesizer for GSM is presented. The modelling, simulation and evaluation process will be illustrated. The paper presents the limits of current development tools and gives requests for future enhanced modelling and simulation tools for the RF design industry. The rms phase error of the investigated architecture has been measured to be below 1.4° and the spectral margin can be maintained to be more than 8 $dB$.

## 1. INTRODUCTION

The development of CMOS processes is going on for decades and hasn't finished yet. A feature size of 90 $nm$ is state of the art and 65 $nm$ will soon be reached. Together with this miniaturization more and more digital circuitry and functionality has been assembled on a single die. The integration of application and baseband processors as well as analog RF circuitry is the goal. The miniaturization of analog circuits has reached its boundaries, as the needs of analog circuits are contrarily to deep-submicrometer CMOS processes. The very low voltage headroom and the increased sensitivity to power supply and substrate noise make analog circuits often incompatible to deep-submicrometer CMOS processes. Though cost and reduction demand both analog and digitals circuits in a single chip. Thus new approaches, which favour the integration of analog functionality in deep-submicrometer CMOS processes, are necessary. For instance Figure 1 shows a typical mixed-signal phase-locked-loop (PLL) architecture in which only the multi modulus divider (MMD) and the phase frequency detector (PFD) can be considered as digital components [2]. A digital intensive implementation of a RF frequency synthesizer, in which only the oscillator remains as an analog component, has been presented in [11] and will be further studied in this paper.

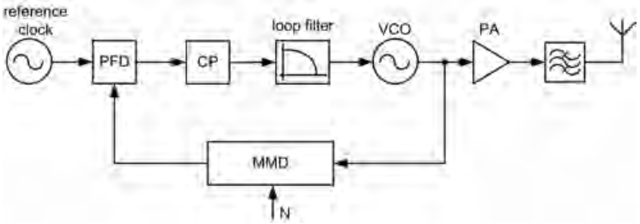

**Figure. 1. Typical mixed-signal PLL architecture.**

The circuit level is not the only area in which new basic approaches have to be found. Considering the cost of a modern mask set in a deep-submicrometer CMOS process, redesigns have to be avoided by an improved design process and the IP reuse has to be increased through new modelling techniques. To achieve best circuit behaviour modelling, the requirements on development tools are increasing together with the complexity of the chip. New modelling and simulation techniques and tools are required, which support not only the design of single blocks but also the functional and connectivity verification of an entire transceiver chip [6].

This paper will point out some new approaches and possible solutions in mixed-signal RF circuit design. It starts with the description of the investigated RF frequency synthesizer architecture in Section 2. The modelling approach, problems during the modelling process and requirements for future development tools are listed in Section 3. A description of the evaluation process as well as simulation results and analysis is presented in Section 4 followed by a conclusion in Section 5.

## 2. SYNTHESIZER ARCHITECTURE

The investigated RF frequency synthesizer architecture is shown in Figure 2. The inner PLL works as follows: A accumulator sums up a discrete reference phase. Its input is the ratio of the wanted channel frequency $f_{\mathrm{ch}}$ and the reference frequency $f_{\mathrm{ref}}$, which corresponds to the divide value $N = \frac{f_{\mathrm{ch}}}{f_{\mathrm{ref}}}$. In the feedback path the oscillator phase accumulator calculates the oscillator phase by counting all rising edges of the oscillator output signal. The oscillator phase

0-7803-9352-X/05/$20.00 © 2005 IEEE.

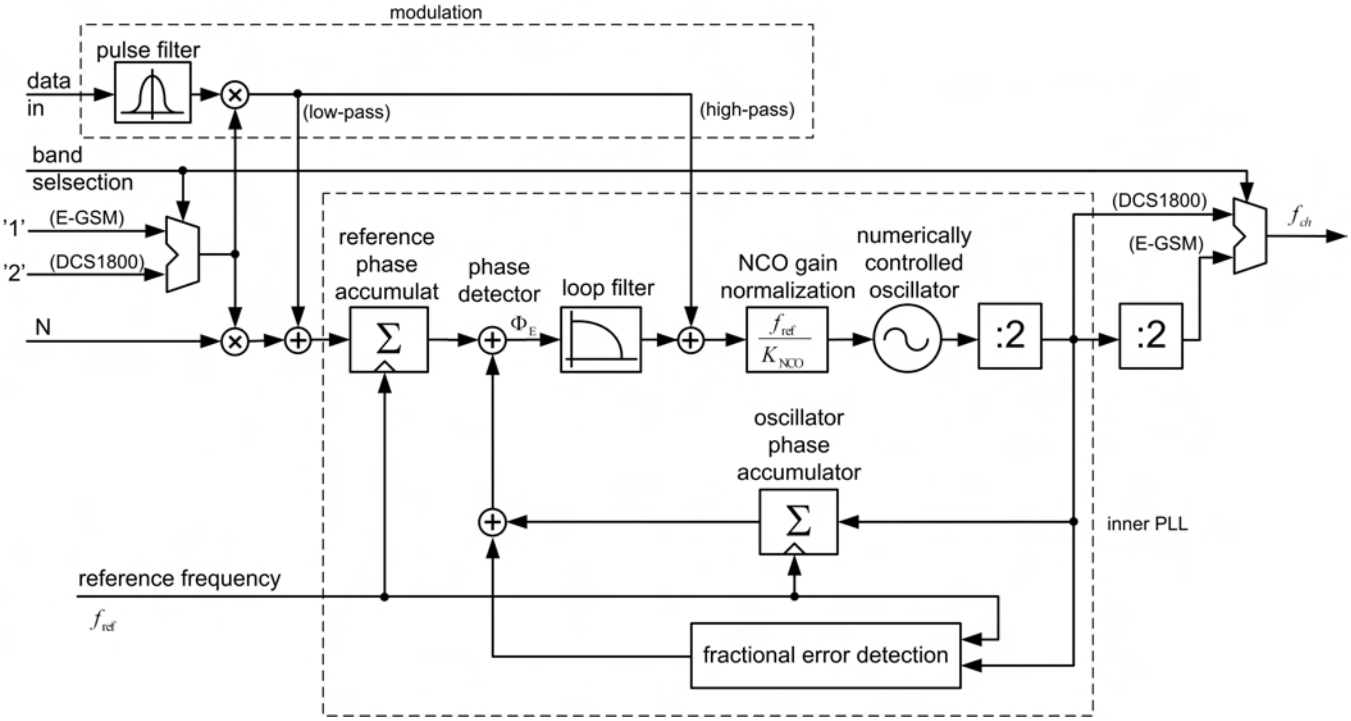

**Figure. 2. RF frequency synthesizer architecture with band switching.**

determined in that way is limited to a phase resolution of $2\pi$ (one oscillator clock cycle). To increase the phase resolution an additional fractional phase measurement circuit in the feedback path is necessary, which consists of a time-to-digital converter [10]. Knowing that the reference phase and the oscillator phase are represented by a digital word, the phase detector (PD) simplifies to an adder. Accordingly the phase error $\Phi_E$ is presented as a digital word and the loop filter can be implemented digitally. Before the filtered phase error is assigned to the oscillator input, it has to be normalized to the corresponding oscillator gain $K_{NCO}$ at the given channel frequency. Because all phase measurements are related to reference phase, the filter output has additionally to be multiplied by $f_{ref}$. Furthermore it has to be multiplied by two, because it is advantageous to have the oscillator running at double the channel frequency. That way it is easy to generate inphase and quadrature phase components at the channel frequency by dividing the oscillator output by two. Since the oscillator is controlled by a digital control word, a numerically controlled oscillator (NCO) is used instead of a classical voltage controlled oscillator (VCO).

To use the PLL for modulation, a modulation path has to be added. The modulation works as follows: The data is shaped and normalized by a pulse shaping filter and inserted at two points of the loop. One insertion point is in front of the reference phase accumulator. It is called the low-pass modulation point, because of the low-pass characteristic of the loop, which limits the available modulation bandwidth and data rate. To overcome this bandwidth limitations another modulation feed called direct feed at a high-pass modulation point is necessary. The input of the normalization circuit can be used here. A direct modulation exclusively at the high-pass point is not possible, because the loop would identify the modulation as a disturbance and try to control it out in the long term.

The synthesizer system was designed for a GSM system using the E-GSM and the DCS1800 band, hence a band switching capability has to be implemented. It is the best way to have the loop running in a fixed frequency range and do the band switching outside the loop. The PLL works in the DCS1800 band. The E-GSM band is covered by multiplying the channel frequency and the pulse shaping filter output by two and inserting an additional divide-by-two frequency divider at the loop output. The desired band can be selected easily now.

This almost full digital PLL architecture except for the analog oscillator is quite appropriate for the integration of RF circuits together with digital application and baseband processors in a single chip using deep-submicrometer CMOS technologies.

To get a better understanding about the PLL and its control properties an examination of the equivalent Laplace domain model as shown in Figure 3 is essential. The implemented filter structure can also be taken from the Laplace domain model [10].

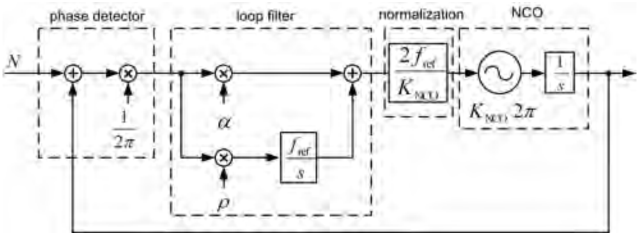

**Figure. 3. Modell of the frequency synthesizer in the s-domain.**

Starting from this model the closed-loop PLL transfer function can be determined to

$$H_{cl}(s) = N\frac{s + f_{\mathrm{ref}}\alpha + f_{\mathrm{ref}}^2\rho}{s^2 + sf_{\mathrm{ref}}\alpha + f_{\mathrm{ref}}^2\rho}. \quad (1)$$

As the damping factor $\zeta$ and the natural frequency $\omega_{\mathrm{n}}$ are defined as follows [9]:

$$\zeta = \frac{\alpha f_{\mathrm{ref}}}{2\omega_{\mathrm{n}}} = \frac{1}{2}\frac{\alpha}{\sqrt{\rho}}, \quad (2)$$

$$\omega_{\mathrm{n}} = 2\pi f_{\mathrm{n}} = \sqrt{\rho}f_{\mathrm{ref}}, \quad (3)$$

equation (1) describes a classical two-pole system

$$H_{cl}(s) = N\frac{2\zeta\omega_{\mathrm{n}}s + \omega_{\mathrm{n}}^2}{s^2 + 2\zeta\omega_{\mathrm{n}}s + \omega_{\mathrm{n}}^2}. \quad (4)$$

The PLL is called a type-II PLL because of its two poles in the origin at $s = 0$. One pole is located in the oscillator and the other in the loop filter.

In practice a high loop bandwidth is desired to achieve a low settling time. But on the other hand a low bandwidth is required to decrease the phase noise. Therefore a mechanism for bandwidth switching is necessary, which would cause an additional settling process. This is not acceptable, when a low bandwidth and a fast locking process are needed. Due to the zero steady state phase error of the integrating loop filter, the bandwidth can be changed without causing a further settling process.

## 3. MODELLING AND SIMULATION

Currently RF engineers typically use a bottom-up design flow. They start creating schematics and focus on noise figures and gain values in their design process. The existing gap between system concept engineering and transistor level design is commonly closed by spreadsheets, which contain simulated requirements and additional specifications. Therefore a more efficient design process is needed in the RF design industry.

The above described PLL has been implemented using the VHDL hardware description language. It is suitable for top-down design, starting from a parametric system level description and down to implementation level. VHDL allows the use of high level behavioural models, structural models and hardware related synthesisable descriptions. Furthermore the event-driven simulation affords a high simulation speed [8]. VHDL is supported by many simulation tools and an extension to the pure digital VHDL hardware description language like VHDL-AMS, which also supports analog modelling, is available. Most simulators even support the use of several description languages within a single simulation e.g. VHDL and Verilog.

The first PLL model has been implemented using ModelSim from Mentor Graphics. In a second step the PLL model was transferred into the Cadence Analog/Mixed-Signal (AMS) Design Environment. The goal is to setup a test bench for RF transceiver simulation based on schematic driven mixed-signal design [7] and to be able to easily switch between different cellviews. Doing so a loss of simulation speed could be observed. For simulating 2 $\mu$s including noise respectively jitter simulation a standard personal computer consisting of an AMD Athlon 64 3200+ CPU and 1.5 GB RAM takes approximately 12 min. using ModelSim. As the PLL only exhibits few noise sources because of its digital manner it is sufficient to model the oscillator noise, which can in this case be modelled as digital jitter [5]. The same simulation on the same PC using the same model takes about 2 h (a factor of ten longer) using the Cadence AMS Environment or ADVance MS the analog/mixed-signal simulator from Mentor, which can be used stand-alone or out of the Cadence design environment. The cause of the significant lower simulation speed could not be determined yet. However, the simulation is still times faster than a simulation, which uses the AMS extension of VHDL. This is caused by the long transient analysis and the small minimum step width due to the required resolution in the RF domain.

The schematic driven layout using Cadence Virtuoso allows an easy switching between different cellviews, which may implement different architectures on different levels of abstraction (e.g. behavioral, structural and transistor level models). Therefore the schematic driven AMS design approach using hardware description languages among other techniques is suitable for system and circuit designers allowing a persistent uniform top-down design process. Difficulties occur when not only the level of abstraction is switched but also the domain in which the simulation is placed, e.g. baseband or passband simulation [3]. This is necessary to decrease the computational effort and to facilitate the simulation of entire chips, which is essential to reduce the development costs, redesigns and test runs and thus the time to market. Additionally the IP reuse is simplified and increased. Therefore state of the art developments tools have to support not only the implementation and functional verification but also connectivity verification.

Another problem concerns the waveform viewer of current simulation environments. Tools which target the design of digital systems frequently lack the ability to properly display value-continuous signals. Whereas current mixed-signal environments often aren't as advanced as existing analog environments. They usually don't afford adequate analysis options (for example spectrum analyzer and eyediagram plotting) or the waveform export into a common file format. Therefore a sample and export process within simulated system and an additional external tool like Matlab from MathWorks for evaluation are often necessary (see Section 4.1.).

### 3.1. Requirements for HDLs

The request for future developments in the EDA industry is to provide transistor level simulation as well as complex baseband signal processing. Therefore the support and integration of hardware description languages (HDLs) like VHDL-AMS or System Verilog has to be improved. These new languages allow new top-level constructs like 'interfaces' and 'structs' (System Verilog) or 'records' (VHDL), which are not yet supported by design tools. For example Cadence uses Verilog-AMS as netlist format and ADVance MS uses spice in mixed-signal simulations. VHDL-AMS would be preferable, because it supports analog and digital description whereas System Verilog does not support analog design. A kind of 'System Verilog-AMS' or connectivity to analog simulators would be desirable to provide a complete flow.

With these new features everybody from system to circuit designer could use the same development environment and even work with the same schematics on different levels of abstraction with various cellviews. The system design, implementation, functional and connectivity verification are then combined in a single environment. A suggestion of how such a description could look like is given by the example of the NCO, whose interface can be seen in the simplified equivalent network (Figure 4). A possible interface description using VHDL respectively System Verilog is shown in Listing 1 and 2. All required signals and information are combined in a single data type (record or struct). This data type can then be used on schematic level to provide each designer with the same schematic view. Depending on the level of abstraction not every component of the record/struct is needed. For example a binary passband signal in the RF domain can be used in the feedback path of the loop. Whereas another record or struct can be used to pass a complex wave to the next stage, typically a mixer. The complex wave contains the modulated data and consists of real valued baseband signals. It doesn't matter if the data is represented as in-phase and quadrature phase signals or as amplitude and phase. The current channel center frequency may not be left out, if the data is passed in the baseband regime. A discrete time interface allows the connection to an analog simulator. The digital PLL uses an event-driven simulator and is therefore less demanding in computing time compared to an analog solver.

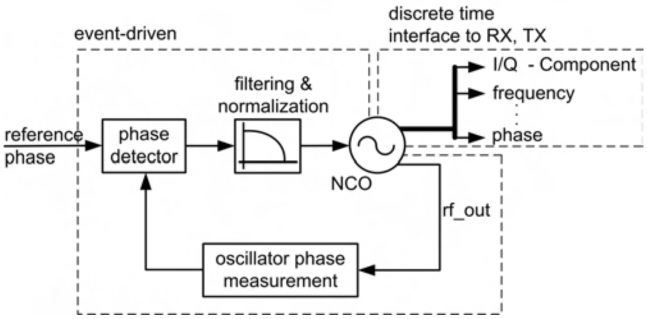

**Figure. 4. Proposed NCO interface.**

**Listing 1. NCO interface description using VHDL.**

```
TYPE complex_wave_t IS RECORD
I , Q : real ;          -- in phase, quadrature phase
frequency : real ;  -- channel center frequency
phase : real ;        -- modulated baseband phase
END RECORD;

TYPE nco_out_t IS RECORD
complex_wave : complex_wave_t;
. . .
rf_out : bit ;          -- rf_output for loop feedback
END RECORD;

ENTITY nco IS
PORT( fcw :IN bit_vector( fcw_length-1 downto 0);
        nco_output :OUT nco_out_t);
END ENTITY nco;
```

Even the modelling of a vector modulator is practicable using the above given description. An extension of the interface description to meet additional requirements is possible.

**Listing 2. NCO interface description using System-Verilog.**

```
typedef struct {
real I , Q;          // in phase, quadrature phase
real frequency ;  // channel center frequency
real phase;         // modulated baseband phase
} complex_wave_t;

typedef struct {
complex_wave_t complex_wave; //discrete time interface
. . .
logic rf_out ;      // rf_output for loop feedback
} nco_out_t ;

module nco (output nco_out_t nco_output,
            input [fcw_length-1:0] fcw);
```

Sampling processes within the oscillator model are required to generate oversampled baseband signals. The sampling rate has to be high compared to the data rate of the system, but is still low compared to the channel frequency. The advantage in speed of the equivalent baseband-model is enormous in comparison to the passband-model and even much bigger in comparison to a mixed baseband-transistor-model [4]. Through this advantage in speed the simulation of an entire transceiver architecture becomes possible.

# 4. SIMULATION RESULTS

## 4.1. Simulation Evaluation

As a result of insufficient analysis and data export options within current mixed-signal simulations environments additional tools for simulation evaluation are needed. Instead of explicitly exporting simulation data from a waveform viewer, VHDL routines have been implemented, which store all required data into a file. Using VHDL the modulated random bit sequence, which is also generated with VHDL, and the sampled oscillator RF-phase are stored into files. These files can then be read by MathWorks Matlab and be analysed.

Starting from the stored random bit sequence Matlab calculates the ideal baseband phase (Figure 5). The simulated baseband phase can be calculated by subtracting the ideal carrier phase from the stored oscillator output phase and is also shown in Figure 5. In order to get a better impression of the phase error, the difference between the ideal and the simulated baseband phase is shown in Figure 6. The simulation length is about 2.1 $\mu$s, because of the GSM specification [1], which says that the absolute phase error within a 511 bit sequence has to be $\Phi_{\mathrm{abs,max}} < 20^\circ$ and the root mean square (RMS) phase error of any 148 bit subsequence within the 511 bit sequence has to be $\Phi_{\mathrm{rms,max}} \leq 5^\circ$. Figure 6 shows that the simulated PLL architecture meets these requirements ($\Phi_{\mathrm{abs}} = 3.97^\circ$, $\Phi_{\mathrm{rms}} = 1.41^\circ$).

The simulated baseband phase can also be used to compute the the eyediagram (see Figure 7). The clearly observable eye-opening shows the good modulation qualities of the utilised PLL architecture.

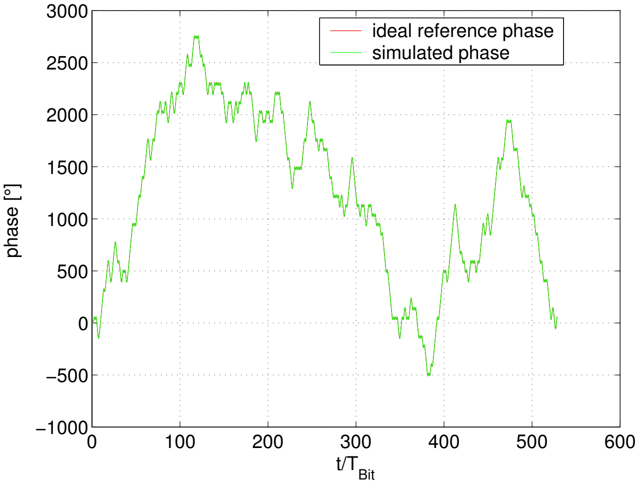

**Figure. 5. Baseband Phase.**

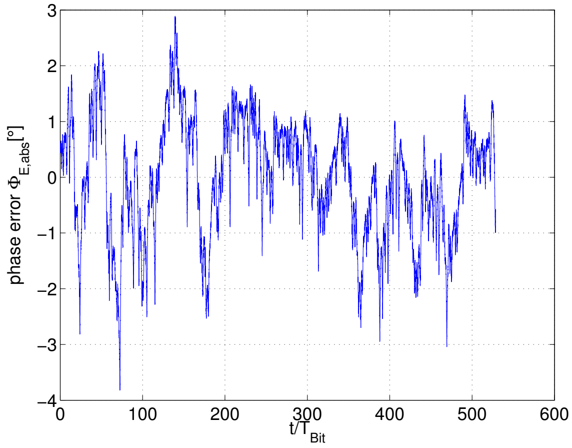

**Figure. 6. Phase error.**

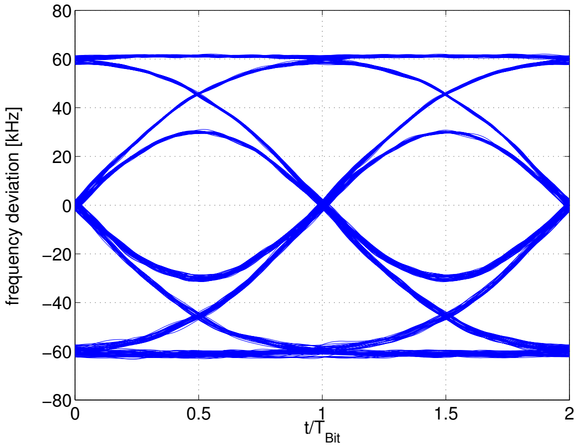

**Figure. 7. Eyediagram.**

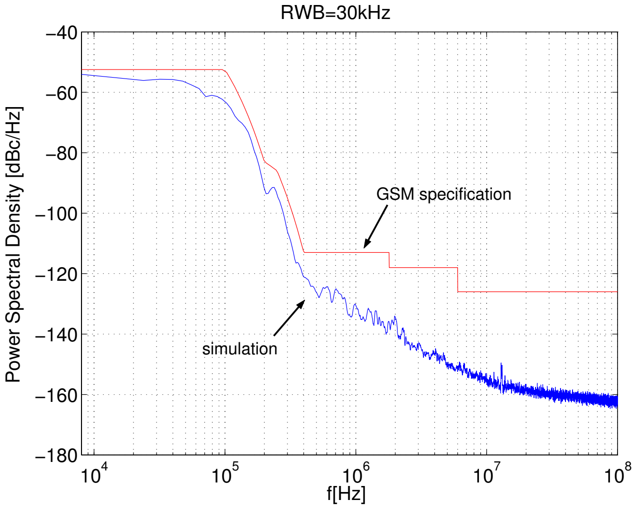

**Figure. 8. Power Spectral Density.**

Furthermore the power spectral density can be calculated from the simulated baseband phase using Matlab and fulfils the GSM specification (see Figure 8). Typical design parameters of a 130 $nm$ CMOS process have been used in the simulation. New more advanced CMOS processes provide even better simulation results because of a higher frequency resolution of the numerically controlled oscillator [11]. The above shown results have been simulated using the following parameters $f_{\mathrm{ch}} = 1.785$ GHz, $f_{\mathrm{ref}} = 13$ MHz.

## 4.2. Parameter Optimization

Since the evaluation procedure is given now, the optimal design parameters of the PLL architecture can be determined. The dependency of the absolute and root mean square phase error on the natural frequency has been analysed and are shown in Figure 9. As expected the phase error becomes smaller when the loop bandwidth increases, because the loop can react faster. The minimal natural frequency should be $f_{\mathrm{n}} = 5$ kHz.

On the other hand the bandwidth may not be to big, because the spectrum worsens as Figure 10 shows. Displayed is the distance of the measured spectrum to the spectral mask at a critical frequency offset of 400 kHz to the carrier. It can be read that the natural frequency shouldn't be greater than $f_{\mathrm{n}} = 20$ kHz.

The optimal bandwidth can thus be determined to 5 kHz $< f_{\mathrm{n}} <$ 20 kHz. The meeting of the GSM specifications is ensured. By simply switching the loop bandwidth the PLL is fully operational after a settling time of 10 $\mu$s. The RMS and absolute phase error are usually better than $\Phi_{\mathrm{rms}} = 1.4°$ and $\Phi_{\mathrm{abs}} = 4.5°$ and errors down to $\Phi_{\mathrm{rms}} = 0.4°$ and $\Phi_{\mathrm{abs}} = 1.3°$ have been simulated. As the distance of the simulated spectrum to the spectral mask is as well usually greater than 8 dB, the PLL offers enough reserve for production variations.

## 5. CONCLUSIONS

In this paper a novel approach for mixed signal circuit design has been presented. A new all-digital PLL based frequency synthesizer architecture has been investigated. The modelling, simulation and analysis has been presented. Simulation shows that the rms phase error of the architecture is

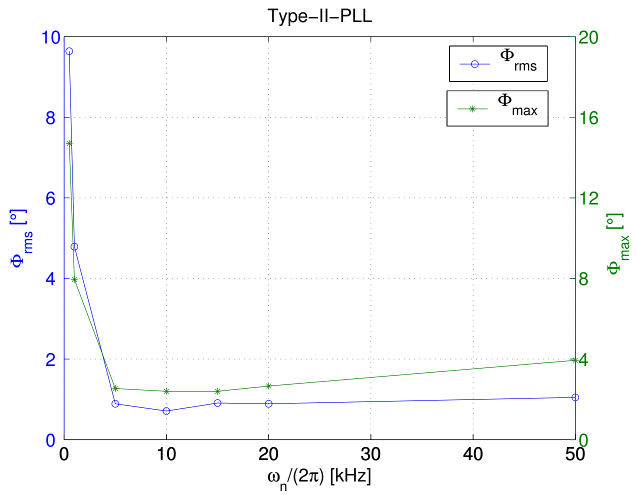

**Figure. 9. Dependency of the rms-phase error on the natural frequency of the type-II PLL.**

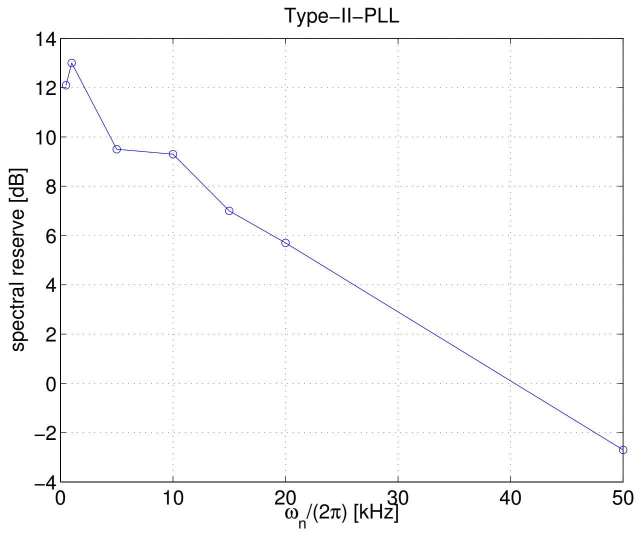

**Figure. 10. Dependency of the distance of the measured spectral density from the spectral mask on the natural frequency of the type-II PLL.**

be below 1.4° and the absolute phase error below 4.5°. The spectral margin can be maintained to be more than 8 $dB$ and the locking time is as low as 10 $\mu$s.

It has been determined that the support of mixed-signal hardware description languages in current design environments has to be improved. Especially the use of VHDL-AMS or System Verilog on schematic level as netlist format and new constructs like records or structs in interface declarations have to be supported. Thereby a novel self contained top-down design flow is enabled, which utilizes the same schematics on different levels of abstraction. By switching cellviews and the use of baseband and passband models the advantage in simulation speed is enormous and the functional and connectivity verification of a whole transceiver becomes possible.

## 6. ADDITIONAL AUTHORS

Additional authors: Ralf Wunderlich(Chair of Integrated Analog Circuits, email: `RW@ias.rwth-aachen.de`) and Stefan Joeres (Chair of Integrated Analog Circuits, email: `Joeres@ias.rwth-aachen.de`) and André Kruth (Chair of Integrated Analog Circuits, email: `Kruth@ias.rwth-aachen.de`).

## 7. REFERENCES

[1] ETSI: 3rd Generation Partnership Project (3GPP),3GPP Specifications Home Page. http://www.3gpp.org/specs/specs.htm.

[2] R. E. Best. *Phase-Locked-Loops*. McGraw-Hill, New York, 5 edition, 2003. ISBN: 0-07-141201-8.

[3] J. E. Chen. Modeling RF Systems. www.designers-guide.org, March 2005.

[4] S. Joeres and S. Heinen. Mixed-Mode and Mixed-Domain Modelling and Verification of Radio Frequency Subsystems for SoC-Applications. In *IEEE International Behavioral Modeling and Simulation Conference*, September 2005.

[5] K. Kundert. Modeling Jitter in PLL-based Frequency Synthesizers. www.designers-guide.org, April 2003.

[6] K. Kundert. Principles of Top-Down Mixed-Signal Design. www.designers-guide.org, February 2003.

[7] K. Kundert, H. Chang, D. Jefferies, G. Lamant, E. Malavasi, and F. Sending. Design of Mixed-Signal Systems on Chip. www.designers-guide.org, December 2000.

[8] C. Muenker. VHDL, A Mixed Signal Simulator for RF Systems Solutions. In *IEEE Radio Frequency Integrated Circuits (RFIC) Symposium, WSH: Mixed Signal Design Methodology & Environment*, June 2004.

[9] Y. R. Shayan and T. Le-Ngoc. All digital phase-locked loop: concepts, design and applications. In *Radar and Signal Processing, IEE PROCEEDINGS F*, volume 136, pages 53–56, February 1989.

[10] R. B. Staszewski, D. Leipold, C.-M. Hung, and P. T. Balsara. TDC-Based Frequency Synthesizer for Wireless Applications. In *IEEE Radio Frequency Integrated Circuits Symposium*, pages 215–218, June 2004.

[11] R. B. Staszewski, D. Leipold, K. Muhammad, and P. T. Balsara. Digitally Controlled Oscillator (DCO)-Based Architecture for RF Frequency Synthesis in a Deep-Submicrometer CMOS Process. *IEEE Transactions on Circuits and Systems*, 50(11):815–828, November 2003.