Obtaining a behavioral model for evaluating the definition of significant moments of the digital signal from perfect provisions in time

A A Pirogov¹, A V Bashkirov¹, I S Bobylkin¹, A V Muratov¹, V V Glotov¹

¹Voronezh State Technical University, Voronezh, Russia
E-mail: vadik-livny@mail.ru

Abstract: The increasing complexity of the systems-on-a-chip, as well as their raising operating frequencies, leads to the more complex and accurate design of the systems for generating clock signals. In this case, it becomes important not only to define all the characteristics that directly affect the accuracy of the generated signals, but also the practical possibility of estimating them using computer simulation. The reason for this is that the numbers of systems, which require modeling, develop faster than the computing power of a data processor. This paper discusses the possibilities of analyzing clock generation systems to estimate the value of random noise. The development of data transmission interfaces seeks to increase the speed of information transmission, whereas this development is proceeding in several directions. Firstly, the increase in speed may be achieved refining modulation systems. Another way to increase the speed of information transfer is to change the data transfer protocol. However, the most effective method of increasing the data transfer rate for any interfaces is still to increase the transmission frequency, i.e. reducing the duration of a single interval, which leads to stricter requirements for both the transmitter and the receiver involved in the physical implementation of the data transfer interface (PHY).

1. Introduction

The source of the internal clock signal of the microcircuit is, in most cases, the phase locked loop (PLL). This unit can perform many functions, but its main function is to multiply the low-frequency reference clock signal (usually obtained from a quartz resonator) by a certain amount, thus obtaining a high-frequency clock signal. It is also possible to simultaneously generate several equally distant phases of the generated signal. A typical system that implements a physical data interface is shown in Fig. 1. It includes two main parts: the Serializer/Deserializer (SERDES) and the logical encoding unit. SERDES acts as a transceiver, which forms the physical representation of the signal, and also receives such a signal from another receiver. Also typically, SERDES generates a clock signal for a logic coding unit that provides information processing between SERDES and a PHY operating circuit.

It is obvious that the reference clock signals generated by the PLL are used both in the transmission lines and in the logical encoding unit. Sometimes PLL clocks are also used in the receiver, but much less frequently. Thus, the stability and accuracy of the signal generated by the PLL determine the stability and accuracy of all other units of the system [1].
2. PLL structure
PLL is a classical feedback system studied by the theory of automated control (TAC). Such systems contain a control object, a sensor and a regulator. The control object is a voltage controlled oscillator (VCO), which generates clock signals. The types of the detector and the regulator differ depending on the type of PLL. Let us review the main blocks common to all PLL systems.

VCO, a voltage controlled oscillator, is a key block of all PLLs. Its task is to generate a periodic signal, with a frequency specified by the so-called ‘control voltage’. The frequency dependence of the control voltage in the operating range is usually linear, and the slope of this dependence is called the gain (marked by the Latin letter ‘K’) [2,3]. The main element that determines the frequency properties of the PLL is the loop filter. The most common solution is to use a first order filter. The very first PLL system was a fully analog circuit (Analog PLL, APLL).

Due to the fact that the reliability of the analog multiplier was not enough to ensure sufficient PLL accuracy, methods were developed to evade this problem. One of these methods is the transition to a digital phase-frequency detector (PD) and a charge pumping circuit (CPD). Such an approach allowed achieving high accuracy and a simpler design process. It is this version of the PLL that is currently the most common, as well as other branches appear on its basis. Systems with a phase-frequency detector and a charge pumping circuit are already partly digital and in practice they always include another element — a frequency divider (DF), located between the VCO and the phase-frequency detector. This element is necessary to obtain the desired frequency multiplication factor, since the detector provides error reduction, i.e. reducing the discrepancy in frequency and phase between the reference clock signal and the feedback signal.

In the case of a digital PLL, a digitally controlled oscillator (DCO) is applied. It generates an output signal with a frequency specified by the control code. The block diagram of the digital PLL is shown in Fig. 2 [5].
3. Factors affecting the quality of the system

Jitter is the deviation from true periodicity of a presumably periodic signal, often in relation to a reference clock signal [6]. This parameter is of great importance both in data transmission systems and in data processing systems. Most microprocessors and digital signal processing units (DSPs) work with clock signals during instruction implementation.

Basic instructions, such as shifting by one bit, can take one clock signal, while addition and subtraction instructions can take three to four clock cycles, and more complex instructions, such as multiplication, can take dozens of clock cycles.

In all these cases, the basic assumption is that the clock signal is accurate; i.e. clock cycles are all the same in duration. However, in reality, the clock cycles are identical only with finite (not infinite) accuracy. This is due to the fact that clock signals are generated by physical devices, which are inevitably associated with some uncertainty or randomness. For example, clock signals may be generated by CMOS inverters [4, 5]. If we assume that \( t_{pd} \) is the propagation delay of each inverter in response to a change at the input, this delay varies from inverter to inverter for each switching moment. This randomness is inherent in any physical device that deals with the movement of electrons at the temperatures above zero degrees Kelvin.

Jitter itself has a broad classification by type, frequency, and reason for occurrence, but jitter is most often distinguished as deterministic and random. Deterministic jitter is set by the characteristics of the PLL structure and can be measured directly using circuit simulation in the time domain. One type of jitter is jitter due to the transmitted data.

An ideal clock generates a data stream that has a fixed length of a single interval, where each user interface is exactly equal to the synchronization period. In our paper we view this data stream as a perfect clock signalsample. An imperfect clock signal leads to inaccurate data flow, where each unit interval is slightly different from the nominal unit interval. Due to the random nature of the data sequence, data transitions do not occur on every front of the user interface. Thus, jitter is not observed from the data if the transition is absent [6].

4. Jitter simulation overview

Random jitter does not manifest itself in the time domain, and can only be measured using specialized analysis, as shown in [7]. However, in real-world circuits such an approach is not justified, in view of the computational complexity of the corresponding analysis. Proceeding from the impossibility of direct measurement, the value of random jitter of the PLL is obtained using behavioral models. There are two main approaches: models in the time domain [8] and in the phase domain [9]. Both approaches involve obtaining all the necessary characteristics of each block included in the PLL and creating for each such block its idealized model, including jitter. Models in the phase domain, despite the name, are modeled as part of temporal analysis, but the values generated by the models are not a voltage or current, but a phase. Such models cannot be integrated into the general test circuit, and are only suitable for determining the value of jitter. Models in the time domain are characterized by behavior close to the original scheme. Such models, in addition to the possibility of estimating jitter directly, make it possible to integrate into more complex test circuits to determine the effect of measured jitter on their operation. An example would be a test of the stability of the serializer to the generated PLL interference. In this paper, this approach is used to ensure further integration into the overall scheme.

5. Simulation of mathematical phase noise model

Jitter is the deviation of the time of occurrence of a certain event from their ideal position. In the case of electronics, the event is the overcoming of the front of a digital signal of a certain threshold value. The concept of an ideal position can be viewed in two aspects: as comparing the position of the front of a given signal with the corresponding front of its ideal variant or as comparing the front of a signal with another front of the same signal. The first option leads to the concept of absolute jitter, and the second to the concept of period jitter. In general, the periodic signal can be represented as the expression (1):
\[ v(t) = A(t)\sin(\omega_0 t + \varphi(t)), \quad (1) \]

where \( A(t) \) is the signal amplitude
\( \omega_0 \) is the signal frequency;
\( \varphi(t) \) is the phase shift.

If we assume that the parameter \( A(t) \) is constant with respect to time, and the parameter \( \varphi(t) \) is 0, then the signal will be an ideal sinusoid. The moments of intersection of this signal of zero value will be located at equal intervals from each other equal to \( kT \), where \( k \) is the number of the interval, and \( T \) is the oscillation period. Such signal can be considered an ideal clock signal. The parameter \( \varphi(t) \), added to the ideal phase \( \omega_0t \) confuses the moments of crossing the zero value of the sinusoid, forming jitter. This parameter in various sources may be called differently, but in this case it is called the phase shift.

Again, if we assume that the clock signal has a nominal period \( T \), then despite the jitter that changes the period duration, the average period value will be \( T \). Comparing the response times of this distorted signal and its variant not subject to jitter, you can get the absolute jitter value, which is described by the expression (2):

\[ a_k = t_k + kT, \quad (2) \]

where \( a_k \) is the absolute jitter value for the \( k \)-th front;
\( t_k \) is the instant of the \( k \)-th edge of the signal under investigation.

The term “absolute” is used because each front under investigation is compared with the corresponding ideal front, independently of the others. It is also possible to evaluate jitter, which is called relative. In this case, the fronts of the studied signal are compared not with the fronts of the ideal, but with the fronts of another signal, also subject to distortion. Mathematically, this can be expressed as (3):

\[ r_k = t_{1k} - t_{2k} = a_{1k} - a_{2k}. \quad (3) \]

It is not difficult to prove that relative jitter can be expressed in absolute terms, as shown in (3).

In the two previous options, the comparison occurred between two different signals. It is also possible to compare the signal with itself, an example of which is period jitter, mathematically expressed (4):

\[ p_k = T_k - T, \quad (4) \]

where \( T_k \) is the duration of the \( k \)-th period; and
\( T \) is the nominal period duration.

Therefore, the period jitter is a change in the position of the signal's front from its previous position with time, which is similar to a change in the duration of the period relative to its nominal value with time.

This type of interference can also be expressed in terms of absolute jitter, as shown in (5):

\[ p_k = a_{k+1} - a_k. \quad (5) \]

Some variation of the period jitter is the jitter in the adjacent periods, defined as the change in the duration of the \( k \)-th period of the signal relative to its previous value and is expressed as (6):

\[ c\bar{c}_k = T_{k+1} - T_k. \quad (6) \]
This type of jitter is fixed in the JEDEC committee standard JESD65B and is most often used to characterize the stability of a spread spectrum clock signal. To understand this, let us consider a signal modulated by a triangular signal. The phase shift in this case will be determined by the quadratic function of time. Since the period jitter is a first order derivative of absolute jitter, its change is described by a linear function of time. The jitter in the neighboring periods is in turn a derivative of the first order from the jitter of the period, which means that it remains constant throughout the duration of the modulated signal. After getting the jitter values for the signal, you can conduct a statistical study of it. The main methods of its statistical representation are the probability density function and the spectral power density. Since jitter is a function of integer argument k, each of which corresponds to some real number jk, then a histogram can be constructed that displays the dependence of the deviation on the number jk, as shown in Fig. 3.

With such statistics, it is not difficult to calculate the probability function density. Let all the measurements of jitter be divided into m parts, each of which represents the number of measurements h(ti) falling within the interval [ti+Δt/2, ti−Δt/2], where Δt is the width of the interval, ai = 1..m.

It is obvious that the sum of all h(ti) is equal to the number of ‘n’ dimensions:

\[ \sum_{i=1}^{m} h(t_i) = n. \]

If we define the function f(ti) as measurements multiplied by the factor \( \frac{1}{n\Delta t} \), then we get:

\[ f(t_i) = \frac{f(t_i)}{n\Delta t}. \]

Summing up all the elements of the function - f(ti) is equal to 1, and its integral over the interval from ‘a’ to ‘b’ is equal to the ratio of the number of measurements on this gap to the total number of dimensions, i.e. probabilities. Graphically, the probability density function can be represented as a graph shown in Fig.4.
In addition to probability density, jitter can also be represented in the frequency domain. To do this, you first need to introduce the concept of phase noise.

In the presence of a phase shift caused by various noise effects, the signal fronts will cross the threshold value at the points defined by equation (7):

$$\omega_0 t + \varphi(t) = 2\pi k.$$  \hspace{1cm} (7)

Based on this equation, you can derive an equation for absolute jitter, which will have the form (8):

$$a_k = -\frac{\varphi(kT_0)}{\omega_0}.$$  \hspace{1cm} (8)

Thus, absolute jitter is a discrete random process obtained by sampling phase shift values at intervals of multiples of $T_0$ divided by the frequency $\omega_0$ [12].

The main tool for the study of jitter and the frequency domain is the power spectral density (PSD), which expresses how the random process is distributed along the frequency axis. If the process has a flat PSD over the entire frequency band, then this process is called white. An example is the thermal noise of a resistor, evenly distributed over the frequency band, up to $6 \cdot 10^{12}$ Hz [13]. If you integrate the PSD on a given frequency band, you can find out the amount of energy that the random process contributes to it. Integration over the entire frequency band will give the total energy of the process, which is equivalent to dispersion.

The power spectral density can be expressed by the following equation (9):

$$S_a(f) = \frac{1}{\omega_0^2} \sum_{n=-\infty}^{+\infty} S_{\varphi}(f + nf_0),$$  \hspace{1cm} (9)

where $S_a(f)$ is power spectral density; $S_{\varphi}(f)$ - phase shift and power spectrum; $f$ - range frequency:

$$-\frac{f_0}{2} < f < \frac{f_0}{2}.$$  

The spectral power is considered in the frequency range limited by $f_0 = 1/T_0$. Any phase shift outside the specified range will be similar to a phase shift within that range. Phase noise is the ratio of the signal power with a frequency $f_0$ taken at a frequency band of 1 Hz to the total power of this signal and can be expressed by equation (10):

Figure 4. Probability density function.
\[ S_\varphi(f) = \frac{S_v(f_0 + f)SSB}{P}, \]  

(10)

where \( S_\varphi \) is the Power spectral density
\( P \) is the total signal strength;
\( f \) is the frequency deviation.

Phase noise is considered for the reason that it is in this form that it is calculated in CAD using noise models of elements. Based on the Khinchin-Kolmogorov theorem, one can find the variance of jitter, by integrating its SPM as shown in (11):

\[ \sigma_a^2 = \int_{-f_0/2}^{+f_0/2} S_a(f) \, df. \]  

(11)

The mean square value of jitter can be obtained by integrating the phase noise in the positive plane, as shown in (12). The factor 2 under the radical means that the phase noise is symmetric about zero:

\[ \sigma_a = \sqrt{\frac{2}{\omega_0^2}} \int_{0}^{+\infty} S_a(f) \, df. \]  

(12)

Thus, based on the equations, it can be concluded that it is possible to transfer from the voltage region to the phase region, followed by obtaining the jitter value [12].

6. Obtaining data to build a model

Behavioral PLL Modeling

On the other hand, to construct a PLL behavioral model, it is necessary to create behavioral models of its main units: a phase detector, a charge pump, a VCO, a frequency divider and a loop filter. Together they can form a working PLL block, as shown in fig. 5.

![PLL simulation model](image)

**Figure 5. PLL simulation model.**

The term “behavioral” in this case means that not all properties are modeled, but only those that have meaning in this context, i.e. those that allow the PLL to function. Also for each block the parameters defining its behavior from the point of view of random jitter will be calculated. Interacting together, they will provide an opportunity to assess the overall jitter of the system. All the models are written in the Verilog-A language, which is used in cases when it is necessary to add a non-standard element to the circuit, process data in real time or to accelerate the simulation by replacing individual blocks with their behavioral models.

Each model in the language of Verilog-A is a module consisting of three parts. The first part contains the definition of the module, i.e. its name and the list of findings that will be visible on the diagram. It
also indicates their direction and type. For historical reasons, the type of all outputs is the same and is called “electrical”. The extended language standard used for mixed modeling adds other types of outputs used for logic signals. In the second part of the module is a list of parameters - the internal properties of the module, user-defined. Module parameters will be available in the element properties window in the schematic editor.

They make it possible to achieve greater universality of the module, so that it can be used to model a class of circuits similar in behavior but different in properties. An example is the model of a charge pump, whose parameter is its output current. Inside the module, the parameters rearrange the immutable values read-only. Finally, the third part of the module defines the element's behavior directly. The work is based on the event level, i.e. The Verilog-A language has a number of tools in it that allow one to obtain information about certain events related both to the state of external signals and to the stages of modeling. For example, the event "cross" allows you to find out if there was an intersection of a certain signal of value 0 in the specified direction. This is one of the most frequently used events. Another important event is the "initial_step", which occurs once during module initialization before starting the simulation. During this event, the initial values of the variables are usually set. Changing the state of signals at the output of a module can only be done outside the body of the handler of an event.

7. Obtaining Voltage Controlled Oscillator (VCO) characteristics
VCO has the following main characteristics:
- frequency of free oscillations;
- frequency dependence of the control voltage;
- phase noise.

To measure the first two parameters, it is necessary to use PSS analysis. To obtain the value of the phase noise it is also necessary to conduct a PNoise analysis.

As a test circuit, we took the circuit shown in fig. 6.

![Figure 6. Scheme for obtaining the characteristics of VCO](image)

The constant voltage source G1 is required to set the control voltage.

The result of such simulation is shown in Fig. 7. To obtain such a schedule, it is necessary to build the first harmonic procedure in the menu of the “Results → Direct Plot → PSS”.
To obtain a phase noise figure, it is necessary to set a certain value of the control voltage (Vctrl). In this case, because the required frequency is 500 MHz, the control voltage is set to the value of 312.828 mV obtained in the previous step. All measurements will be made in the 1 GHz region. To do this, also in the settings of the PSS analysis, you must disable the variable change (the “Sweep” option).

The result can be seen in the menu: “Results → Direct Plot → Main Form → PNoise → Output Noise”. (Fig. 8.).

8. **Distinguishing the charge pump and phase detector on the scheme**

To build a behavioral model of the PLL, it is convenient to consider PD and DPS not as separate blocks, but as an aggregate of them. In contrast to the VCO, all the characteristics of these blocks, except for the values of phase noise, are known. To determine it, it is necessary to assemble the test circuit shown in fig. 9.
Two signals with the same frequency and phase should be fed to the input of the PD using G1 and G2 square wave generators, and a loop filter and a load simulating the VCO input stage should be installed at the output of the charge pumping circuit, vctrl circuit.

It is also necessary to set the initial conditions for the vctrl circuit. The voltage on it should be equal to the control voltage with which the VCO is supposed to work. You can do this by going to the ADEL window in the Simulation - Convergence - Initial Condition menu. According to the results of the calculation, the value of the phase noise is set at 14.28 fs. You should also conduct a temporary simulation to determine the characteristics of a large signal. To do this, the signals from the generators G1 and G2 should be given with some delay. Equal, for example, half the period of the input signal and a frequency equal to the reference frequency of the PLL. After the simulation, an ELF output current graph can be plotted, an example of which is shown in Fig. 10. From the graph you can get the following parameters. The maximum and minimum current corresponding to the \( I_{\text{max}} \) and \( I_{\text{min}} \) lines, as well as the rise and fall times corresponding to the periods \( T_r \) and \( T_f \).

**Figure 10.** Output current.

**9. Distinguishing the frequency divider**

With regards to phase noise, frequency dividers can be divided into two categories: with independent elements and with elements synchronized with the input signal. Examples of both solutions are shown in Fig. 11.
The figure shows that in the case when the output trigger is synchronized with the input signal, the value of the phase noise can be significantly reduced, and equal to the value of the phase noise of the output trigger. In the case of independent elements, the total phase noise is formed from the contributions of the phase noise of each of the triggers.

To measure the phase noise of a trigger, you can use the circuit shown in Fig. 12.

This diagram presents the frequency divider. A variant with a connected capacity that simulates the input stage of the load is possible. In this case, it must be connected to the "Out" circuit.

The G1 generator sets the clock signal with the specified frequency. In this case, it will be necessary to perform a simulation at an input frequency of 1 GHz. The result of the frequency divider is removed from the circuit "Output". It will be necessary to measure the phase noise, as well as the delay between the input and output fronts.

After the simulation, it is necessary to go to the results and obtain the rms value of the phase noise over the entire frequency range. This expression is similar to the following mathematical model:

$$J_{RMS} = \sqrt{\int_{1 \text{ Hz}}^{500 \text{ MHz}} S(f)^2 \, df}$$

where $S(f)$ is the spectral density of phase noise obtained by the drplJitter command $f$ - frequency, Hz.

The limits of integration are defined by the last two parameters of the expression range from 1 Hz to the output frequency of the divider. It is also necessary to carry out a transient analysis (tran) to determine the delay between the input and output of the divider. In order to do this, we used a source of G1 to form a signal of rectangular pulses. To estimate the delay between signals, you can use the delay function. The essence of the function is shown in Fig. 13.
10. Obtaining the PLL main characteristics

After the models of all the internal PLL units have been created, you can proceed to getting the total value of the phase noise. For testing, it is necessary to assemble the circuit shown in Figure 14. The reference frequency is set by the G1 generator and in this case is equal to 100 MHz. The output frequency is removed from the \( f_{exit} \) circuit. Since the models created do not support any analyzes other than time analysis and DC analysis, phase noise will be obtained using time analysis.

The simulation settings are following the TRAN analysis: The “Simulation Time” parameter should be set to a value equal to the required time for the PLL to enter the mode, as well as the time required to collect the results. In this case, the first stage takes 1 \( \mu \)s, the second also 1 \( \mu \)s. Thus, the total simulation time is 2\( \mu \)s.

In this case, the accuracy of the simulation can be set to the value “Conservative”, since Verilog-A models have great modeling speed.

As a result of the simulation, the time dependence of the output frequency of the PLL was obtained, as shown in Fig. 14. It can be seen from the figure that as a result, the frequency of the PLL has been set to 500 MHz.

Figure 14. Dependence of the output frequency PLL on time

Also from the figure we can conclude that the capture frequency occurred after 1 \( \mu \)s. All measurements of the signal should be carried out during this period. If we take into account that the output signal chain is called \( \text{pllclk} \), then the expressions for obtaining the necessary section will look like this: \( \text{clip}(\text{VT}(/ \text{pllclk}) 1\mu 2\mu) \). To build an eye pattern diagram, go to the following menu: “Measurements -> Eye Pattern Diagram”.
In the “Start / Stop” penta it is necessary to indicate the range for the progressive. In this case, it is 1u - 2u. In the column "Period" indicates the period of the signal. In this case, it is 4ns. As a result, an eye pattern will be constructed, an enlarged fragment of which is shown in Fig. 15.

![Figure 15. Fragment of the eye pattern diagram.](image)

Based on the above diagram we can make conclusions about the numerical value of the absolute jitter, considering the width of the overlapping area of the fronts. In this case, it is $J_a = 13$ps.

The period jitter in this case is equal to $J_c = 518$ fs.

11. Conclusions
The increasing complexity of interfaces for data transmission sets more challenges for engineers who develop each unit of the device. The problem is no less acute in the development of the reference clock generation unit. To assess the reliability budget of the entire system, new techniques are required to determine the value of all necessary characteristics of the designed unit, which also includes random jitter. The main requirement for such methods is accuracy and speed, which allow us to achieve behavioral models.

References
[1] Slivkin V G 2004 Electromagnetic compatibility of electrical equipment of information technologies under the influence of pulsed electromagnetic fields (Samara, SGTU)
[2] Tryapitsyn A V Development of the methodology for the design of wireless local area networks with EMC (Moscow: Mir)
[3] Williams T E 2004 EMC for Product Designers (Moscow: Publishing house "Technologies")
[4] Kechiev L N 2007 Proektirovaniepechatnykh plat dlyatsirovobyistrodeystvuyushcheyapparatury (Moscow; Gruppa IDT)
[5] Surikov V V 2010 Elektromagnitnayasovmestimost' priemno-peredayuschikhustroystv, raspolozhennykhnaelementakhkonstruktsiysozhnoyformy (St. Petersburg: Eva)
[6] Knyazev A D 1984 Elementyteoriiipraktikiobespecheniyaelektromagnitnoysovmostiradioelektronnykhreds tv (Moscow: Radio isvyaz)
[7] Hale U K 1980 Frequency assignment. Theory and applications. 68 112
[8] John R B 1990 Electronic System Design: Interference and Noise Control Techniques (Moscow: Mir)

[9] Habiger E H 1995 Electromagnetic Compatibility. Fundamentals of its provision in technology. (Moscow: Energoatomizdat)