Introduction to FPGA-Based ADPLLs

1. Introduction

1.1. FPGA Approach to Electronics

System architects and board designers must constantly trade off costs, development time, performance, and supportability. Field Programmable Gate Arrays (FPGAs) are programmable ICs that give the designer many of the advantages of a highly integrated digital IC without the NRE costs and fabrication time of a custom digital ASIC. The general functional flexibility of FPGAs invites their consideration in communications, control, and test applications where different standards or data rates may need to be supported by the same hardware.

Likewise, the frequency flexibility of programmable oscillators such as the Skyworks Solutions Si514, Si570, and Si598 frequently lands them on the same boards where they are used as independent asynchronous oscillators. However, these programmable oscillators can also be placed under FPGA control to implement a particular form of highly configurable Phase Locked Loop. This application note is intended to serve as a brief introduction to this approach and its advantages.

1.2. Taxonomy of PLLs

Phase Locked Loops or PLLs are electronic feedback circuits which lock an output signal's phase relative to an input reference signal's phase. The signals of interest may be any periodic waveform, but are typically sinusoids or digital clocks. These circuits are widely used in communications, computers, control, and measurement applications for frequency synthesis, clock and data recovery, clock distribution, and other more specialized functions.

PLLs are typically divided into the broad categories listed in Table 1, and are described following Roland Best's terminology (1997). The hardware PLLs may be implemented in discrete or integrated technology. Software PLLs will not be described in this application note other than to state that SPLLs can mimic any of the hardware PLLs, provided there is sufficient sampling, and the instructions can be executed fast enough for the application.

Table 1. General PLL Categories

PLL Category	Phase Detector	Phase Error Signal	Loop Filter
LPLL (Linear PLL)	Analog	Analog	Analog
DPLL (Digital PLL)	Digital	Analog	Analog
ADPLL (All Digital PLL)	Digital	Digital	Digital
SPLL (Software PLL)	Software	Software	Software

Linear (or analog) PLLs use analog 4-quadrant multipliers, such as mixers, as phase detectors. LPLLs are often used for frequency translation and are therefore found in frequency synthesizers, radios, and phase noise instrumentation.

The most common PLL in use today is the classic Digital PLL, so-called due to its use of a digital phase detector. However, these circuits typically follow the digital phase detector with a charge pump whose output is converted to an analog voltage phase error signal. This analog phase error signal is then filtered and applied to a Voltage Controlled Oscillator (VCO). A better description might be to refer to this version as a mixed signal or charge pump PLL. Classic DPLLs and LPLLs have generally been preferred for high-performance (low phase noise), high frequency applications.

By contrast, PLLs operating only on digital signals are commonly referred to as ADPLLs or All Digital PLLs. The filtered digital error signal is applied to a Digitally Controlled Oscillator (DCO). The general benefit for considering this approach are discussed below.

A popular ADPLL option today is to instantiate as much digital circuitry as possible in an FPGA. This application note will focus on FPGA-based ADPLLs where the DCO is off chip, i.e., not implemented in the FPGA. In particular, we will concentrate on DCOs, which can be directly controlled by a serial interface. Skyworks Solutions offers several I^2C oscillators (e.g., Si514, Si570, Si598) which can be used as DCOs for such FPGA-based ADPLL applications.

2. Benefits

2.1. Advantages to ADPLLs

There are a number of reasons to consider an ADPLL as opposed to a more conventional PLL:

- Analog components and circuits tend to have more variation across PVT (Process, Voltage, and Temperature) and to drift or age. They are more sensitive to parasitics compared to digital circuits which also have higher noise immunity. The Vc or analog control voltage is typically a very sensitive node for example.
- An ADPLL may be able to implement loop filter topologies which are impractical to realize using analog components.
- An integrated CMOS digital design is usually lower cost, smaller area, and more easily ported than its analog counterpart. In short, integrated ADPLLs can take better advantage of Moore's Law. A digital design flow is also an easier development than a more analog intensive one.
- ADPLLs can be designed to be flexible with respect to loop parameters. Implementations can take advantage of this flexibility for calibration, digital freeze and hold operations, self test, etc.
- ADPLLs can control a DCO as if it has virtually infinite Absolute Pull Range (APR). This capability eliminates the need for high Kv VCOs, which may have increased phase noise within their modulation bandwidths.
- ICs rely on increasingly lower voltages and smaller geometries. It can be argued that modern integrated PLLs are fundamentally better suited for the ADPLL approach. See, for example, Staszewski and Balsara (2006, page xiii):

  "In a deep-submicron CMOS process, time-domain resolution of a digital signal edge transition is superior to voltage resolution of analog signals."

3. Architecture

3.1. Basic PLL

The principal elements in a basic PLL are illustrated in the PLL block diagram in Figure 1. The dividers may not necessarily be needed but are included for generality. A reference frequency, fREF, is divided down by the input reference divider R and applied to the phase detector, PD, here shown as a mixer. Likewise, the output frequency, fOUT, is divided by feedback divider N and fed back to the phase detector. The phase detector compares the phase of fREF/R with fOUT/N. The output of the phase detector is an error voltage that is filtered by a low pass filter (LPF) and presented to the VCO as a control voltage Vc. When the loop is locked, fREF/R = fO/N, and the error voltage is minimized.

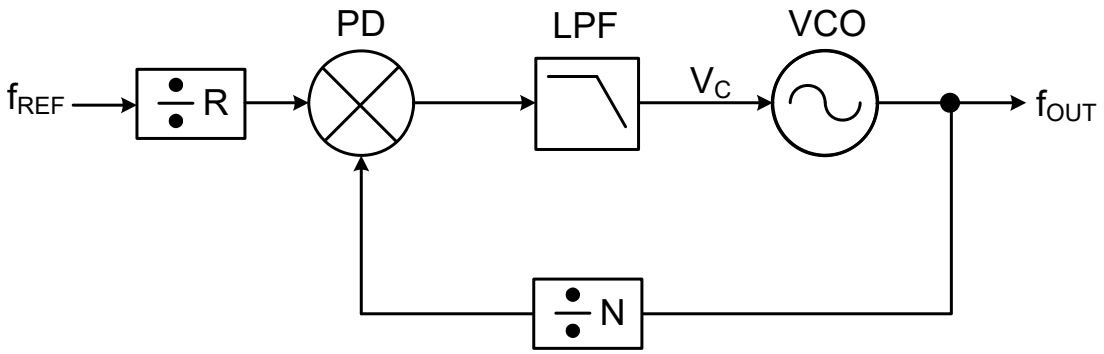

Figure 1. Basic PLL Block Diagram

3.2. Basic ADPLL

3.2.1. Basic ADPLL Block Diagram

The all digital PLL counterpart to the basic PLL substitutes digital signals and blocks as shown in Figure 2. The output of the digital phase detector is a digital error signal that is filtered and presented to the DCO as a control word “M” or its logical equivalent.

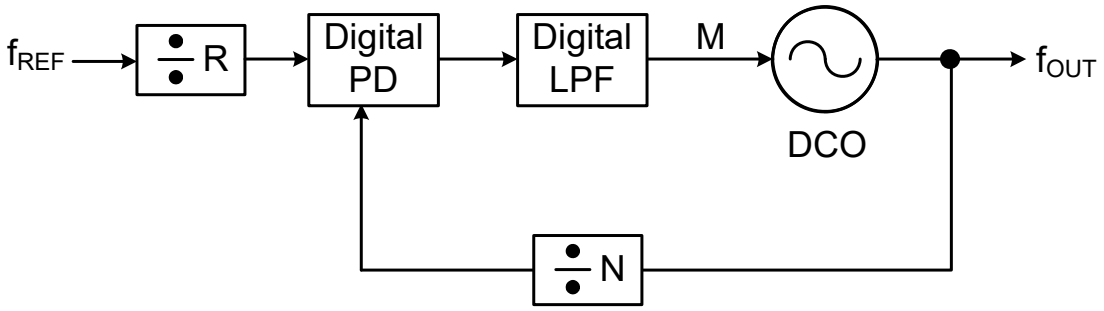

Figure 2. Basic ADPLL Block Diagram

3.3. FPGA-based ADPLL

3.3.1. Block Diagram

It is possible to instantiate an ADPLL entirely within an FPGA and, in fact, some FPGAs include built-in system timing PLLs as resources for the designer. However, the purpose of this application note is to describe custom FPGA-based ADPLLs, which implement all the blocks in a single FPGA except for the DCO. The previous ADPLL block diagram is then partitioned as illustrated in Figure 3. If the DCO is controlled serially, then the control word "M" described previously is typically transferred as a series of register updates.

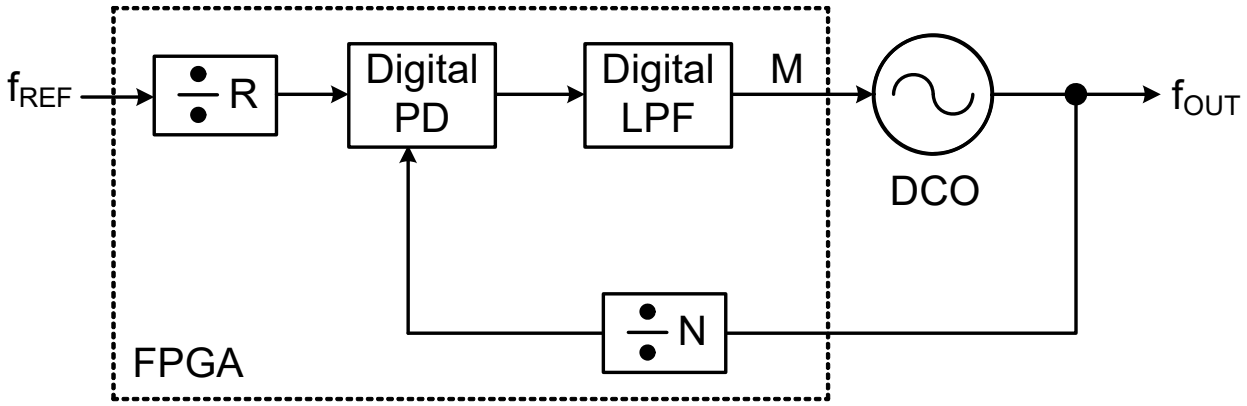

Figure 3. Basic FPGA-based ADPLL Block Diagram

3.3.2. Advantages

For the ADPLL to be competitive in a given application, it needs to have sufficient accuracy, resolution, and speed to support the desired loop behavior while yielding a relatively low phase noise (low jitter) output clock with few or insignificant spurs. For the board or system designer, the FPGA-based ADPLL with external DCO is one approach to achieving these goals.

Partitioning the ADPLL in this way has a number of potential benefits:

- Digital designers can exploit their expertise in DSP to implement a custom PLL, taking full advantage of the economies of scale, device flexibility, and tools provided by the FPGA manufacturers.
- Since many board designs have one or more FPGAs already, there may be less incremental cost for adding an FPGA-based ADPLL as opposed to other implementations.
- The DCO, despite being digitally controlled, is a mixed signal device requiring analog and mixed signal design expertise. In this sense, the "hardest" part of the ADPLL is taken off the chip and the design can be optimized by vendors specializing in DCO technology.
- Custom FPGA-based ADPLLs can exploit an external DCOs specific performance features such as frequency range, steps, resolution, and jitter.
- Finally, it may be advantageous from a power supply, switching noise, and crosstalk point of view to "dis-integrate" the DCO from the FPGA.

It is also possible to make the functional equivalent of a DCO out of a suitably matched DAC + VCO combination. (See for example Xilinx App Note XAPP854). This combination would not result in the creation of an ADPLL as described here; however, it would perform in much the same way. In addition, there might be some relative advantage regarding loop bandwidth since serial communication would not be a limiting factor. However, this combination would have to be balanced against the cost, complexity, and likely higher noise contribution of another component in the signal path.

3.3.3. An Example FPGA-based ADPLL

A very basic example of an FPGA-based ADPLL is illustrated in the block diagram in Figure 4, where there are six major Verilog blocks:

1. Reference Input Divider
2. Digital Phase Detector
3. Digital Low Pass Filter (Loop Filter)
4. Command Conversion
5. I^2C Master
6. Feedback Divider

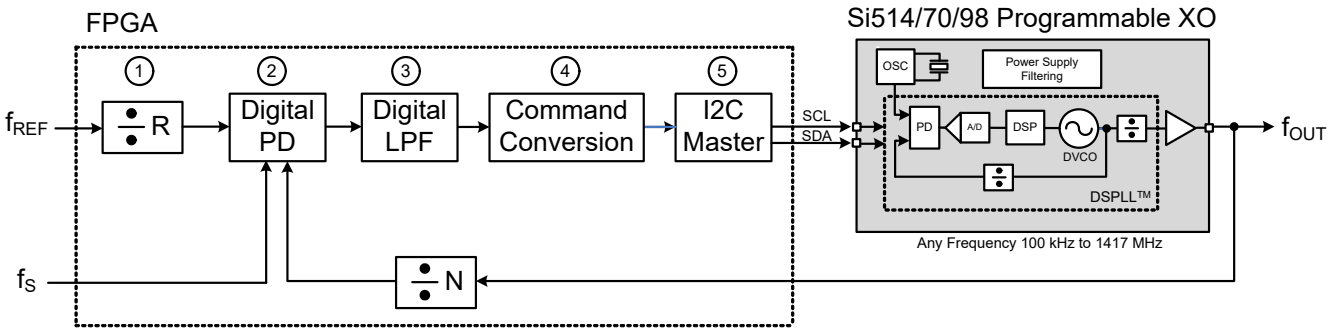

Figure 4. Example FPGA-based ADPLL Block Diagram

This example design uses the following major hardware components:

- Altera FPGA P/N EP3C25F324 (mounted on Altera Cyclone 3 Starter Kit evaluation board)
- Skyworks Solutions Si514, Si570, and Si598 I^2C-programmable oscillators

In this approach, a sampling clock (f_s) is supplied to the Flip Flop–Counter phase detector. This version of ADPLL is a relatively basic example. However, it is sufficient to demonstrate some very important points regarding the application of ADPLLs. Additional details and performance measurements pertaining to this design will be included in future versions of this application note.

4. Additional Reference Resources

- Best, Roland E. Phase Locked Loops: Design, Simulation, and Applications. 3rd ed. New York: McGraw-Hill, 1997. See pages 5-6 for Best's classification of PLLs.
- Staszewski, Robert and Poras T. Balsara. All-Digital Frequency Synthesizer in Deep-Submicron CMOS. Hoboken, New Jersey: Wiley-Interscience, 2006. The authors refer to the paradigm cited throughout the text. See discussion on pages xiii and 26.
- Xilinx App Note XAPP854, "Digital Phase-Locked Loop (DPLL) Reference Design", Justin Gaither, October 10, 2006.

NOTES:

SKYWORKS®

ClockBuilder Pro

Customize Skyworks clock generators, jitter attenuators and network synchronizers with a single tool. With CBPro you can control evaluation boards, access documentation, request a custom part number, export for in-system programming and more!

www.skyworksinc.com/CBPro

Portfolio
www.skyworksinc.com/ia/timing

SW/HW
www.skyworksinc.com/CBPro

Quality
www.skyworksinc.com/quality

Support & Resources
www.skyworksinc.com/support

Copyright © 2021 Skyworks Solutions, Inc. All Rights Reserved.

Information in this document is provided in connection with Skyworks Solutions, Inc. ("Skyworks") products or services. These materials, including the information contained herein, are provided by Skyworks as a service to its customers and may be used for informational purposes only by the customer. Skyworks assumes no responsibility for errors or omissions in these materials or the information contained herein. Skyworks may change its documentation, products, services, specifications or product descriptions at any time, without notice. Skyworks makes no commitment to update the materials or information and shall have no responsibility whatsoever for conflicts, incompatibilities, or other difficulties arising from any future changes.

No license, whether express, implied, by estoppel or otherwise, is granted to any intellectual property rights by this document. Skyworks assumes no liability for any materials, products or information provided hereunder, including the sale, distribution, reproduction or use of Skyworks products, information or materials, except as may be provided in Skyworks' Terms and Conditions of Sale.

THE MATERIALS, PRODUCTS AND INFORMATION ARE PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND, WHETHER EXPRESS, IMPLIED, STATUTORY, OR OTHERWISE, INCLUDING FITNESS FOR A PARTICULAR PURPOSE OR USE, MERCHANTABILITY, PERFORMANCE, QUALITY OR NON-INFRINGEMENT OF ANY INTELLECTUAL PROPERTY RIGHT; ALL SUCH WARRANTIES ARE HEREBY EXPRESSLY DISCLAIMED. SKYWORKS DOES NOT WARRANT THE ACCURACY OR COMPLETENESS OF THE INFORMATION, TEXT, GRAPHICS OR OTHER ITEMS CONTAINED WITHIN THESE MATERIALS. SKYWORKS SHALL NOT BE LIABLE FOR ANY DAMAGES, INCLUDING BUT NOT LIMITED TO ANY SPECIAL, INDIRECT, INCIDENTAL, STATUTORY, OR CONSEQUENTIAL DAMAGES, INCLUDING WITHOUT LIMITATION, LOST REVENUES OR LOST PROFITS THAT MAY RESULT FROM THE USE OF THE MATERIALS OR INFORMATION, WHETHER OR NOT THE RECIPIENT OF MATERIALS HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Skyworks products are not intended for use in medical, lifesaving or life-sustaining applications, or other equipment in which the failure of the Skyworks products could lead to personal injury, death, physical or environmental damage. Skyworks customers using or selling Skyworks products for use in such applications do so at their own risk and agree to fully indemnify Skyworks for any damages resulting from such improper use or sale.

Customers are responsible for their products and applications using Skyworks products, which may deviate from published specifications as a result of design defects, errors, or operation of products outside of published parameters or design specifications. Customers should include design and operating safeguards to minimize these and other risks. Skyworks assumes no liability for applications assistance, customer product design, or damage to any equipment resulting from the use of Skyworks products outside of Skyworks' published specifications or parameters.

Skyworks, the Skyworks symbol, Sky5®, SkyOne®, SkyBlue™, Skyworks Green™, Clockbuilder®, DSPLL®, ISOmodem®, ProSLIC®, and SiPHY® are trademarks or registered trademarks of Skyworks Solutions, Inc. or its subsidiaries in the United States and other countries. Third-party brands and names are for identification purposes only and are the property of their respective owners. Additional information, including relevant terms and conditions, posted at www.skyworksinc.com, are incorporated by reference.

Skyworks Solutions, Inc. | Nasdaq: SWKS | sales@skyworksinc.com | www.skyworksinc.com
USA: 781-376-3000 | Asia: 886-2-2735 0399 | Europe: 33 (0)1 43548540 |