ANALOG DEVICES

AN-0983
APPLICATION NOTE

One Technology Way • P.O. Box 9106 • Norwood, MA 02062-9106, U.S.A. • Tel: 781.329.4700 • Fax: 781.461.3113 • www.analog.com

Introduction to Zero-Delay Clock Timing Techniques

by Ken Gentile

Zero-delay refers to the ability of a clock synthesizer to provide an output signal that is edge aligned with a clock reference source. Applications include many synchronous systems, such as the SONET and SDH networks, high speed network servers, network line cards, as well as baseband timing for W-CDMA and Wi-Fi.

ZERO-DELAY ARCHITECTURE

At a minimum, an integrated zero-delay clock synthesizer requires three building blocks (see Figure 1). The first building block is a phase-locked loop (PLL), of either the common analog variety or one of the more recent all-digital designs. The second building block is two (or more) output drivers with matched propagation delay. The third building block is a variable delay element in the feedback path of the PLL. In addition, the zero-delay architecture requires equal interconnect delay from the synthesizer outputs to their associated target devices. Equal interconnect delay is a fundamental component of the zero-delay architecture. Without it, clock edge alignment at the target devices is not possible.

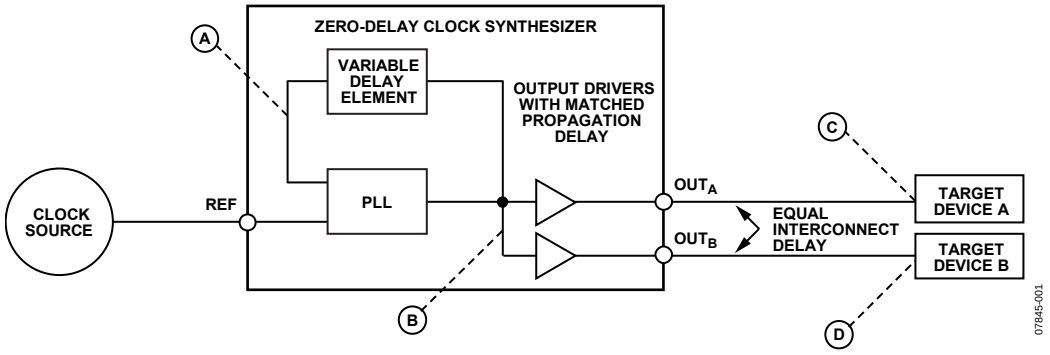

Figure 1. Generic Zero-Delay Synthesizer

TABLE OF CONTENTS

ZERO-DELAY OPERATION

To understand zero-delay operation, refer to Figure 2, which is the timing diagram for the zero-delay architecture shown in Figure 1. Note that the action of the PLL causes the clock edges at Point A to align with the REF clock edges. The fact that the variable delay element resides in the PLL feedback path means that the clock edges at Point B precede those at Point A by the amount of the variable delay.

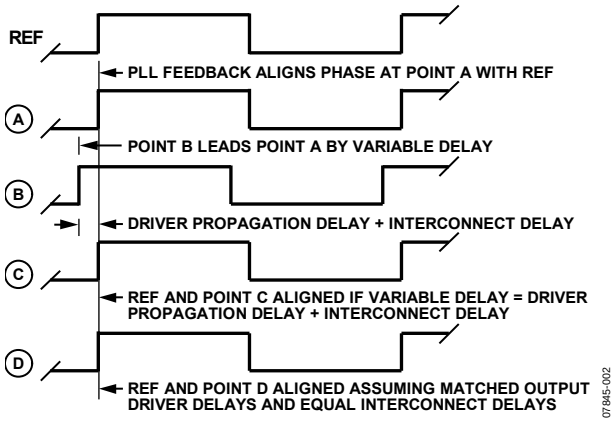

Figure 2. Zero-Delay Timing Diagram

By setting the variable delay equal to the sum of the output driver propagation delay and the interconnect delay, the clock edges at Point C coincide with those at Point A, which coincide with the REF clock edges. Therefore, the clock edges at REF and Target Device A are edge aligned. Furthermore, because the output drivers share the same input signal, and this signal experiences the same delay through each driver and the same delay through each interconnect, then the clock edges at REF are aligned with the clock edges at Target Device B as well as with the clock edges at Target Device A. In fact, this concept can be extended to as many outputs as desired, as long as all the drivers exhibit equal propagation delay and all the interconnects have equal delay.

The timing diagram in Figure 2 indicates that the PLL provides a frequency translation ratio of unity (that is, $f_{REF} = f_{OUT}$). This simplifies the presentation but is not a requirement. In fact, any multiplication or division factor that provides an integer (or reciprocal integer) output/input frequency ratio is acceptable.

In general, the frequency of the input and output signals must be related harmonically in order for clock edge alignment to be meaningful.

It is important to realize that the term zero-delay applies to relative time rather than absolute time. In Figure 2, it is clear that the variable delay element causes the clock edges at Point B to precede the REF edges. Obviously, the clock edges at Point B cannot precede the REF edges in absolute time because this requires travelling backward in time. In fact, the absolute clock edges at Point B occur at least one entire clock period (less the variable delay) after the REF edges. In any practical timing application, only relative edge alignment to a reference clock is necessary, so the term zero-delay is completely appropriate in this context.

Although Figure 1 is useful for demonstrating the concept of zero-delay architecture, as an integrated solution it is limited. A more general solution includes a programmable frequency divider in both the REF and feedback paths of the PLL (see Figure 3). The inclusion of the REF and feedback dividers allows a single device to satisfy a multitude of applications by making it possible to program the input/output frequency ratio. However, only harmonically related input/output frequency ratios are relevant in the context of zero-delay applications. Note that a second variable delay unit appears in the REF path. Its main function is to provide edge timing compensation for any delay introduced by the REF divider.

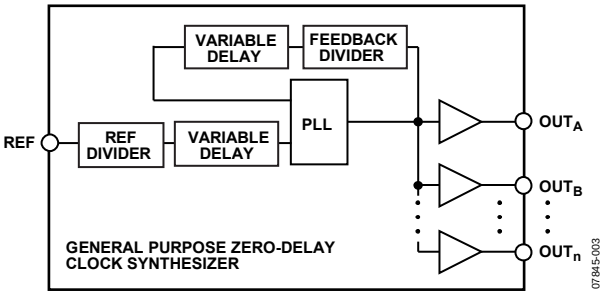

Figure 3. General-Purpose Zero-Delay Synthesizer

A further extension of the general-purpose zero-delay architecture is to provide external access to the PLL feedback path. This allows the zero-delay architecture to accommodate an external fanout buffer, as shown in Figure 4.

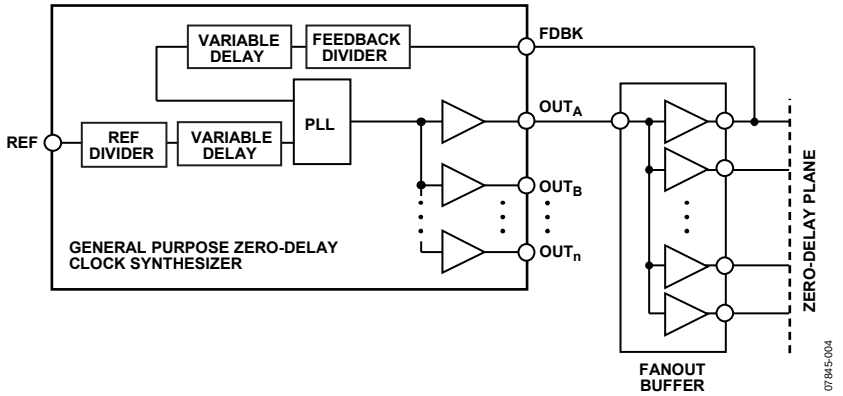

Figure 4. External Zero-Delay Architecture

AN INTEGRATED SOLUTION

The AD9520 and AD9522 are general-purpose clock synthesizers that offer an integrated zero-delay solution for a host of applications. The only difference between the two devices is that the AD9520 has LVPECL output drivers while the AD9522 has LVDS drivers. The output drivers of either device are programmable as CMOS, as well. A block diagram of the AD9520/AD9522 appears in Figure 5.

The device has a serial programming interface enabling a wide range of clock synthesis solutions and a flexible architecture enabling both internal and external zero-delay applications. The PLL portion of the AD9520/AD9522 consists of the PFD/charge pump, external loop filter, and integrated VCO (with optional K-divider). The R-divider is the reference divider and the N-divider is the feedback divider. Following each divider is a programmable delay unit. Both of the programmable delay units provide approximately 1100 ps of delay adjustment in steps of approximately 120 ps. The AD9520/AD9522 has twelve output drivers (Channel 0 through Channel 11) partitioned into four groups of three. In addition, each group of three channels has access to an optional channel divider (M_0 through M_3).

The channel dividers provide an additional layer of frequency division that enhances the flexibility of the device for frequency synthesis applications. In addition to frequency division, the channel dividers offer coarse edge timing adjustment. The output edge timing of each channel divider is programmable in steps of ΔT (up to 32 steps), where $\Delta T = 1/f_{CLK}$ and f_{CLK} is the frequency at the input to the channel divider.

Although the channel dividers enhance the frequency synthesis capabilities of the AD9520/AD9522, they complicate applications involving zero-delay. The reason is that the zero-delay architecture relies on equal delay from the output of the PLL to the input of each of the output drivers. The presence of the channel dividers breaks the required delay equivalence unless the user applies the appropriate action. Specifically, any channels used to perform the zero-delay function must be treated the same. That is, if one group of zero-delay outputs employs a channel divider, then the other zero-delay outputs must also employ a channel divider. Furthermore, the same divide and delay values must be applied to all the channel dividers associated with zero-delay outputs. These steps ensure that the delay from the output of the PLL to the output of each channel driver is the same for all channels (at least to the extent of the internal delay matching between channels).

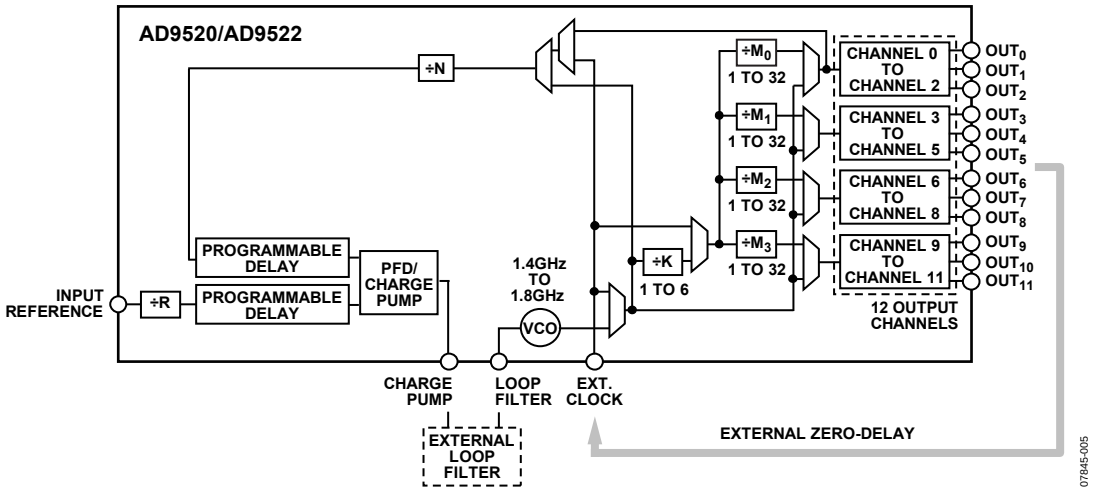

Figure 5. AD9520/AD9522 Functional Block Diagram

TIMING ERRORS: SKEW AND OFFSET

In theory, with perfect matching of the external interconnect delay, the clock signals appearing at the zero-delay plane are edge aligned with the REF signal. In practice, however, two types of timing error are typically present. One is timing skew, which is a slight variation in edge alignment distributed randomly among multiple clock signals. The other is timing offset, which is a constant offset in edge alignment (relative to the REF signal) that is the same for all the output clock signals.

Regarding timing skew, one error source is within the device itself. This internal timing skew is a result of slight differences in propagation delay among the device's output channels. Although the designers of the AD9520/AD9522 strove to minimize timing skew, some variability is inevitable. Another source of timing skew is slight errors in the physical or electrical lengths of the external interconnections.

Of the two timing skew sources, the internal source is typically negligible in comparison to the external timing skew. This is especially true when using the AD9520 LVPECL output, which exhibits a skew of only ~50 ps across all channels. Even when using the AD9520/AD9522 CMOS or AD9522 LVDS outputs, the skew across all channels is only ~300 ps or ~200 ps, respectively. If the user is diligent about nulling out the external skew by design, however, then internal skew can become the dominant skew source. Keep in mind that, in practice, some amount of timing skew is inevitable, so one must determine the maximum tolerable timing skew for the specific application and tailor the design accordingly.

The other timing error, timing offset, manifests itself differently in the internal and external zero-delay architectures of the AD9520/AD9522. With the internal architecture, there are two considerations (see Figure 6). First, the output driver resides outside of the PLL feedback path, so its propagation delay appears as a timing offset relative to the REF signal. The offset amounts to ~100 ps for the AD9520 LVPECL drivers, ~1400 ps for the AD9522 LVDS drivers, and ~1900 ps for the AD9520/AD9522 CMOS drivers. The second consideration involves an inherent delay mismatch between the REF and feedback divider paths. The reason is that the architecture of the N- and R-dividers is fundamentally different, resulting in a delay mismatch. In addition, the REF path includes a clock receiver stage. The result is that the REF path exhibits ~900 ps more delay than the feedback path (assuming the variable delay units are bypassed or have equal delay values). This means that the total unadjusted offset is 1000 ps, 2300 ps, or 2800 ps with the LVPECL, LVDS, or CMOS output, respectively.

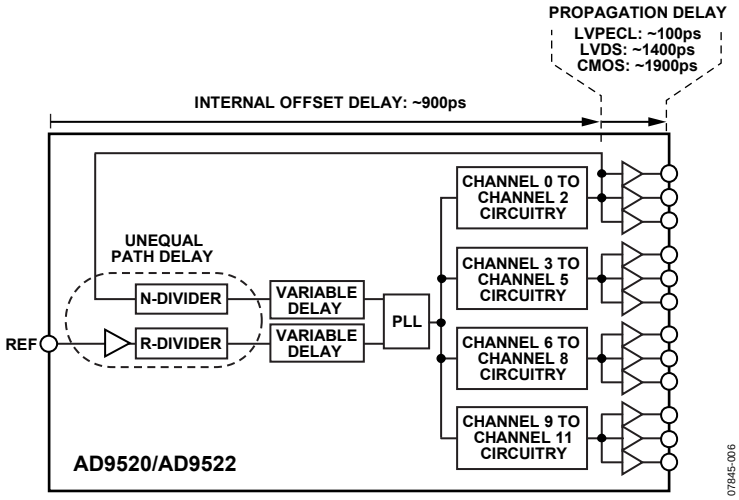

Figure 6. AD9520/AD9522 Internal Zero-Delay Timing Offset Errors

It is possible to reduce this inherent offset by bypassing the variable delay unit in the REF path and enabling the one in the feedback path. With the 1100 ps adjustment range offered by the variable delay unit in the feedback path, the entire 1000 ps of intrinsic offset (when using the LVPECL output) can be eliminated. However, when using the LVDS or CMOS outputs the offset is only reducible to about 1200 ps and 1700 ps, respectively (the amount that the intrinsic offset exceeds the maximum adjustment range). In any case, one must ensure that any residual timing offset falls within the maximum tolerable limits for the specific application.

Figure 7 and Figure 8 are actual time domain measurements using the internal zero-delay architecture. These figures demonstrate the virtual elimination of timing offset error using the AD9520 LVPECL output. The measurements were taken by using matched probes connected directly to the REF input and OUT_1 pins of the device.

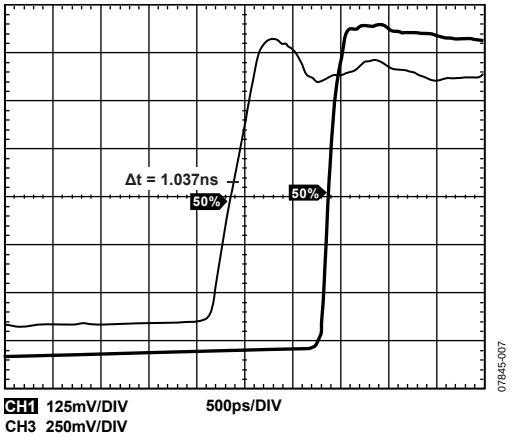

Figure 7. Internal Zero-Delay Intrinsic Offset Timing Error (AD9520)

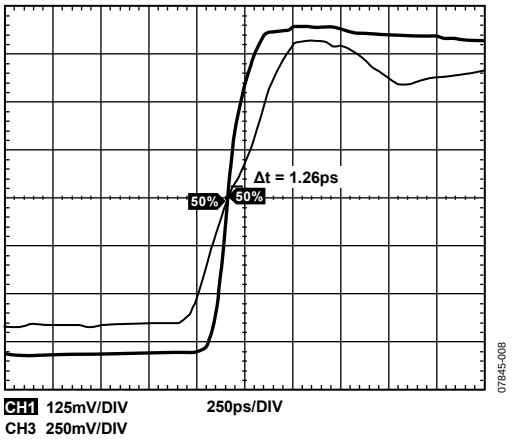

Figure 8. Internal Zero-Delay Adjusted Offset Timing (AD9520)

When using the external zero-delay architecture, the propagation delay of the output drivers has no impact on offset timing error because the driver (and any external buffer) is included in the feedback path of the PLL (see Figure 9). The PLL effectively absorbs the propagation delay of the OUT_0 buffer (arbitrarily chosen) along with the delay through the external circuit. However, as with the internal zero-delay architecture, there remains an inherent delay mismatch between the REF and feedback divider paths. Furthermore, with the external architecture, the external clock input requires a clock receiver (shown at the input to the N-divider). The overall offset delay is 600 ps, which is 300 ps less than that of the internal architecture due to the additional delay associated with the external clock input receiver.

Since the inherent offset timing error is only 600 ps, it is possible to null it out by making the variable delay in the feedback path 600 ps greater than the variable delay in the REF path. Of course, the 120 ps granularity of the variable delay units means that a residual offset of as much as 60 ps may still exist. Thus, one must ensure that up to 60 ps of residual timing offset falls within the maximum tolerable limits for the specific application.

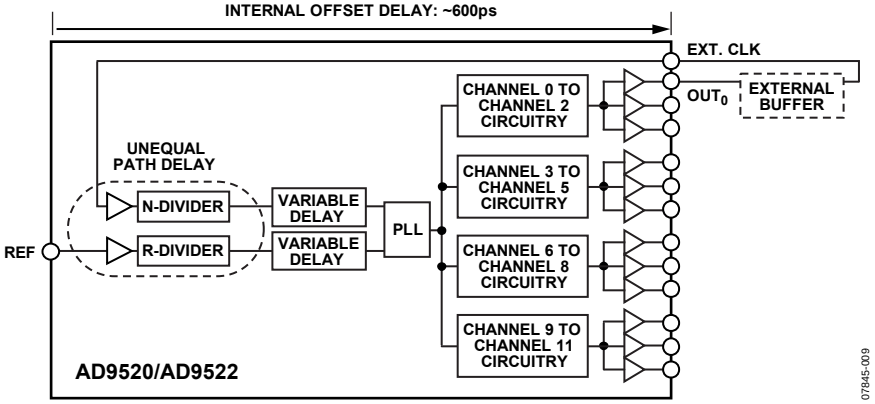

Figure 9. AD9520/AD9522 External Zero-Delay Timing Offset Error

Figure 10 and Figure 11 are actual time domain measurements demonstrating the virtual elimination of timing offset error using the external zero-delay architecture. The measurements were taken by using matched probes connected directly to the REF input and external clock pins of the device and with OUT_0 fed back to the external clock input (similar to Figure 9).

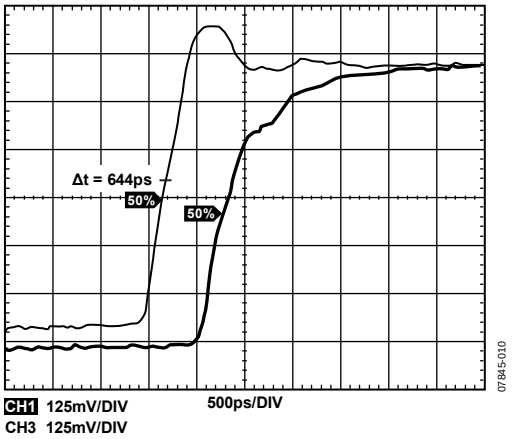

Figure 10. External Zero-Delay Inherent Offset Timing Error (AD9520)

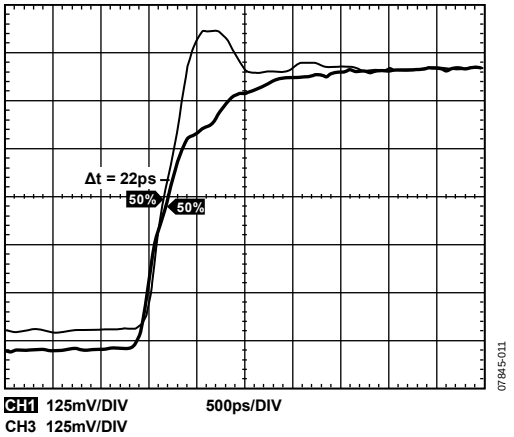

Figure 11. External Zero-Delay Adjusted Offset Timing (AD9520)

CONSIDERATIONS INVOLVING EXTERNAL ZERO-DELAY

In applications that require the external zero-delay architecture, be careful to avoid the introduction of excessive external delay and the coupling of unwanted signals into the PLL feedback path. External delay can result in PLL loop instability while unwanted signals injected into the PLL feedback path can corrupt the output clock signal(s).

Note that any delay appearing in the external signal path becomes part of the PLL feedback loop. If the PLL loop bandwidth is too wide, the additional delay in the external path could destabilize the loop. The user should choose the external loop filter components so that the loop bandwidth can accommodate the additional delay while still ensuring stability of the loop.

Because the external zero-delay architecture includes external circuitry in the PLL feedback path, the accidental injection of unwanted signals into the feedback path can pose a serious problem. The reason is that any signal injected into the PLL feedback path is subject to the gain of the PLL. Thus, even a very low level signal can grow to an intolerable level at the output. Proper printed circuit board layout and shielding techniques are the best defense against the intrusion of unwanted signals.

NOTES

AN07845-0-12/08(0)