Interface single-ended signals to DDR devices

Paul Li - November 28, 2002

Today's CPUs require high bandwidth on the pc board. As a result, differential signals, including DDR (double-data-rate) signals, are becoming common in ICs such as ASICs, clocks, and DRAM devices, and are gradually dominating pc-board design. Traditional TTL devices still remain, however, and system designers now face the task of interfacing the DDR devices with conventional single-ended signals. Numerous design guidelines make the job easier.

Working with differential signals helps to overcome the limitations of TTL signals. The TTL signal on a pc board becomes the bottleneck because the maximum frequency of TTL signals traveling through the board is less than 200 MHz. The large amplitude of TTL signals takes a longer time to swing between rails, and the high capacitance associated with TTL signals works in parallel with the bus structure to tremendously attenuate the edges of TTL signals and limits the maximum speed. TTL signals also consume high power at high speed due to the wide signal swing and high capacitance.

Differential receivers can reject common-mode noise and therefore can work at smaller amplitudes and at higher speeds. Differential signals have much lower capacitance due to their point-to-point (nonbus) structure. Thus, at low swing range and capacitance, differential signals can work at a much higher speed with low power consumption. The DDR clock also benefits from the double data rate, because it uses every edge as a clock.

An example of interfacing a single-ended signal to a DDR device uses a common DDR PLL clock buffer, the 855 (Figure 1). The series resistor $R_{TER1}$ is a termination resistor that matches the impedance of the trace between the clock driver, IC$_1$, and the DDR receiver, IC$_2$. The signal at a trace that has a well-matched impedance forms an ideal shape of rising and falling edges at the end of trace. There is no overshoot, undershoot, or significant attenuation on the edges if the capacitive load is slight.

Transmission-line theory shows that the impedance of a trace is matched when $R_{OUT} + R_{TER1} = R_{TRACE}$. $R_{OUT}$ is the output impedance of clock-driver IC$_1$, which is normally 8 to 45Ω. $R_{TER1}$ is the series-termination resistor, which is normally 0 to 40Ω. The location of this resistor should be as close to the output pin as possible. $R_{TRACE}$ is the impedance of the trace between clock-driver IC$_1$ and DDR-clock IC$_2$. Trace parameters-and pc-board-fabrication techniques control the impedance of the trace, which is normally 50 to 60Ω.
For example, if $R_{\text{OUT}}$ is 30Ω, and $R_{\text{TRACE}}$ is 50Ω, then matching the trace impedance requires that $R_{\text{TER1}}$ equals 20Ω.

**Reflection effect versus RC effect**

The transmission-line-reflection effect and the RC effect affect the shape of the reflected waveform at the end of a trace. When the impedance of a trace is well-matched, according to the previous equation, the reflection at the end of the trace forms a perfect rail-to-rail waveform without overshoot, undershoot, or edge attenuation (Figure 2, Trace A). Overshoot and undershoot appear proportionally when $R_{\text{OUT}}+R_{\text{TER1}}>R_{\text{TRACE}}$ (Trace B). Edge attenuation appears proportionally when $R_{\text{OUT}}+R_{\text{TER1}}<R_{\text{TRACE}}$ (Trace C).

The RC effect is the result of the RC circuit’s comprising $R_{\text{OUT}}$, $R_{\text{TER1}}$, the trace capacitance, and the capacitance of the DDR receiver. When the capacitive load is higher than 40 pF, the RC effect slows the edges and overrides the overshoot, undershoot, and attenuation (Figure 3).

The transmission-line-reflection effect determines the shape of waveforms at the end of traces when the capacitive load is less than 20 pF and frequency is less than 200 MHz. In this condition, matching the impedance is essential. The RC effect dominates the shape of waveforms when the capacitive load is higher than 40 pF and significantly slows the edges of signals.

Therefore, when matching a trace that has heavy capacitance, such as a bus structure and long trace or cable, properly reducing the output impedance from the matching value is a common approach to compensate the loss due to the RC effect, especially in a bus structure. An IBIS-model simulation is a good tool to help in a system designer’s signal-integrity analysis.

At frequencies higher than 500 MHz, the reactance of a capacitive load is much lower and tremendously attenuates the signal. Therefore, the major goal for a high-speed-circuit design is a capacitive load of less than 10 pF at 500 MHz with a matched impedance. The low capacitance associated with the point-to-point structure of a differential signal is one of the main reasons that differential signals dominate current pc-board design.

**Adjust feedback delay for system timing**

The feedback loop comprises the trace and circuit between $FB_{\text{OUT}}$ and $FB_{\text{IN}}$ in Figure 1. The main advantage of a PLL clock over a non-PLL clock is the PLL clock’s ability to adjust the system-clock timing by altering the feedback time delay. The timing relationships among the clock input, clock output, and the feedback loop determine the amount of leading or lagging skew (Figure 4).

The feedback edge of $FB_{\text{IN}}$ at time $T_2$ is always chasing and locking onto the clock-input edges of the CLK input at time $T_1$ (traces A and B). As long as the PLL is functioning properly, the time difference between $T_1$ and $T_2$ is zero ($T_1-T_2=0$). This principle mechanism is the reason that the PLL can generate a precise output leading skew, in which the output is earlier than the input, or lagging skew.

The feedback delay between $FB_{\text{OUT}}$ and $FB_{\text{IN}}$ or $T_2-T_3$, generates the same amount of output leading skew between $Y_X$ (X ranges from 0 to 4 in Figure 1) and CLK, or $T_1-T_4$, respectively, because the drivers of $FB_{\text{OUT}}$ and $Y_X$ are identical and connect to the same input. As long as the PLL is functioning, the following equation will always be true: $T_2-T_3=T_4-T_1$. And because $T_1=T_4$, the following is also true: $T_2-T_3=T_1-T_4$. Thus, by varying the time delay between $FB_{\text{OUT}}$ and $FB_{\text{IN}}$ ($T_2-T_3$), you can generate the same amount of output leading skew ($T_1-T_4$) at $Y_X$ (traces C and D).
For instance, when the feedback delay between \( FB_{OUT} \) and \( FB_{IN} \) (\( T_{2} - T_{3} \)) is zero, the output skew between \( Y_{X} \) and CLK (\( T_{1} - T_{4} \)) should be zero. If the feedback delay is 1.2 nsec, a 1.2-nsec leading skew occurs at \( Y_{X} \), and the output edge at \( Y_{X} \) is 1.2 nsec earlier than the edges at CLK. The feature of generating leading skew is useful for a PLL DDR clock; it provides the approach of system clock-timing adjustments.

**Feedback delay**

The delay of the trace in the feedback loop causes 70 to 85% of the feedback delay (\( T_{2} - T_{3} \), Figure 5). The speed of a signal traveling in a trace is 50 to 70% of the speed of light, which is approximately 6 to 8 in./nsec, depending on the type and parameter of the trace. The 70 to 85% trace-generated time delay provides a stable timebase. A common technique is to use a nonconnected second pair of traces at different lengths on the pc board for a spare timing option (Figure 5). You can jump-connect the second pair of traces to the feedback loop through pads if you choose to disconnect the first pair due to unsuitable timing after pc-board fabrication.

The RC circuit of \( R_{1}, R_{2}, C_{1} \), and the input capacitance of \( FB_{IN} \) should generate 15 to 30% of the feedback time delay. This 15 to 30% delay provides fine-tuning flexibility. Fine-tuning the values of \( R_{1}, R_{2}, R_{TER2}, \) and \( C_{1} \) after pc-board fabrication provides accurate system-clock timing. The default specification for \( R_{TOTAL} \), or \( R_{1} + R_{2} + R_{TER2} \), is 120Ω. Adding resistance of 0 to 22Ω to \( R_{1} \) and \( R_{2} \) and keeping \( R_{TOTAL} \) at 120Ω adds extra feedback delay due to the RC delay of \( R_{1}, R_{2}, \) and the input capacitance of \( FB_{IN} \) and \( C_{1} \), if any. The default value of \( C_{1} \) is 0 pF because capacitance slows the slew rate, which is undesirable. Adding capacitance of 2 to 10 pF to \( C_{3} \) generates more feedback delay. Reducing \( R_{TOTAL} \) reduces the peak-to-peak swing range. In this case, the signal swings faster in the smaller range, and, therefore, the feedback delay decreases. Similarly, increasing the swing range increases the feedback delay.

If the lengths of the output traces at outputs \( Y_{X} \) are not equal in the case when you want equal clock timings, you can match up the shorter traces with the longest trace with extra dummy routing. For instance, in Figure 5 there is a 1.2-nsec trace delay between \( Y_{0} \) (\( T_{4} \)) and the clock receiver, \( T_{5} \). If you want a zero delay between CLK and \( IC_{1} \), the edge at \( Y_{0} \) must occur 1.2 nsec earlier than CLK, \( T_{1} \), to compensate for the trace delay between \( Y_{0} \) and \( IC_{1} \). Therefore, you need a 1.2-nsec feedback delay between \( FB_{OUT} \) and \( FB_{IN} \) (\( T_{2} - T_{3} \)). Figure 6 shows a detailed timing relation.

In this example, the feedback trace generates 75%, or 0.9 nsec, of the 1.2-nsec feedback delay. The RC circuit comprising \( R_{1}, R_{2}, R_{TER2}, C_{1} \), and the input capacitance of \( FB_{IN} \) generates the remaining 25%, or 0.3 nsec. When you connect 3.3 and 5V single-ended CMOS drivers to a DDR receiver, you can use the circuit in Figure 7 and the values in the corresponding table help to fine-tune \( R_{i} \) for a 0 to 2.5V swing at \( V_{1} \). The value of \( R_{TER1} \) is 0 to 15Ω. Increase \( R_{TER1} \) if overshoot and undershoot exists at \( V_{i} \); reduce \( R_{TER1} \) if the signal appears attenuated. You can use any single output of a DDR output pair as a 2.5V single-ended clock driver to drive a 2.5V clock receiver (Figure 8).

**Adjust for duty-cycle balance**

DDR standards require more precise tolerance for duty-cycle balance than do normal TTL signals. Thus, you need to adjust the duty-cycle balance of the DDR output of a non-PLL DDR buffer when interfacing to a single-ended input signal (Figure 9). A DDR PLL clock-buffer has no duty-cycle balance issue because a DDR PLL clock-buffer is a PLL device; its feedback locks only to the input rising edge at CLK and ignores the duty cycle of the input signal. Therefore, its output duty cycle depends on the internal duty-cycle balance circuit, and you cannot adjust it externally. In Figure 9a, the \( R_{i}/R_{2} \) voltage divider determines the level at \( V_{1} \). Normally, \( V_{1} = V_{CC}/2 = 1.25V \). Adjusting the voltage
of $V_1$ can improve the duty cycle of the output clock (Figure 9b). The preferred values of $R_1$ and $R_2$ are 1 to 5 kΩ. Lower resistances cause higher dc current but provide better noise immunity, and vice versa.

Fine-tuning $R_1$ to adjust $V_1$ according to the following equation improves the output clock's duty cycle: $V_1 = V_{CC}(R_2/(R_1+R_2))$.

When adjusting the output clock's duty cycle, fix the resistance of $R_2$ at 2 kΩ. Then replace $R_1$ with a 3-kΩ variable resistor. Then, measure the crosspoints of the DDR output clock's duty cycle at $+Y_0$ and $-Y_0$ using two probes and adjusting $R_1$ until the output duty cycle is balanced. Carefully selecting the single-ended signal driver with a good duty cycle within 47 to 53% and minimal lot-to-lot variation is essential. It's also important to perform routine duty-cycle sampling tests in production for lot-to-lot or processing variation. This duty-cycle-adjustment technique is also suitable for other differential signals, such as LVDS (low-voltage differential signaling) and LVPECL (low-voltage pseudo-emitter-coupled logic).

References

3. JEDEC Standard No. 44, "Standard for definition of CU877 clock driver for registered DDR2 DIMM applications," JC40 Item No. 44, Revision 0.95; second showing: April 2002; May 3, 2002.