System Clock Delay for IDT 64-bit Microprocessors

64-bit Processors Application Note AN-300

Notes

By Seroj Babaian and Neven Matkovic

Revision History

May 21, 2001: Initial publication.

Overview

IDT has changed two parameters in the data sheet for all its 64-bit RISController microprocessor devices (RC4640, RC4650, RC4700, RC5000, RC64474, RC64475, RC64574, and RC64575) to better reflect the actual behavior of the devices. The two parameters changed are:

Tdoh: minimum output data hold time from 1 nsec to 0 nsec

Tdm: minimum output data time from 1 nsec to 0 nsec (for devices that had this parameter specified)

These changes are not expected to have an impact on most systems in which the devices are used. However, systems that use the Galileo Technology system controllers (the GT64xxx family)—*and* use the input clock to drive both devices—will be impacted by that change. The GT64xxx devices require a data input hold time of 1 nsec. It is very possible that the IDT 64-bit microprocessor may change the data before it is clocked in by the GT64xxx device, resulting in system failure. This scenario can affect existing as well as new designs.

Workaround

The easiest solution to implement is to delay the clock signal to the processor relative to the clock for the Galileo device (GT64xxx) in order to gain back the hold time. This can be accomplished in several ways:

- **passing the clock through a buffer**
- **inserting an RC delay**
- **using a ready made delay line**
- **using a programmable clock skew driver**
- **creating a delay line from a board trace.**

Of these, only the programmable clock skew driver and the board trace delay line are workable solutions. The exact amount of delay required depends on the system but is usually in the range of 1 ns.

Using the buffer method is unacceptable because of the large variation between minimal and maximal delay of a buffer chip, even within the same batch. Variations between different batches could be even larger.

The RC delay method can introduce signal distortion, thus causing other problems.

Ready-made delay lines are not targeted for low delay applications.The minimal delay for these devices begins in the range of 5 to 10 ns, and they are therefore unsuited to this application.

Notes

Programmable Skew Clock Driver

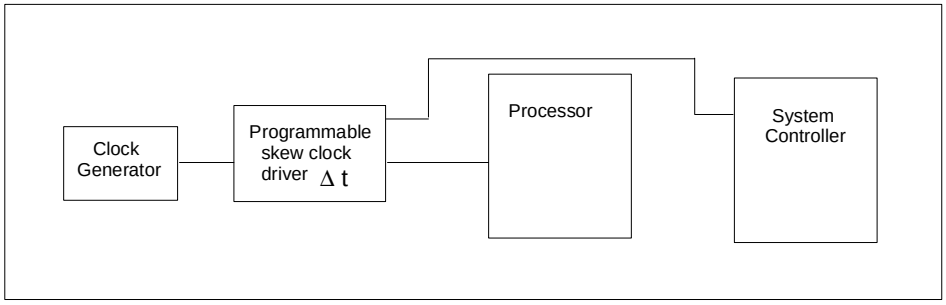

Figure 1 System Block Diagram with Programmable Skew Driver

A programmable clock skew driver (IDT5T995A Turboclock II for example) can also be used to create this delay. The basic time increment of the IDT part is 625ps. Two time units of skew should create exactly the right amount of hold time.

Clock lines should be terminated with series or AC termination. A typical value for AC termination is 47 Ω, with capacitor values between 47 and 100 pF. (These are good starting values, but exact values need to be determined by simulation or experiment.)

Because this is a PLL based device, due attention should be paid for decoupling power pins because noise on the power line could introduce jitter in the clock signals.

Delay Line

The solution with the delay line is very simple, requires only limited board space, and does not require any additional components.

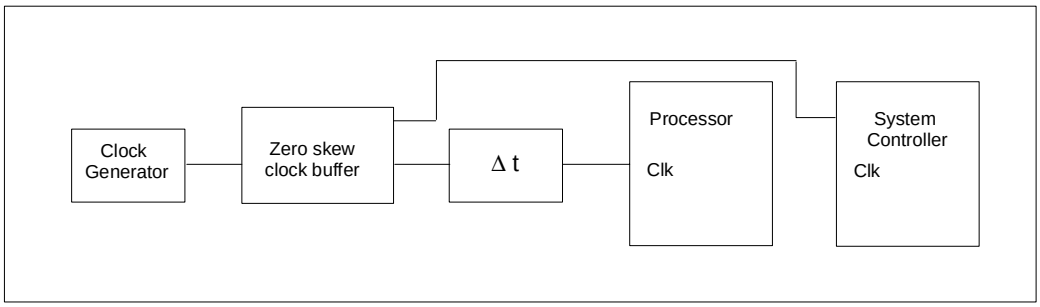

Figure 2 System Block Diagram with Delay Line

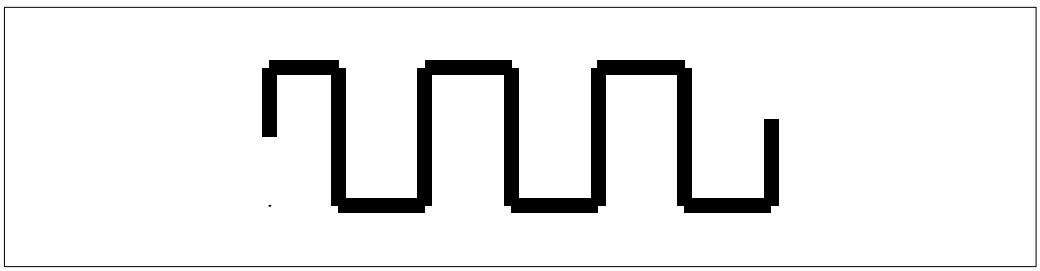

Figure 3 PCB Trace Delay Time

Approximate formula for calculating delay of a PCB trace is given by the formula:

$t_{pd} = 1.017\ sqrt(a\ \varepsilon_r + b)$

Where:

Notes

t_{pd} = propagation delay of the trace in ns/ft

ε_r = relative dielectric coefficient (typical value is 4.2 for FR-4; usual range is between 4.0 and 4.4 or more, depending on the material)

a=1 b=0 for Stripline

a=0.475 b=0.67 for Microstrip

This formula is only an approximation that is sufficient for practical purposes. For exact values, it is necessary to run a simulation using Spice or Hyperlinx.

It is advisable to use a low-skew clock buffer (like IDT QS5810) so that one output is driving only the delay line. This permits proper termination of the processor clock with either series termination at the clock buffer or Thevenen termination at the clock input of the processor. Typical values for series termination are between 22 and 33 ohms.

Conclusion

It is recommended that these fixes be applied immediately to all systems using the 64-bit RISController and any of the GT64xxx system controllers.