


## Introduction

On-chip series termination ( $R_S$  OCT) improves signal integrity and I/O performance due to optimized impedance matching. On-chip series termination eliminates the need for external series termination resistors and simplifies the design of a printed circuit board (PCB).

$R_S$  OCT is implemented through output driver impedance control. The output driver of an I/O standard consists of a group of transistors in parallel. Each transistor can be individually enabled or disabled to control the output driver impedance to achieve the desired OCT value.

 Stratix® II devices support both calibrated  $R_S$  OCT and non-calibrated  $R_S$  OCT.



For more information on I/O types and banks that support  $R_S$  OCT, refer to the *Selectable I/O Standards in Stratix II Devices* chapter of the *Stratix II Handbook*. For more information on  $R_S$  OCT tolerance specifications, refer to the *DC & Switching Characteristics* chapter of the *Stratix II Handbook*.

This application note describes calibrated Stratix II  $R_S$  OCT support and design implementation (e.g., DDR and DDR2 using SSTL and TTL I/O standards, which require termination for impedance matching).

## Calibrated On-Chip Series Termination in Stratix II Devices

Stratix II devices only support calibrated  $R_S$  OCT in column I/O banks. The calibration can occur in either configuration mode or user mode. User-mode calibration is the super set of configuration-mode calibration and has greater tolerance than configuration-mode calibration.

### Configuration-Mode Calibration

Configuration-mode calibration is the default mode for calibrated  $R_S$  OCT and is performed automatically in the device configuration stage. No user control is involved. Once the device is configured, the calibration for OCT is complete. No further calibration can be initiated in this mode.

If temperature or voltage changes over time after the device configuration stage, the  $R_S$  OCT tolerance changes as well.

### User-Mode Calibration

User-mode calibration is the super set of configuration-mode calibration whereby calibration occurs in both configuration mode and user mode.

In the device configuration stage, user-mode calibration works exactly the same way as the configuration-mode calibration. No user control is involved.

After the device is configured, designers can dynamically control user-mode calibration. If temperature or voltage changes over time, designers can recalibrate to maintain a tight tolerance. There is no limit to the number of times user-mode calibration can be initiated.

One calibration cycle takes 256 clock cycles (the clock is the clock input signal `in_clkusr` for the calibration circuitry. See [“Implementing User-Mode Calibration” on page 2–4](#) for details). Once the calibration circuit finds the correct impedance, it stops changing the characteristics of output drivers.

User-mode calibration can be initiated at anytime after the device is configured. The device is not required to stop real-time operation during the calibration. However, due to the dynamic nature of the calibration, there will be a small perturbation to output drivers with calibrated  $R_S$  OCT enabled during the calibration. Altera recommends that you initiate configuration while the device is not sending data traffic. Calibration takes 256 clock cycles (6.4  $\mu$ s for a 40-MHz clock). The calibration process does not affect any input operations.

## Implementing Calibrated $R_S$ OCT in Stratix II Devices

This section explains how to implement calibrated  $R_S$  OCT in Stratix II devices.

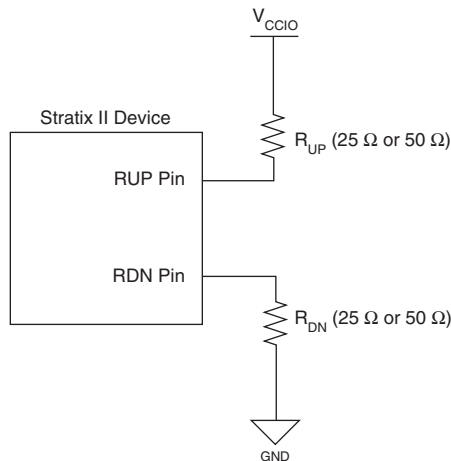
### Placing $R_{UP}$ & $R_{DN}$ Resistors on the Board

The  $R_{UP}$  and  $R_{DN}$  resistors are reference resistors used by the calibration circuitry to control the value of the series resistance. They determine whether you are trying to achieve 50- or 25- $\Omega$  series resistance. The calibration circuit relies on  $R_{UP}$  and  $R_{DN}$  for accurate on-chip series termination.

Stratix II devices have two separate sets of calibration circuits: one calibration circuit for top banks 3 and 4; and one calibration circuit for bottom banks 7 and 8. One pair of  $R_{UP}$  and  $R_{DN}$  pins exist in bank 4 for the calibration circuit for banks 3 and 4. Similarly, there is one pair of  $R_{UP}$  and  $R_{DN}$  pins in bank 7 for the calibration circuit for banks 7 and 8. Refer to the Quartus® II software for locations of  $R_{UP}$  and  $R_{DN}$  pins in Stratix II devices.

To use calibration circuits, connect the  $R_{UP}$  pin to  $V_{CCIO}$  through an external 25- $\Omega$  or 50- $\Omega$  resistor ( $R_{UP}$ ). This provides a  $R_S$  OCT value of 25  $\Omega$  or 50  $\Omega$ , respectively, as shown in figure 1. Connect the  $R_{DN}$  pin to ground through an external 25- $\Omega$  or 50- $\Omega$  resistor ( $R_{DN}$ ). Both  $R_{UP}$  and  $R_{DN}$  are required for correct operation. This provides a  $R_S$  OCT value of 25  $\Omega$  or 50  $\Omega$  respectively, as shown in Figure 1.

**Figure 1. External  $R_{UP}$  &  $R_{DN}$  Connections on a PCB**



$R_{UP}$  and  $R_{DN}$  pins are dual-purpose I/O pins. If they are not used for  $R_S$  OCT calibration purposes, they can be used as general-purpose I/O pins.

During the calibration, a current flows through the external pull-up and pull-down resistors. The maximum current is calculated as follow:

$$I = 0.5 \times V_{CCIO} / R_S$$

For example, for a 1.5-V output with 50- $\Omega$   $R_S$  OCT requires the following current:

$$I = 0.5 \times 1.5 \text{ V} / 50 \text{ } \Omega = 15 \text{ mA.}$$


Use resistors with proper power ratings to ensure they can stand the maximum current. No current flows through the external pull-up and pull-down resistors when the  $R_S$  OCT is not being calibrated.

Altera recommends the following guidelines for optimal  $R_S$  OCT performance:

- Use 1% discrete resistors for  $R_{UP}$  and  $R_{DN}$
- Use wide and short traces for  $R_{UP}$  and  $R_{DN}$  resistor connections on the board

### Enabling Series OCT with Calibration for an I/O Pin in the Quartus II Software

To enable series OCT with calibration for an I/O pin, using the Assignment Editor, make a **Termination - Stratix II/HardCopy II** assignment with a value of **Series 50 Ohms with Calibration** or **Series 25 Ohms with Calibration** for the pin.

 Since configuration-mode calibration is performed automatically during the device configuration stage, no further steps are required.


### Implementing User-Mode Calibration

You must use the `octcal` function to use user-mode calibration. Call the `octcal` function explicitly in your design to control each calibration circuit in real time. You must include additional files to the design project directory as well. Refer to the appendix for reference designs in AHDL, Verilog HDL, and VHDL.

The `octcal` function is defined as:

```
FUNCTION octcal (in_clkusr, in_clkenusr, in_pinrup, in_pinrdn)
    RETURNS (dummy_out);
```

Four inputs and one output exist. Connect the `dummy_out` output signal to an output pin at the top level of a design compiled in the Quartus II software version 4.2 Service Pack 1.

 For versions of the Quartus II software greater than version 4.2 Service Pack 1, you do not need to connect the `dummy_out` output signal to an output pin.

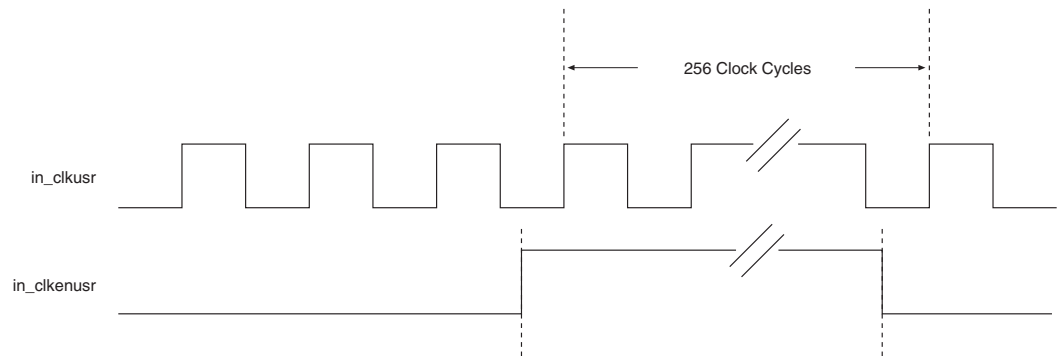
The `in_clkusr` input signal is the user clock input to the calibration circuit. This clock has no minimum frequency requirement. The clock's maximum frequency is:

- 40 MHz for 3.3-V, 2.5-V, and 1.8-V I/O standards
- 20 MHz for 1.5-V I/O standards

The active high signal `in_clkusr` is the user clock enable input to the calibration circuit. It is used to start and stop the calibration, and must be asserted for 256 clock cycles. This enable signal must be synchronized with the signal `in_clkusr` user clock.

Figure 2 shows the timing relationship between the clock signal and the clock enable signal.

**Figure 2. Timing Diagram for Signals `in_clkusr` & `in_clkusr`**



`in_clkusr` and `in_clkusr` input signals to the calibration circuit can come from internal logic or external input pins. Altera recommends using the `in_clkusr` clock signal to generate the synchronous `in_clkusr` clock enable signal in the FPGA to avoid any setup or hold time concerns.

Use the `RUP` and `RDN` pins, respectively, to drive the `in_pinrup` and `in_pinrdn` input signals. Do not connect the `RUP` and `RDN` pins to any other internal logic in a design.

When using  $R_S$  OCT, you must also setup the `use_high_voltage_compare` parameter in the `octcal.tdf` file in the project directory according to the following guidelines:

- For 3.3-V and 2.5-V  $V_{CCIO}$ , set this parameter to “true”
- For 1.8-V and 1.5-V  $V_{CCIO}$ , set this parameter to “false”

You can start the user-mode calibration process using the two input signals `in_clkusr` and `in_clkenusr` at any time after the device is configured. The calibration process takes 256 clock cycles.



Contact Altera Applications for reference designs in AHDL, Verilog HDL, and VHDL.

## Appendix A: User-Mode Calibration Reference Design in AHDL

Appendix A shows only the top-level design file compiled in the Quartus II software version 5.0 and the names of other files required in the project directory.

The following sample is the top-level design file, **topLevel.tdf**, in an AHDL design:

```
INCLUDE "octcal.inc";

subdesign topLevel
(
    in_clkusr:input;
    in_clkenusr:input;
    in_pinrup :input;
    in_pinrdn:input;
    testpinin[1..0]:input;
    testpinout[1..0]:output;
)

variable
octcal_inst: octcal;

BEGIN
    octcal_inst.in_clkusr   = in_clkusr;
    octcal_inst.in_clkenusr = in_clkenusr;
    octcal_inst.in_pinrup   = in_pinrup;
    octcal_inst.in_pinrdn  = in_pinrdn;

    testpinout[0] = testpinin[0];
    testpinout[1] = testpinin[1];
END;
```

Other required files in the project directory are:

- **octcal.inc**
- **octcal.tdf**

## Appendix B: User-Mode Calibration Reference Design in Verilog HDL

Appendix B shows only the top-level design file compiled in the Quartus II software version 5.0 and the names of other files required in the project directory.

This sample is the top-level design file, **topLevel.v**, in a Verilog HDL design:

```
module topLevel
(
    in_clkusr,
    in_clkenusr,
    in_pinrup,
    in_pinrdn,
    testpinin;
    testpinout
);

    input        in_clkusr;
    input        in_clkenusr;
    input        in_pinrup;
    input        in_pinrdn;
    input[1:0]   testpinin;
    output[1:0]  testpinout;

    octcal octcal_inst
    (
        .in_clkusr(in_clkusr),
        .in_clkenusr(in_clkenusr),
        .in_pinrup(in_pinrup),
        .in_pinrdn(in_pinrdn)
    );

    assign testpinout[0] = testpinin[0];
    assign testpinout[1] = testpinin[1];

endmodule
```

Other required files in the project directory are:

- **octcal.inc**
- **octcal.tdf**

## Appendix C: User-Mode Calibration Reference Design in VHDL

Appendix C shows only the top-level design file compiled in the Quartus II software version 5.0 and the names of other files required in the project directory.

This sample is the top-level design file, **topLevel.vhd**, in a VHDL design:

```
LIBRARY ieee;
USE ieee.std_logic_1164.ALL;

ENTITY topLevel IS
    PORT
    (
        in_clkusr      :IN STD_LOGIC:= '0';
        in_clkenusr    :IN STD_LOGIC:= '0';
        in_pinrup      :IN STD_LOGIC:= '0';
        in_pinrdn      :IN STD_LOGIC:= '0';
        testpinin      :IN STD_LOGIC_VECTOR (1 DOWNTO 0);
        testpinout     :OUT STD_LOGIC_VECTOR (1 DOWNTO 0)
    );
END topLevel;

ARCHITECTURE arch OF topLevel IS

    COMPONENT octcal
        PORT
        (
            in_clkusr: IN STD_LOGIC:= '0';
            in_clkenusr: IN STD_LOGIC:= '0';
            in_pinrup: IN STD_LOGIC:= '0';
            in_pinrdn: IN STD_LOGIC:= '0'
        );
    END COMPONENT;

    BEGIN
        octcal_inst : octcal
            PORT MAP
            (
                in_clkusr => in_clkusr,
                in_clkenusr => in_clkenusr,
                in_pinrup => in_pinrup,
                in_pinrdn => in_pinrdn
            );

        testpinout(0) <= testpinin(0);
        testpinout(1) <= testpinin(1);

    END arch;
```

Other required files in the project directory are:

- **octcal.inc**
- **octcal.tdf**



101 Innovation Drive  
San Jose, CA 95134  
(408) 544-7000  
[www.altera.com](http://www.altera.com)  
**Applications Hotline:**  
(800) 800-EPLD  
**Literature Services:**  
[literature@altera.com](mailto:literature@altera.com)

Copyright © 2005 Altera Corporation. All rights reserved. Altera, The Programmable Solutions Company, the stylized Altera logo, specific device designations, and all other words and logos that are identified as trademarks and/or service marks are, unless noted otherwise, the trademarks and service marks of Altera Corporation in the U.S. and other countries. All other product or service names are the property of their respective holders. Altera products are protected under numerous U.S. and foreign patents and pending applications, maskwork rights, and copyrights. Altera warrants performance of its semiconductor products to current specifications in accordance with Altera's standard warranty, but reserves the right to make changes to any products and services at any time without notice. Altera assumes no responsibility or liability arising out of the application or use of any information, product, or service described herein except as expressly agreed to in writing by Altera Corporation. Altera customers are advised to obtain the latest version of device specifications before relying on any published information and before placing orders for products or services.



Printed on recycled paper



I.S. EN ISO 9001