RS232 Reference Project

Revision: July 25, 2008

This document was produced by Digilent Romania. For questions, contact support@digilent.ro.

Overview

This document describes the VHDL implementation of the RS232 reference project. The RS232 reference project files are DataCntrl.vhd and RS232RefComp.vhd. The reference project instantiates the RS232 Reference Component, which communicates data with an RS232 terminal (like HyperTerminal) with a baud rate of 9600.

To run this reference project, the following Digilent components are needed:

- an FPGA-based system board
- a PmodRS232 module (if the system board does not have an on-board RS232 port)

Functional Description

Figure 1 shows the block diagram of the RS232 reference project.

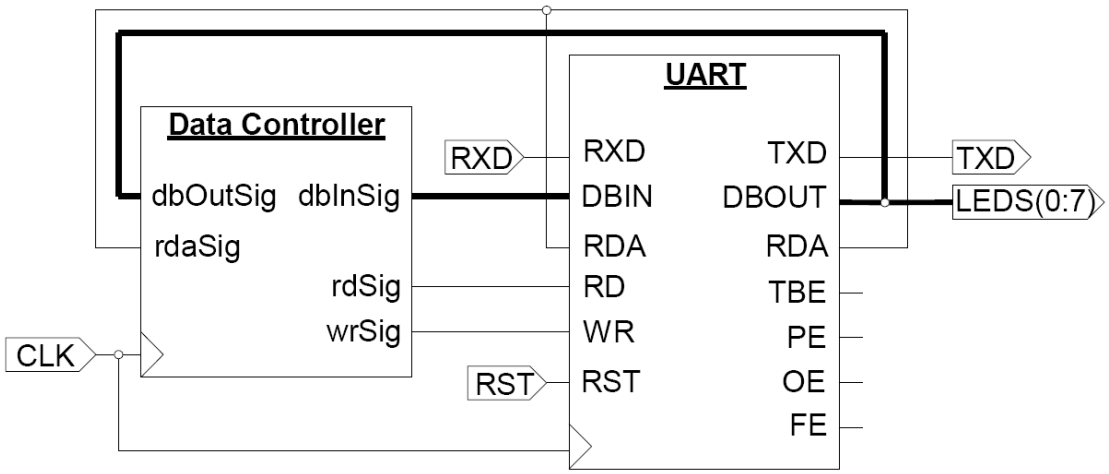

Figure 1 ***The RS232 Reference Project***

When loaded on a Digilent system board, serial data received from the UART is converted into parallel data, which is then displayed on the eight LEDs on the system board. This newly converted parallel data is also reconverted to serial data and transmitted back through RS232. For example, keyboard scan codes sent out through the HyperTerminal at 9600 baud rate will be seen on the LEDs and as well displayed on the HyperTerminal.

NOTE: If there seems to be a delay in transmission, make sure that the parity bit being used for transmission matches the parity the UART is programmed to use (default is odd parity).

Port Descriptions

CLK	50MHz clock input
RST	logic signal which, when active, resets the UART transmission
TXD	RS232 transmission data line
RXD	RS232 reception data line
LEDS(7:0)	LED outputs

RS232 Component

Please refer to the *RS232 Reference Component* manual for more detailed information on the UART component.

Data Controller

The data controller contains the state machine that controls the flow of data around the UART, allowing for data to be changed from serial to parallel, and then back to serial.

Set-Up

The functionality of this project can be demonstrated using an FPGA system board which has at least eight LEDs and a 50MHz clock.

To set up the hardware:

1. Make sure that the clock frequency select jumper on the board (if there is one) is set to the 50MHz position.
2. If the FPGA system board has no RS232 port, connect a PmodRS232 to one of the six pin connectors on the board.

To set up the software:

1. Create a new Xilinx ISE 9.2 project.
2. Place the VHDL files into the project.
3. Create an UCF file and make the following connections:
    - *CLK* connect to a 50MHz clock pin
    - *RST* connect to one of the buttons on the FPGA system board
    - *TXD* connect to the TXD line
    - *RXD* connect to the RXD line
    - *LEDS(7:0)* connect to the eight LEDs on the FPGA system board
4. Synthesize the project and generate the programming file (*.bit).
5. Use Digilent's Adept software to program the resulting *.bit file into the FPGA. Please refer to *Digilent Adept Reference Manual* for more detailed information.