Lab 9: Video Interfaces: HDMI and DVI

EE-459/500 HDL Based Digital Design with Programmable Logic

Electrical Engineering Department, University at Buffalo

Last update: Cristinel Ababei, October 27 2012

1. Objective

The objective of this lab is to learn how to transmit High-Definition Multimedia Interface (HDMI) and Digital Visual Interface (DVI) data streams to HDMI and DVI capable monitors. The top-level design in this lab displays a simple colored pattern. In addition, we will learn how to create ISE WebPack projects that use both VHDL and Verilog source files.

2. Introduction

The Atlys board contains four HDMI ports (see Fig.1), including two buffered – via the TI's TMDS141 buffers – HDMI input/output ports (**type A** connector), one buffered HDMI output port (**type D** connector), and one unbuffered port that can be input or output [1].

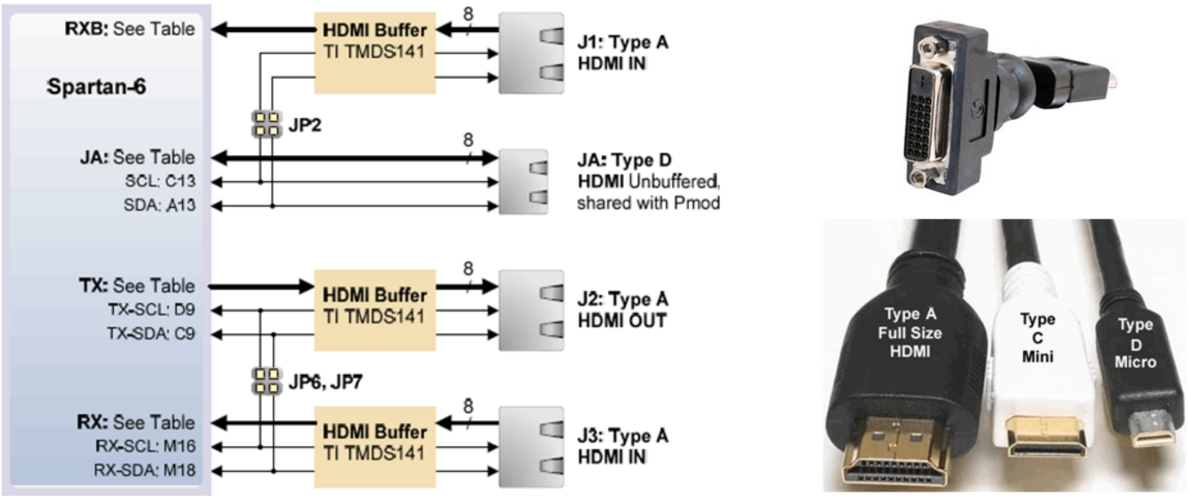

Figure 1 Illustration of the four HDMI ports of the Atlys board (left). HDMI Male to DVI-D Female Rotating Adapter (top right). HDMI connectors (bottom right)

Since the HDMI and DVI systems use the same transition-minimized differential signaling TMDS signaling standard, a simple adaptor shown in the right hand side of Fig.1 (available at most electronics stores such as TigerDirect [2]) can be used to drive a DVI connector from either of the HDMI output ports. The HDMI connector does not include VGA signals, so analog VGA displays cannot be driven. The Atlys board does not have any VGA connector. For examples on how to drive VGA monitors from the Atlys board, please see the supplemental material of this lab on the course's website.

In this lab we will drive HDMI and DVI capable monitors to display a colored pattern. For this we'll use a Verilog project developed by Bob Feng of Xilinx [3]. We'll modify Bob's project by converting to VHDL part of the Verilog code. In this way, this lab becomes a good opportunity for you to create ISE projects that use both Verilog and VHDL source files. By comparing two files, Verilog and VHDL, that implement the same functionality you get a fist time exposure to Verilog too.

*Transition-minimized differential signaling (TMDS)***:**

TDMS is a **method for transmitting high-speed serial data** and is used by the DVI and HDMI video interfaces, as well as other digital communication interfaces.

- Developed by Silicon Image Inc. as a member of the Digital Display Working Group
- Transmitter incorporates an advanced coding algorithm which reduces electromagnetic interference over copper cables and enables robust clock. Recovery at the receiver to achieve high skew tolerance.
- TMDS uses 4 channels: Red, Green, Blue, Clock
- TMDS is a two-stage process. Converts an input of 8 bits into a 10 bit code
    - TMDS signaling uses a twisted pair for noise reduction.
    - Current Mode Logic (CML), DC coupled and terminated to 3.3 Volts.
    - 3 twisted pairs are used to transfer video data - each a different RGB component
    - 8 bit data transmission plus 2 bits of control signals

3. Brief HDMI Description

What s HDMI:

- HDMI is the first & only industry supported, **uncompressed, all-digital audio/video interface**.
- HDMI is a compact audio/video interface for transferring uncompressed digital audio/video data from an HDMI-compliant device ("the source") to a compatible digital audio device, computer monitor, video projector, and digital television.
- HDMI provides an interface between any A/V source, such as a set-top box, DVD player, or A/V receiver and an audio and/or video monitor, such as a digital television (DTV), over a single cable.
- HDMI is a digital replacement for existing analog standards such as composite video, S-Video, SCART, component video, and VGA.
- HDMI supports standard, enhanced, or high-definition video, plus multi-channel digital audio on a single cable.
- Transmits all ATSC HDTV standards and supports 8-channel, 192kHz, uncompressed digital audio, all currently-available compressed formats & lossless digital audio formats with bandwidth to spare to accommodate future enhancements and requirements
- HDMI acts like Cat5, it passes a data signal not an RF signal like CATV.
- DVI is HDMI without the audio - separate cable needed for audio!

HDMI communication channels (see Fig.2):

HDMI has three physically separate communication channels, which are the TMDS, DDC, and the optional CEC:

- The HDMI cable and connectors carry four differential pairs that make up the TMDS data and clock channels.
    - Audio, video and auxiliary data is transmitted across the three TMDS data channels.
    - A TMDS clock, typically running at the video pixel rate, is transmitted on the TMDS clock channel
- HDMI carries a VESA DDC (Display Data Channel) channel. The DDC is used for configuration and status exchange between a single transmitter and a single receiver.
    - The DDC is used by the transmitter to read the receiver’s Enhanced Extended Display Identification Data (E-EDID) in order to discover the receiver’s configuration and capabilities.
- The optional CEC (Consumer Electronics Control) protocol provides high-level control functions between all of the various audiovisual products in a user’s environment.

Advantages of HDMI:

- Because HDMI is a digital interface, it provides the best quality of the video since there are no lossy analog to digital conversions as are required for all analog connections (such as component or S-Video).
- Digital video will be sharper.
- Single cable for both video and audio is the most effective format!
- HDMI devices supporting High-bandwidth Digital Content Protection (HDCP) have the comfort of knowing they will have access to premium HD content now and in the future.

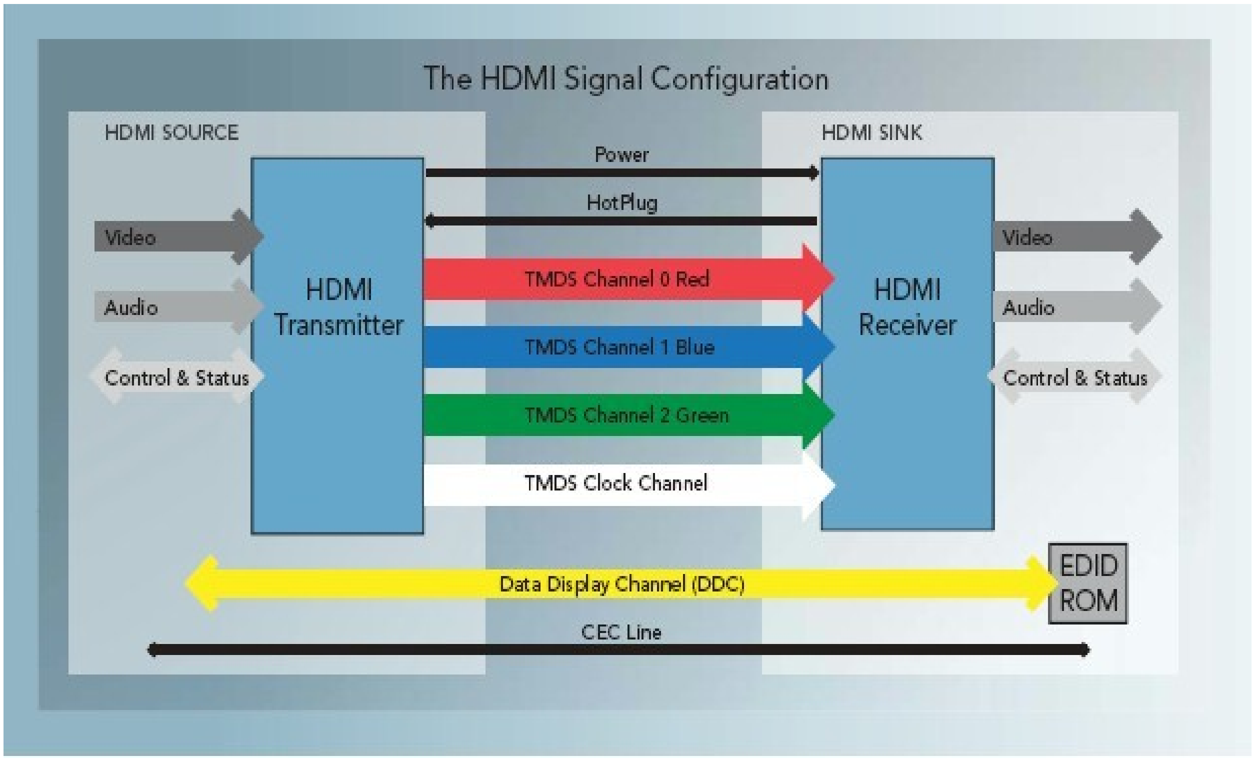

Figure 2 HDMI Signals

Compatibility with DVI:

HDMI is backward-compatible with single-link Digital Visual Interface digital video (DVI-D or DVI-I, but not DVI-A). No signal conversion is required when an adapter or asymmetric cable is used, so there is no loss of video quality. From a user's perspective, a DVI-D monitor would have the same level of basic interoperability unless there are content protection issues with High-bandwidth Digital Content Protection (HDCP), not supported by DVI, or the HDMI color encoding is in component color space YCbCr which is not supported by DVI, instead of RGB.

Because discussing HDMI is not the main purpose of this lab, you may want to take some time to search and read more about HDMI on the Internet. There is tons of information out there. Here are some starting pointers [5].

4. ISE WebPack Project

Xilinx Application Note

One challenge of working with HDMI is that there is no or only a limited number of design examples in the public domain. So, one would need to do many things from scratch, which becomes very challenging in the case of HDMI. However, a design example is the XAPP495 [3]. This is a good start as it was created to work with the Atlys board. But there are a few issues. One is that, generally, HDMI design requires significant effort and attention to many details. Another one is that it does not implement EDID (monitor identification data) nor audio. These will not stop us from using it. However, XAPP495 is written in Verilog while in this course we focus on VHDL. Hence, to put this lab together required some code conversion from Verilog to VHDL. Before continuing, you should take some time now and read XAPP495 paper [3] (also included in the downloadable archive of this lab).

Driving DVI and HDMI Monitors to Display a Colored Pattern

While the XAPP495 provides examples of both DVI transmitters and receivers, in this lab, we focus only on the transmitter part. Specifically, we create a design that drives HDMI and DVI monitors to display a colored bar pattern. The design basically uses all Verilog files related to the transmitting part from the xapp495 archive (downloadable from Xilinx). The only primary exception is the top-level file, which I replaced with a VHDL version of it. During the conversion Verilog-VHDL process, I had to make some other minor changes inside **syncro.v** and **serdes_n_to_1.v** to work around the fact that apparently **parameters** in Verilog modules cannot be instantiated as **generics** in VHDL (at least not with the ISE WebPack?). The VHDL top-level file is simply a VHDL counterpart of the **vtc_demo.v** file from the xapp495 archive. The name of the new VHDL top-level file is **vtc_demo.vhd**.

At this time, you should open both files **vtc_demo.vhd** (located in **lab9_files_ISE** folder) and **vtc_demo.v** to read and compare them. *Notice similarities and differences between VHDL and Verilog. Also, notice how Verilog modules are declared as components and instantiated in the VHDL top-level file. Comments inserted inside **vtc_demo.vhd** provide additional information on the main elements of the design.*
The block diagram of the design entity described in **vtc_demo.vhd** is shown in Fig.3. While reading the VHDL top-level file, try to identify the signals and components corresponding to those from Fig.3.

The design is based on several IP cores that are available on the Spartan-6 FPGA [5]. These include (see [5] for description of each):

- IBUF - input buffer
- BUFIO2 - Dual Clock Buffer and Strobe Pulse
- BUFG - Global Clock Buffer
- SRL16E - 16-Bit Shift Register Look-Up Table (LUT) with Clock Enable
- OSERDES2 - Dedicated IOB Output Serializer
- DCM_CLKGEN - Digital Clock Manager
- PLL_BASE - Basic Phase Locked Loop Clock Circuit
- BUFPLL - PLL Buffer
- OBUFDS - 3-State Differential Signaling I/O Buffer with Active Low Output Enable

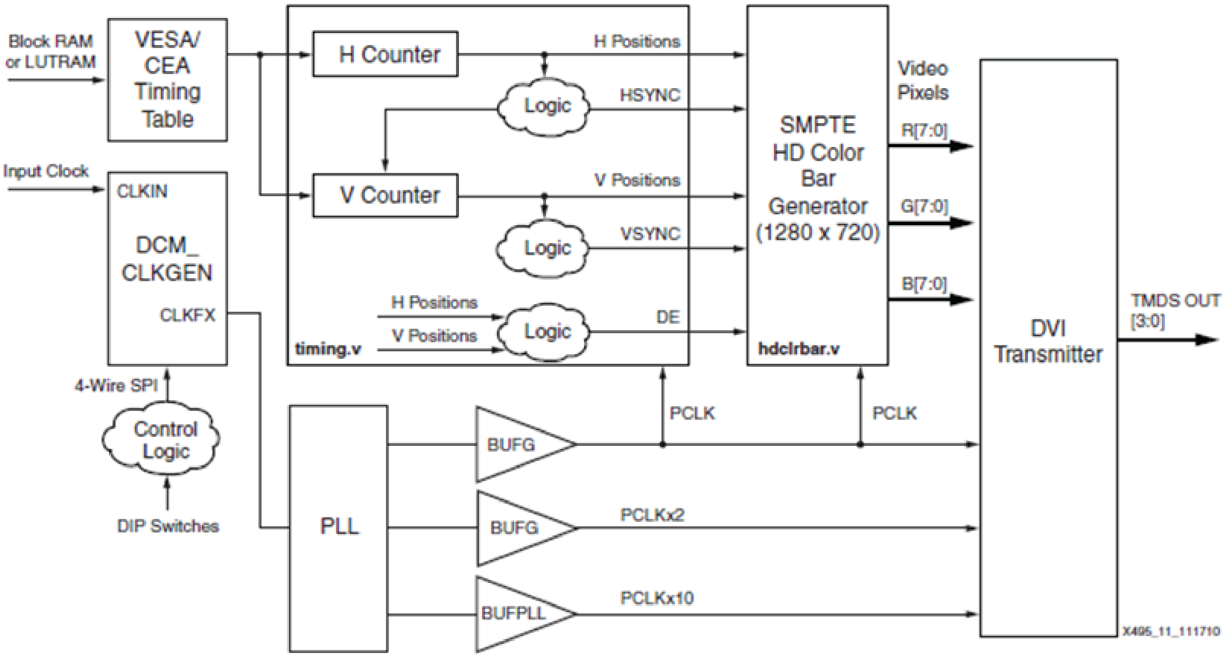

Figure 3 Block diagram of the "SMPTE HD Color bar Generation with Programmable Video Timing" designed by Bob Feng of Xilinx.

Create a new ISE WebPack project and add to it all the Verilog and VHDL files in **lab9_files_ISE** folder. These files together with other useful files (such as the .ucf file) are included in the downloadable archive with all the data for this lab. Synthesize and implement the design. Download bitstream to the FPGA board and test. To test the design, we need to attach a monitor to the HDMI OUT (J2) port of the Atlys board. An HDMI monitor can be connected directly using an HDMI cable. To connect a DVI monitor (like most of today's monitors) we need an HDMI to DVI converter; I got mine from TigerDirect [3] for $10. For this lab, the TA will have one such converter for you to take turns and use; however, if your project in this course involves using a monitor, you may want to buy your own converter.

After setting everything up, you should see your monitor display the colored pattern shown in Fig.4.

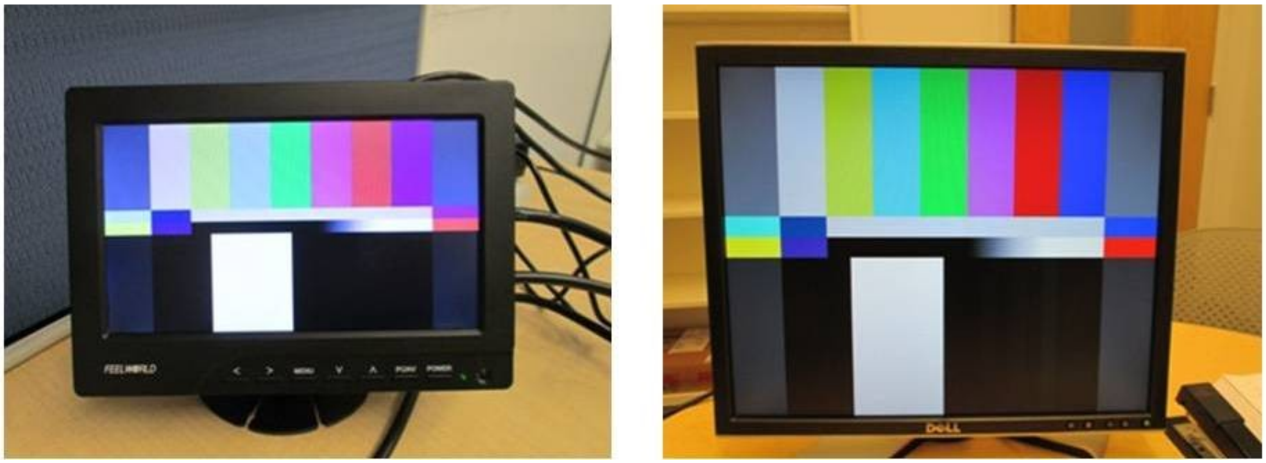

Figure 4 HDMI (left, 7" TFT) and DVI (right, 20" LCD) monitors display colored bar pattern

5. Lab Assignment

Convert to VHDL the **hdcolorbar** module describes in **hdclrbar.h** file. You must create a new VHDL file **hdclrbar.vhd** inside which you must describe the **hdcolorbar** design entity in VHDL (similarly to how I converted the top level Verilog module to top level VHDL entity). Then use the VHDL file to replace the Verilog file in the ISE WebPack project.

Optional (this is very challenging; do not attempt before talking to the instructor): implement the whole design in VHDL. Moreover, design and describe in VHDL your own entities for as many IP cores as possible. That is create your own VHDL entities to replace SRL16E , BUFPLL, etc. to improve the portability of the design to other FPGAs.

6. Credits and references

[1] Digilent Atlys board reference manual;
http://www.digilentinc.com/Data/Products/ATLYS/Atlys_rm.pdf
[2] CTG HDMI Male to DVI-D Female Rotating Adapter; from TigerDirect;
http://www.tigerdirect.com/applications/SearchTools/item-details.asp?EdpNo=3444838&CatId=467
[3] Bob Feng, Implementing a TMDS Video Interface in the Spartan-6 FPGA, Xilinx app note;
http://www.xilinx.com/support/documentation/application_notes/xapp495_S6TMDS_Video_Interface.pdf
[4] HDMI and DVI pointers:
--HDMI resource center; http://www.hdmi.org/learningcenter/
--Wikipedia HDMI introduction; http://en.wikipedia.org/wiki/HDMI
--HDMI specification document Version 1.3; see file included in this lab archive;
--HDMI Hider: TI TMDS141 (datasheet of the chip on the Atlys board);
http://www.ti.com/lit/ds/symlink/tmds141.pdf (also included in the archive of this lab);
--HDMI connectors A,B pinouts; http://pinouts.ru/Video/hdmi_pinout.shtml
--Wikipedia DVI introduction; http://en.wikipedia.org/wiki/Digital_visual_interface
--DVI 10; http://www.ddwg.org/lib/dvi_10.pdf
--DVI pinouts; http://pinouts.ru/Video/dvi_pinout.shtml
--Wikipedia RMDS introduction; http://en.wikipedia.org/wiki/Transition-minimized_differential_signaling
[5] Spartan-6 Libraries Guide for HDL Designs (BUFG, BUFIO2, SRL16E, PLL_BASE, etc.);
http://www.xilinx.com/support/documentation/sw_manuals/xilinx12_4/spartan6_hdl.pdf (also included in the archive of this lab);