

## ***1 Summary***

Subsystem testing instructions for HDMI transmit, HDMI receive, audio codec, USB and Ethernet.

## ***Contents***

<b><i>1</i></b>	<b><i>Summary</i></b>	<b><i>1</i></b>
<b><i>2</i></b>	<b><i>HDMI Transmit/Receive</i></b>	<b><i>2</i></b>
<b><i>2.1</i></b>	<b><i>Enable HDMI Transceivers</i></b>	<b><i>2</i></b>
<b><i>2.2</i></b>	<b><i>Configuration Software</i></b>	<b><i>3</i></b>
<b><i>2.3</i></b>	<b><i>Configuring HDMI Transceivers</i></b>	<b><i>3</i></b>
<b><i>2.4</i></b>	<b><i>Configuring Video Timing Controller</i></b>	<b><i>4</i></b>
<b><i>2.5</i></b>	<b><i>Configuring with Script</i></b>	<b><i>4</i></b>
<b><i>3</i></b>	<b><i>Audio Codec</i></b>	<b><i>5</i></b>

## ***List of Figures***

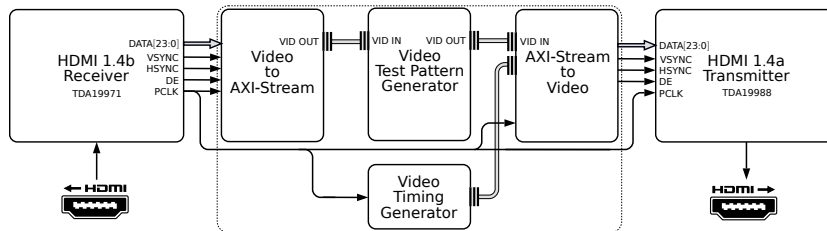
<b><i>2.1</i></b>	<b><i>HDMI Pass-Through Video Programmable Logic Configuration</i></b>	<b><i>2</i></b>
-------------------	--	-----------------

## 2 HDMI Transmit/Receive

A simple HDMI video stream is configured in the programmable logic with a common pixel clock between the HDMI receiver and HDMI transmitter. The source of this pixel clock is the HDMI receiver video bus clock. This clock is driven by logic within the HDMI receiver and the frequency is dependent on the input (TMDS) bus format. This restricts the HDMI transmitter to video formats that can be driven by the input clock frequency. For further simplification, the HDMI transmitter format configuration should be set identical to the received video format on the HDMI receiver.

A video timing generator (VTC) is responsible for generating the appropriate timing signals (blanking and syncing) using the incoming pixel clock and a specified format. The VTC has a set of registers that are used to set the timing specifications and control the timing generator behavior.

Between the HDMI receiver video stream and the output stream sent to the HDMI transmitter is a video test pattern generator (TPG). This is configurable to allow the incoming video stream to be passed through to the output stream or use a test pattern as the output video.



**Figure 2.1:** HDMI Pass-Through Video Programmable Logic Configuration

### 2.1 Enable HDMI Transceivers

The HDMI transceivers can be enabled and disabled using the connected active high GPIO pins. Control of these pins is possible through the Linux sysfs which must be accessed with root level permissions. To enable both HDMI transceivers by setting their respective enable pins to high, the following process can be used.

```
root@pismasher:~# echo 958 > /sys/class/gpio/export
root@pismasher:~# echo 959 > /sys/class/gpio/export
root@pismasher:~# echo out > /sys/class/gpio/gpio958/direction
root@pismasher:~# echo out > /sys/class/gpio/gpio959/direction
root@pismasher:~# echo 1 > /sys/class/gpio/gpio958/value
root@pismasher:~# echo 1 > /sys/class/gpio/gpio959/value
```

## 2.2 Configuration Software

The software required to configure the video blocks can be found at <http://github.com/krtkl/piSmasher-software>. This contains software for configuring the HDMI transceivers, VTC and TPG as well as some additional test utilities that can be used for further development. The software can be built all at once and on a snickerdoodle or on a cross-compilation host. A set of commands for cloning and building the software utilities is found below.

```
admin@pismasher:~$ git clone https://github.com/krtkl/piSmasher-software
```

**Code Listing 1:** Clone piSmasher Software Repository

make can be used to build the projects. A top level makefile is used to call the build process for the device driver dependencies as well as the individual project makefiles. The resulting executable applications are placed in the build directory.

```
admin@pismasher:~$ cd piSmasher-software/projects
admin@pismasher:~/piSmasher-software/projects$ make clean && make
```

## 2.3 Configuring HDMI Transceivers

The HDMI configuration utility (hdmi-config) exists to configure both HDMI transceivers.

```
hdmi-config [OPTIONS]
Options:
  -m MODE    - Configure transmitter format mode

Modes:
  1280x720
  1366x768
  1920x1080
  1920x1200
```

To run the HDMI configuration utility and initialize the HDMI transceivers:

```
admin@pismasher:~/piSmasher-software/projects$ sudo build/hdmi-config -m 1920x1080
```

Configure the test pattern generator to use the video input stream from the HDMI receiver to output to the HDMI transmitter. The -w and -h options should be followed by the active frame width and height values, respectively. These dimensions are in pixels and do not include the blanking

area.

```
admin@pismasher:~/piSmasher-software/projects$ build/vid-tpg-config -w 1920 -h 1080 -b 0 13
```

## 2.4 Configuring Video Timing Controller

```
Usage: uio-rtc [OPTIONS] UIONUM

Options:
  -g          - Configure generator (default)
  -m MODE     - Configure generator for format MODE
  -d          - Configure detector

Modes:
  1280x720
  1366x768
  1920x1080
  1920x1200
```

Configure the video timing controller for the desired video format. This format should match the selected format in the HDMI configuration when running `hdmi-config`

```
admin@pismasher:~/piSmasher-software/projects$ build/uio-rtc -m 1920x1080 12
```

## 2.5 Configuring with Script

Within the projects directory is a script which encapsulates the necessary configuration commands to enable the HDMI transceivers and set the format to 1080p@60Hz. This script can be executed with root level permissions (required) to configure the video pipeline with a single command. The contents of the script are shown below.

```
#!/bin/bash

echo 958 > /sys/class/gpio/export
echo 959 > /sys/class/gpio/export

echo out > /sys/class/gpio/gpio958/direction
echo out > /sys/class/gpio/gpio959/direction

echo 1 > /sys/class/gpio/gpio958/value
echo 1 > /sys/class/gpio/gpio959/value

sleep 2

build/hdmi-config -m 1920x1080
build/uio-vtc -m 1920x1080 12
build/vid-tpg-config -w 1920 -h 1080 -b 0 13
```

To execute the script with root level permissions, invoke the script from the command line as shown below.

```
admin@pismasher:~/piSmasher-software/projects$ sudo ./hdmi_en.sh
```

### ***3 Audio Codec***

Configure the input and output paths for the audio codec.

```
admin@pismasher:~/piSmasher-software/projects$ build/audio-config -i linein -o lineout
```