

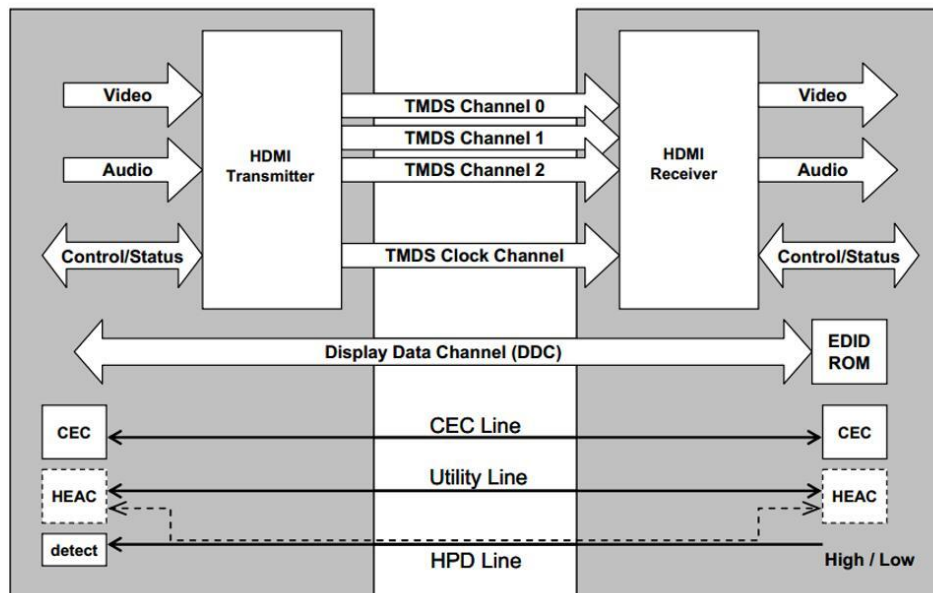
8. HDMI 输出彩条实验例程

8.1 实验目的

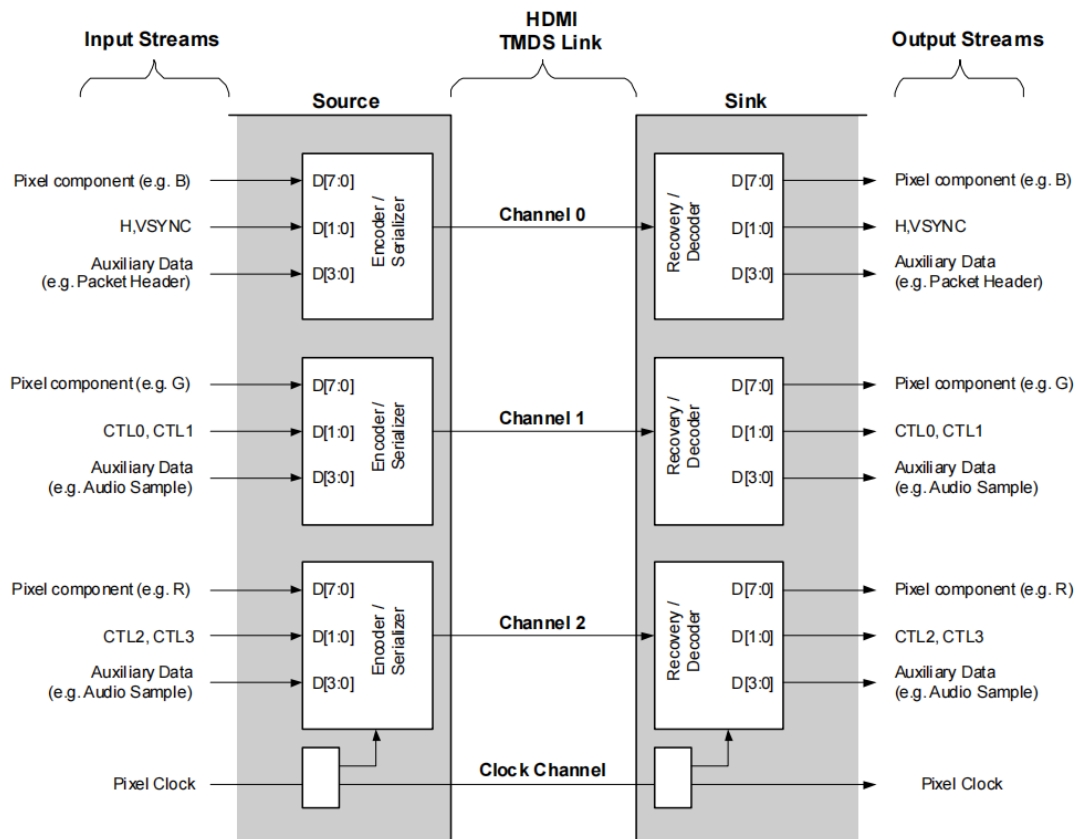
FPGA 输出 HDMI 图像信号（分辨率为 1280*720@60），实现彩条显示。22K 开发板需要通过代码加原语的方式模拟产生 HDMI 信号并输出。

8.2 实验原理

8.2.1 TMDS 传输简介



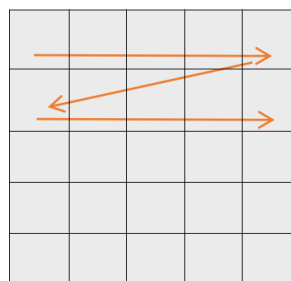
HDMI 重点就在 TMDS 通道，TMDS 传输 HDMI 接口的音视频数据，TMDS 通道各个通路的数据结构组成如下：



HMDI 的 Source 端输入数据主要有像素点数据、行场同步、控制信号、辅助信号（音频）。显示器上一整幅画面由多个像素点组成，实际的显示过程是很快速的把显示器的各个像素点点亮。

8.2.2显示时序标准

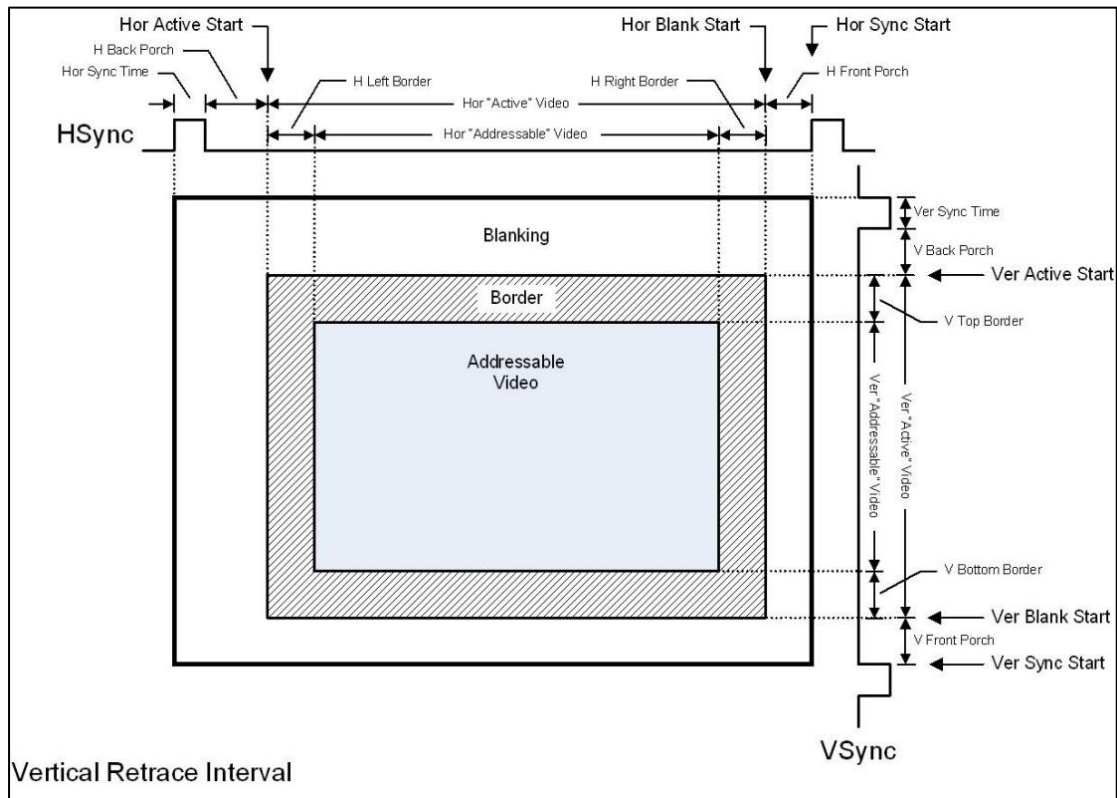
若下图表示一个 5*5 的画面，图中每个格子表示一个像素点，显示图像时像素点快速点亮的过程按表格中编号的顺序逐个点亮，从左到右，从上到下，按图中箭头方向的“Z”字形顺序。



以上图为例，每行 5 个像素点，每完成一行信号的传输，会转到下一行信号传输，直到完成第五行数据的传输，就完成了画面的数据传输了，一个画面也称为一场或一帧，显示每秒中刷新的帧数称为帧率。比如 1920*1080P 像素，就是 1 行有效像素点 1920，一场有效行为

1080 行。

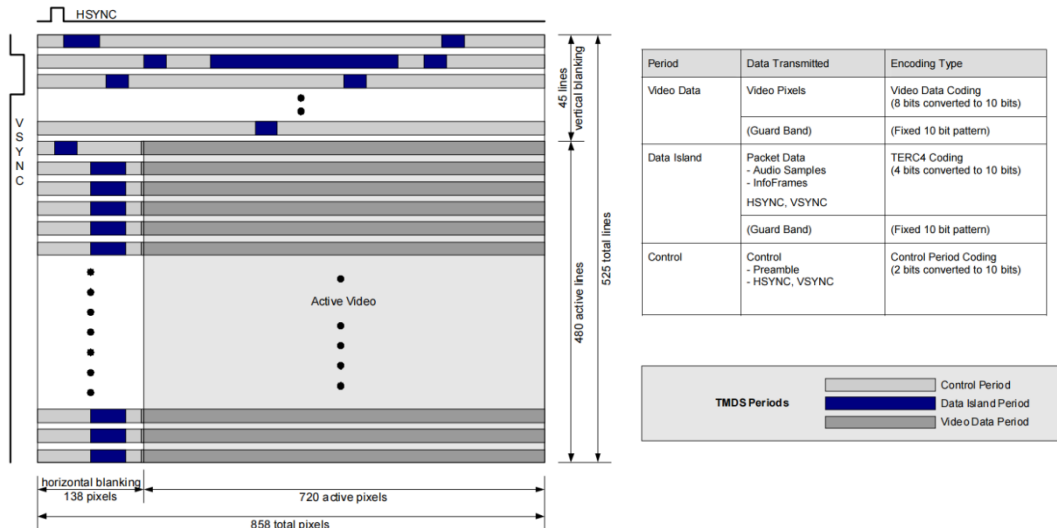
每个像素点的像素值数据,对应每个像素点的颜色。常见的像素值表示格式比如:RGB888, RGB 分别代表: 红 R, 绿 G, 蓝 B, 888 是指 R、G、B 分别有 8bit, 也就是 R、G、B 每一色光有 $2^8=256$ 级阶调, 通过 RGB 三色光的不同组合, 一个像素上最多可显示 24 位的 $256*256*256=16,777,216$ 色。



像素数据源源不断输送进来,行、场的切换通过行场同步信号来控制,即 hsync (行同步) 和 vsync (场同步信号)。

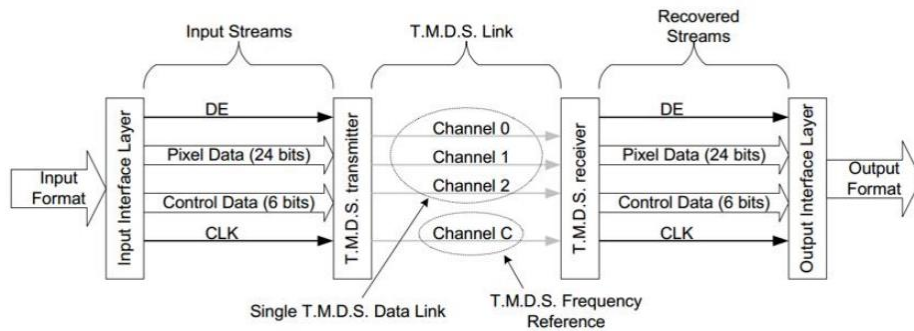
上图中 Addressable 部分内容是在显示器中可看到的区域, 像素点是否有效通过 DE 信号标识; Border 可理解为显示黑边或者显示边框, 通常 Border 显示的像素值是 0 (黑色)。行、场切换过程都是在用户感受不到的区域进行的, 这个区域就是 Blanking 部分, 称为消隐区间。同步信号上升沿表示新的一行/一场开始, Hsync 对应行, Vsync 对应场。

下图是 HDMI 显示的 720X480 分辨率的 TMDS 编码帧结构。

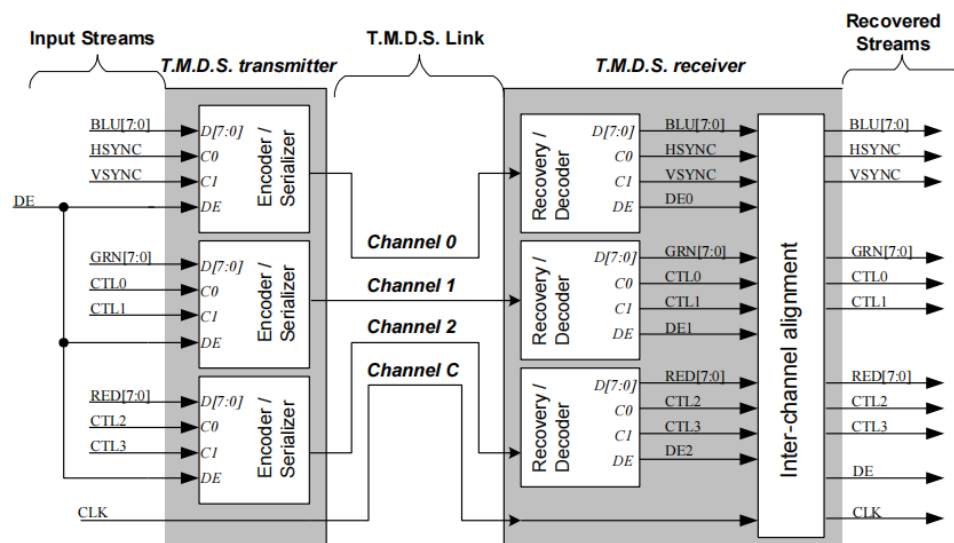


本实验只考虑 video 信号的传输，实现图像的 T.M.D.S. 编码时只需要 Video data Period 和 control period。

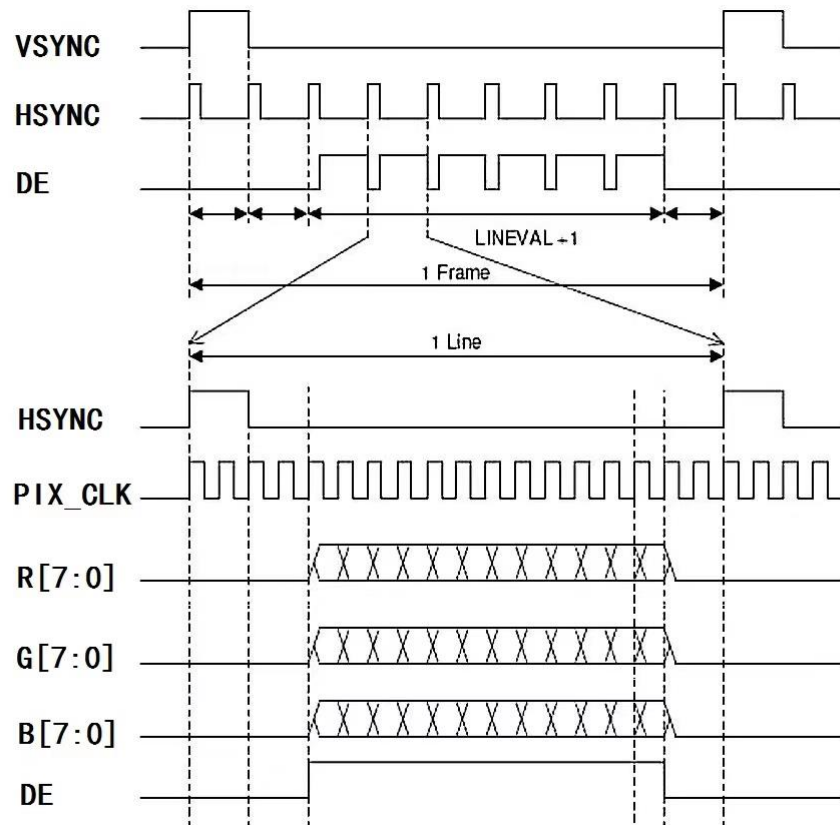
完整的信号链路如下：



每个 channel 链路如下：



8.2.3 Color_bar 产生



HDMI 显示的数据源采用 verilog 编写的 color_bar 模块产生。根据像素点所在位置，即列数和行数确定像素值，实现彩条图案。另外 color_bar 需实现上图的时序要求。

本实验采用 1280*720@60 的视频规格，详细时序参数如下：

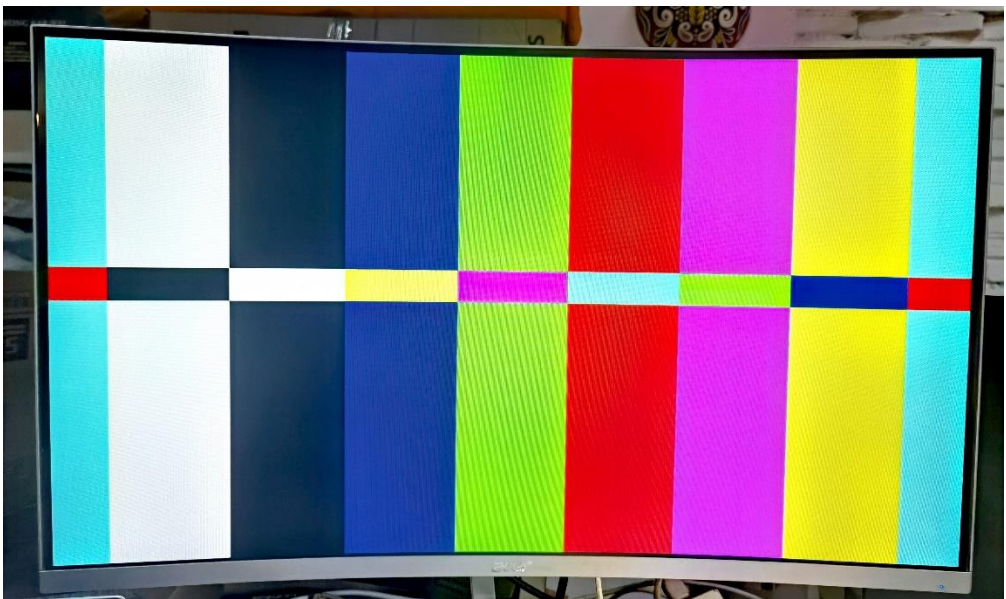
VESA MONITOR TIMING STANDARD

Adopted: 11/17/08
Resolution: 1280 x 720 at 60 Hz (non-interlaced)
EDID ID: DMT ID: 55h; STD 2 Byte Code: 81h, C0h; CVT 3 Byte Code: n/a
Method: *** NOT CVT COMPLIANT ***

Per CEA-861 --- 720p (Code 4) Timing Definitions

Detailed Timing Parameters					
Timing Name	= 1280 x 720 @ 60Hz;				
Hor Pixels	= 1280;	// Pixels			
Ver Pixels	= 720;	// Lines			
Hor Frequency	= 45.000;	// KHz	= 22.2 usec	/ line	
Ver Frequency	= 60.000;	// Hz	= 16.7 msec	/ frame	
Pixel Clock	= 74.250;	// MHz	= 13.5 nsec	± 0.5%	
Character Width	= 1;	// Pixels	= 13.5 nsec		
Scan Type	= NONINTERLACED;		// H Phase	= 3.3 %	
Hor Sync Polarity	= POSITIVE;	// HBlank	= 22.4% of HTotal		
Ver Sync Polarity	= POSITIVE;	// VBlank	= 4.0% of VTotal		
Hor Total Time	= 22.222;	// (usec)	= 1650 chars	= 1650 Pixels	
Hor Addr Time	= 17.239;	// (usec)	= 1280 chars	= 1280 Pixels	
Hor Blank Start	= 17.239;	// (usec)	= 1280 chars	= 1280 Pixels	
Hor Blank Time	= 4.983;	// (usec)	= 370 chars	= 370 Pixels	
Hor Sync Start	= 18.721;	// (usec)	= 1390 chars	= 1390 Pixels	
// H Right Border	= 0.000;	// (usec)	= 0 chars	= 0 Pixels	
// H Front Porch	= 1.481;	// (usec)	= 110 chars	= 110 Pixels	
Hor Sync Time	= 0.539;	// (usec)	= 40 chars	= 40 Pixels	
// H Back Porch	= 2.963;	// (usec)	= 220 chars	= 220 Pixels	
// H Left Border	= 0.000;	// (usec)	= 0 chars	= 0 Pixels	
Ver Total Time	= 16.667;	// (msec)	= 750 lines	HT - (1.06xHA)	
Ver Addr Time	= 16.000;	// (msec)	= 720 lines	= 3.95	
Ver Blank Start	= 16.000;	// (msec)	= 720 lines		
Ver Blank Time	= 0.667;	// (msec)	= 30 lines		
Ver Sync Start	= 16.111;	// (msec)	= 725 lines		
// V Bottom Border	= 0.000;	// (msec)	= 0 lines		
// V Front Porch	= 0.111;	// (msec)	= 5 lines		
Ver Sync Time	= 0.111;	// (msec)	= 5 lines		
// V Back Porch	= 0.444;	// (msec)	= 20 lines		
// V Top Border	= 0.000;	// (msec)	= 0 lines		

彩条分成 8 列 1 横条，竖条图案向右规律性移动，横条图案上下摆动。



8.2.4TMS 编码

TMS 的编码会把 D[7:0]或 C[1:0]编码成 10bit 的数据, 编码的目的是为了是 TMS 在传输过程中维持直流平衡, 也就是说传输的 0 和 1 的数量保持基本一致。下图是 TMS 编码的流程图。

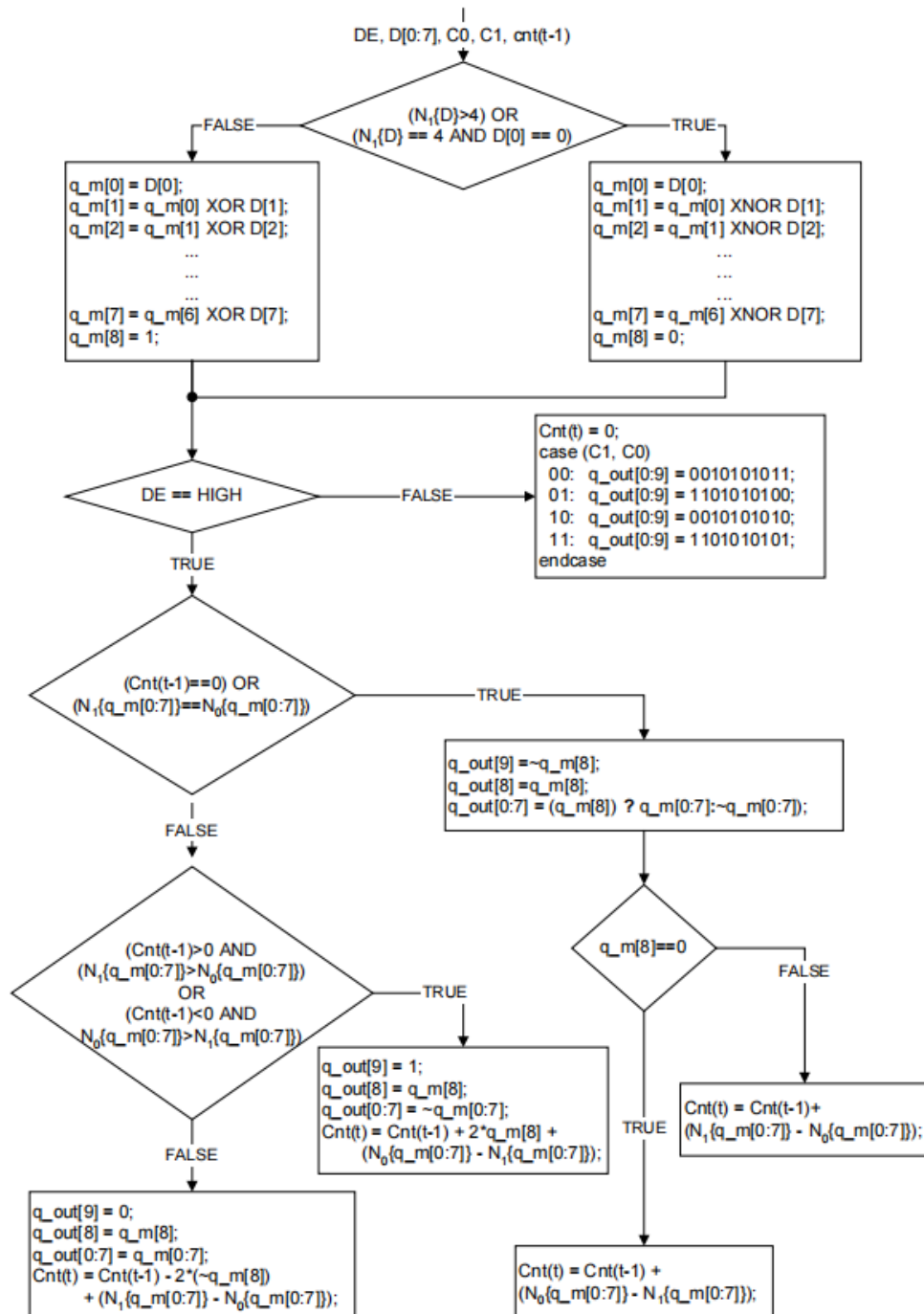
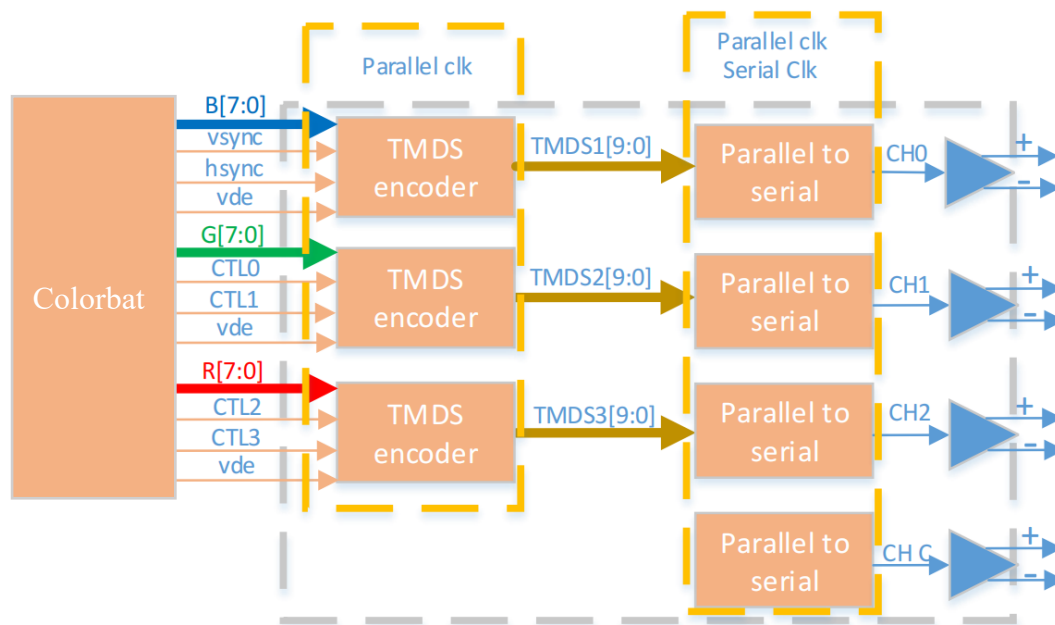


Figure 3-5. T.M.D.S. Encode Algorithm



8.3 实验源码

实验源码详见参考例程。

8.4 实验现象

HDMI_OUT 接口接上支持对应分辨率（1280*720P@60）的屏后，显示彩条图案,且彩条图案规律性移动（竖条图案向右规律性移动，横条图案上下摆动）。

