

# **HDMI 显示模块 AN9134**

## **用户手册**

**Rev. 1.00**



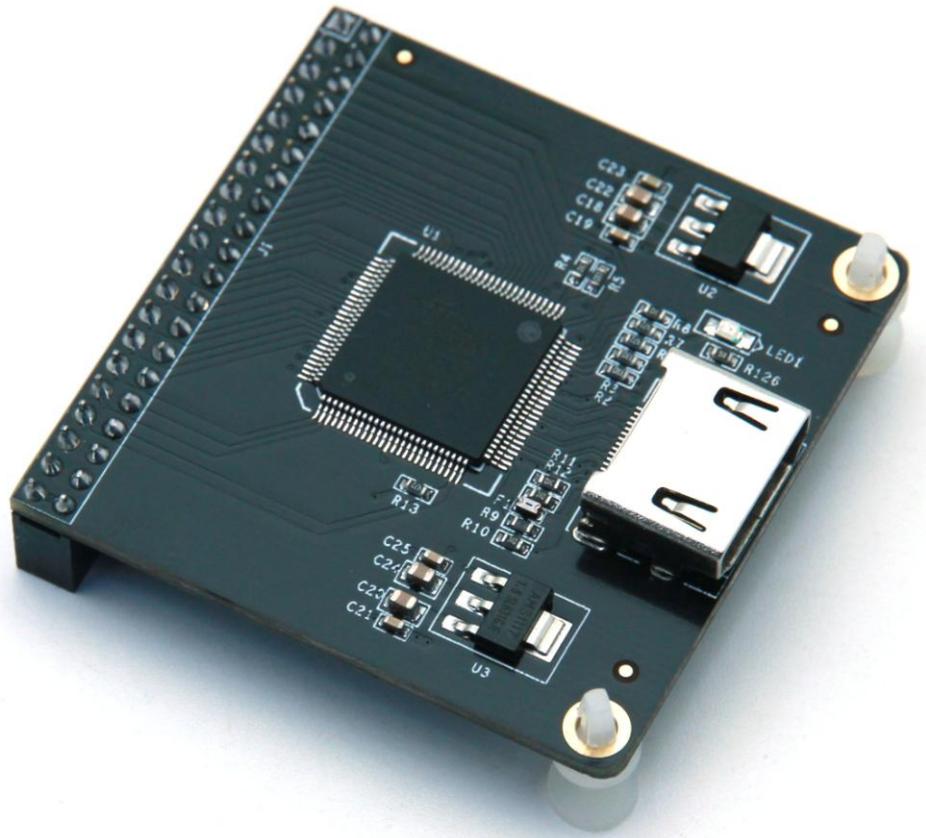
## 版本记录

版本	时间	作者	描述
Rev1.00	2017-11-1		First Release

## 第一部分 HDMI 显示模块说明介绍

黑金 HDMI 显示输出模块 AN9134 采用 Sillion Image 公司的 SIL9134 HDMI (DVI) 编码芯片，最高支持 1080P@60Hz 输出，支持 3D 输出。模块留有一个 40 针的排母用于连接 FPGA 开发板，1 个 HDMI 连接器用于 HDMI 视频信号的输出到 HDMI 显示器。

AN9134 模块实物照片如下：



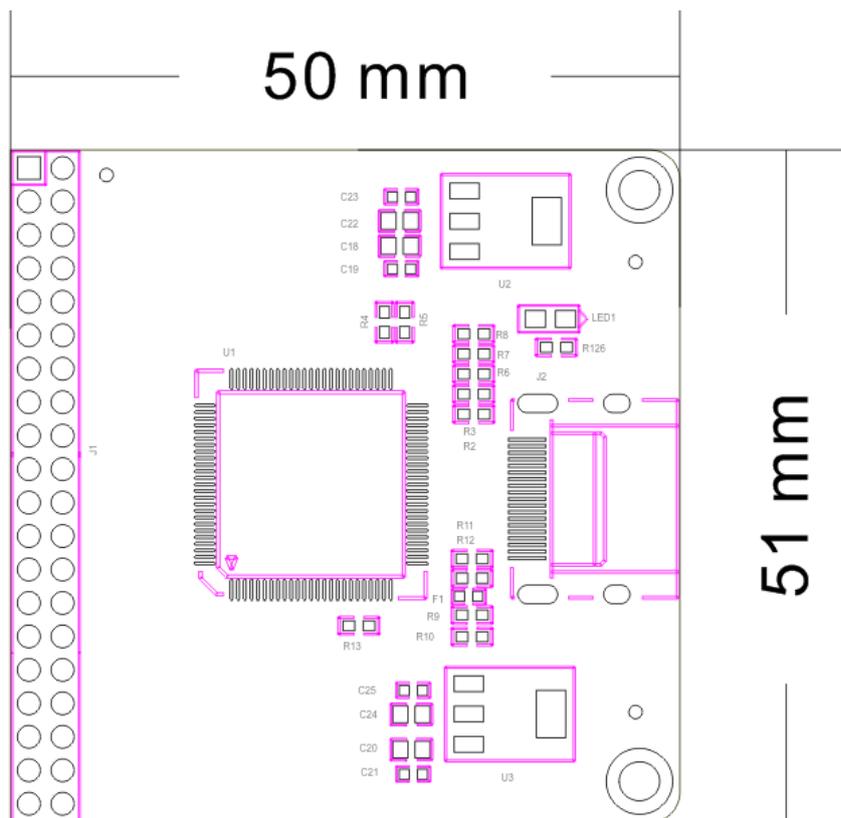
AN9134 模块实物图

## 1.1 AN9134 模块的参数说明

以下为 AN9134 HDMI 显示模块的详细参数:

- HDMI 编码芯片 : SiI9134 ;
- HDMI 显示道数 : 1 通道 ;
- HDMI 接口显示标准 : HDMI 1.4 ;
- 数据宽度 : 24 位 RGB/YCbCr 4:4:4 ;
- 显示最高分辨率和刷新率 : 1080P 60 帧 ;
- 显示参数配置 : I2C 接口配置 ;
- 模块接口 : 40 针 2.54mm 间距排座 , 方向向下 ;
- 工作温度 :  $-0^{\circ}\sim 70^{\circ}$  ;
- 输出接口 : 1 路 HDMI 标准输出接口 ;

## 1.2 AN9134 模块的参数说明

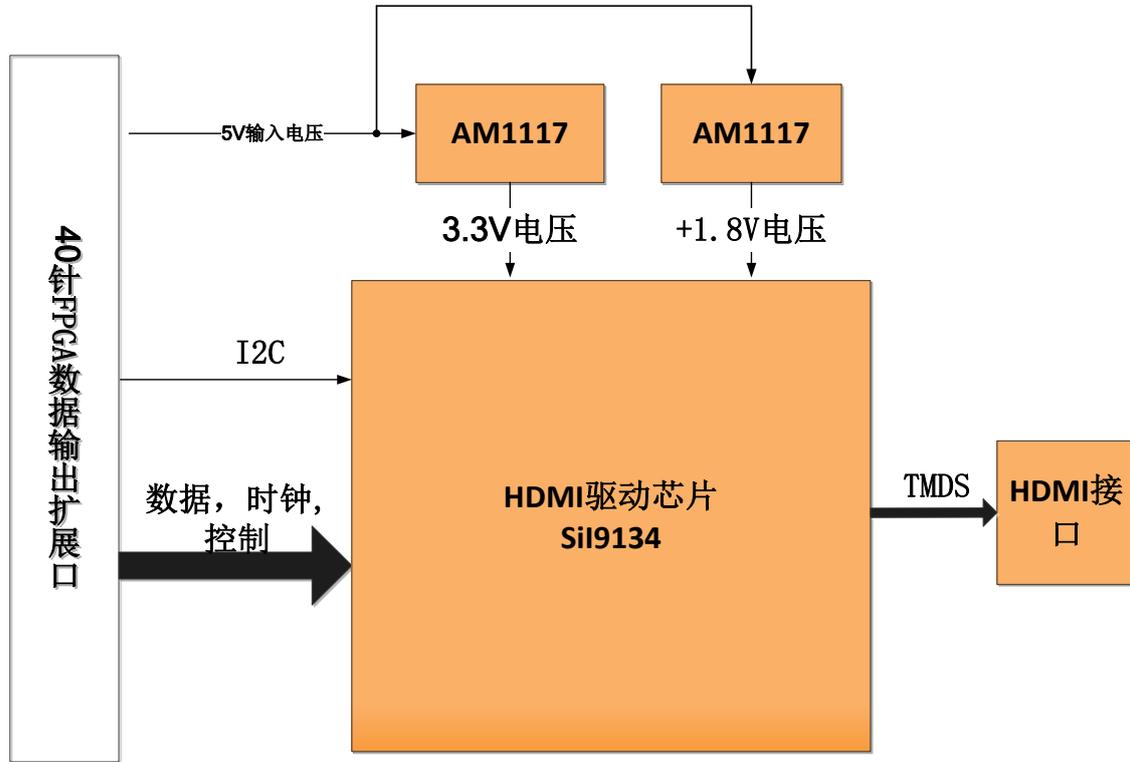


AN9134 HDMI 显示模块尺寸图

## 第二部分 模块功能说明

### 2.1 AN9134 模块原理框图

AN9134 模块的原理设计框图如下：



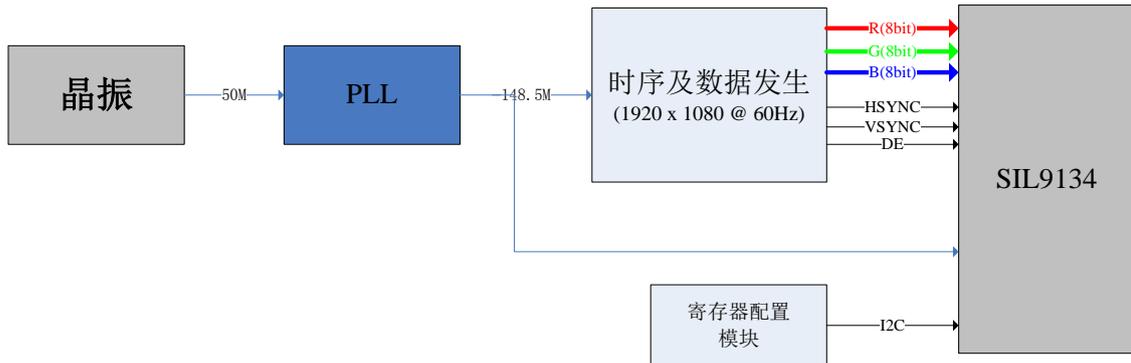
### 2.2 模块 40 针排母的引脚分配：

引脚号	引脚名称	备注
1	GND	地
2	+5V	5V 电源输入
3	9134_SCL	9134 I2C 时钟
4	9134_SDA	9134 I2C 数据
5	9134_D22	9134 数据输入第 22 位
6	9134_D23	9134 数据输入第 23 位
7	9134_D20	9134 数据输入第 20 位
8	9134_D21	9134 数据输入第 21 位
9	9134_D18	9134 数据输入第 18 位
10	9134_D19	9134 数据输入第 19 位
11	9134_D16	9134 数据输入第 16 位
12	9134_D17	9134 数据输入第 17 位

13	9134_D14	9134 数据输入第 14 位
14	9134_D15	9134 数据输入第 15 位
15	9134_D12	9134 数据输入第 12 位
16	9134_D13	9134 数据输入第 13 位
17	9134_D10	9134 数据输入第 10 位
18	9134_D11	9134 数据输入第 11 位
19	9134_D8	9134 数据输入第 8 位
20	9134_D9	9134 数据输入第 9 位
21	9134_D6	9134 数据输入第 6 位
22	9134_D7	9134 数据输入第 7 位
23	9134_CLK	9134 时钟信号
24	9134_D5	9134 数据输入第 5 位
25	9134_D3	9134 数据输入第 3 位
26	9134_D4	9134 数据输入第 4 位
27	9134_D1	9134 数据输入第 1 位
28	9134_D2	9134 数据输入第 2 位
29	9134_DE	9134 数据输入有效信号
30	9134_D0	9134 数据输入第 0 位
31	9134_VS	9134 帧同步信号
32	9134_HS	9134 行同步信号
33	9134_nRESET	9134 复位信号
34	9134_INT	9134 中断输出信号
35	-	NC
36	-	NC
37	GND	地
38	GND	地
39	+3.3V	NC
40	+3.3V	NC

## 第三部分 HDMI 显示程序说明

我们提供了黑金 FPGA 开发板的 HDMI 驱动例程，在这个驱动例程中 FPGA 通过内部产生 HDMI 的时序和测试数据来驱动 SiI9134 的输出显示。通过修改 HDMI 的输出时钟和时序可以产生不同分辨率的测试图像，FPGA 的测试图像可以通过按键 KEY1 来切换。测试程序的功能示意图如下：



下面对 FPGA 程序中用到的各个模块做一下简单的功能介绍：

### 1. mv\_pattern.v

此模块用来产生不同的测试图像和 HDMI 的行列同步信号。模块中产生了 7 种测试图，分别是 8 色横彩条，8 色竖彩条，滚动渐变色，黑白方块，全蓝色，彩色渐变色，边框。HDMI 的行列同步信号是由 TOP 层的定义参数来产生的 HDMI 所需要的时序。

### 2. ax\_debounce.v

此模块用来检测按键的按下动作，也就是检测按键的下降沿。程序中也加入了按键去抖的功能。

### 3. i2c\_config.v

此模块通过调用 I2C 通信模块对 SiI9134 芯片进行寄存器配置，配置的寄存

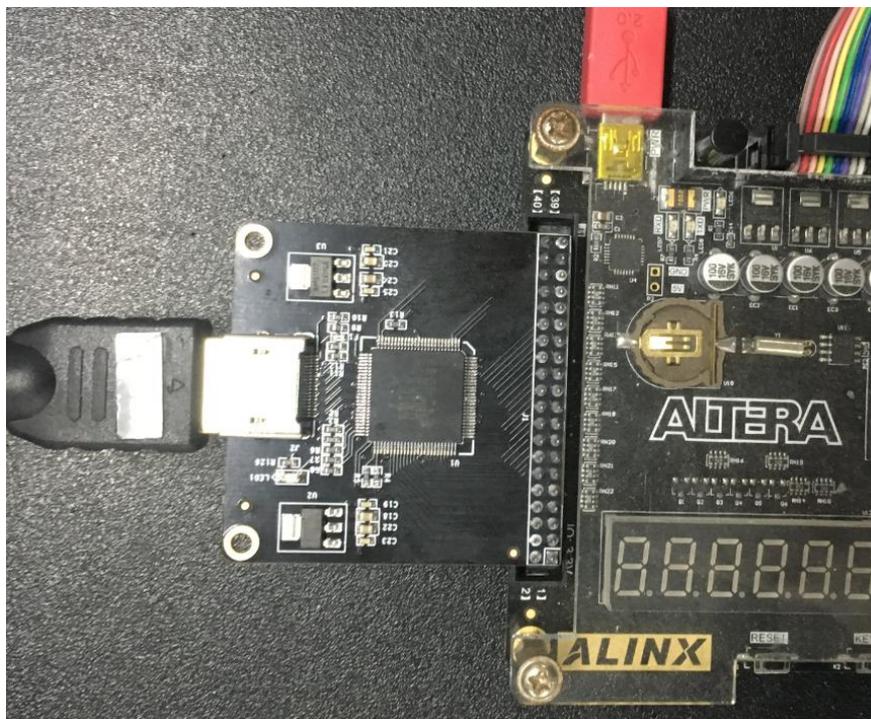
器地址和值定义在 lut\_9134.v 文件里。具体的寄存器含义大家参考 Si9134 的芯片手册。

#### 4. reset\_power\_on.v

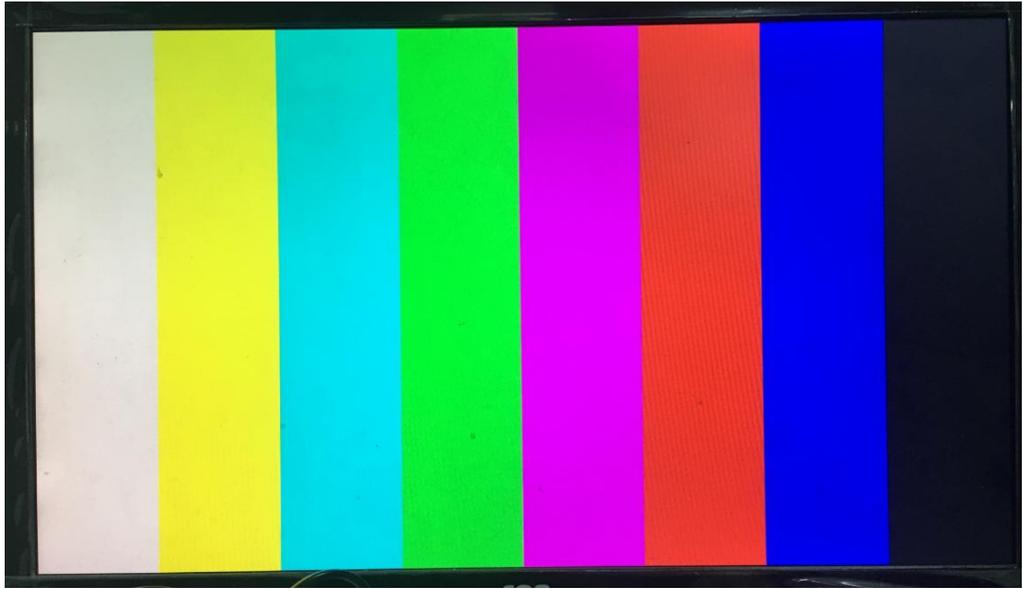
上电复位模块，上电后产生一个软件复位给其它模块初始化。

## 第四部分 硬件连接和测试

AN9134 模块和 FPGA 开发板的硬件连接很简单，只要把模块的 40 针的母座插到 FPGA 开发板的扩展口上，连接器的管脚 1 对齐就好了，然后用 HDMI 线连接到显示器。以下为黑金 AX301 开发板的 J1 扩展口和 AN9134 HDMI 模块的硬件连接图（如果需要连接 J2 扩展口，管脚需要重新分配）：



开发板上电，然后下载程序就可以在 HDMI 显示器上显示彩条图像。



显示彩条

我们可以通过开发板上的按键 KEY1 来切换显示的图像：



显示滚动的灰度条



显示彩色的灰度条

如果用户需要修改输出不同的分辨率图像，只需要修改如下两个地方：

1. an9134\_test 顶层文件里下面的这句话定义成您想要输出的分辨率。

```
41     output hdmi_de,  
42     output[23:0] hdmi_d  
43 )  
44 `define VIDEO_1920_1080  
45 //video timing parameter definition  
46 `ifdef VIDEO_1280_720  
47 parameter H_ACTIVE = 16'd1280;           //horizon  
48 parameter H_FP = 16'd110;               //horizon  
49 parameter H_SYNC = 16'd40;              //horizon
```

2. 修改 PLL 的时钟输出频率，默认是输出 1080P 的时钟频率（148.5M），您需要修改下图中的值就可以了。不同的分辨率输出的时钟频率在程序里都有备注。

MegaWizard Plug-In Manager [page 6 of 12]

**ALTPLL** [About] [Documentation]

1 Parameter Settings | 2 PLL Reconfiguration | 3 Output Clocks | 4 EDA | 5 Summary

clk c0 > **clk c1** > clk c2 > clk c3 > clk c4

video\_pll

inclck0 → [inclck0 frequency: 50.000 MHz  
Operation Mode: Normal  
Clk Ratio Ph.(dg) DC.(%)  
c0 297/100 0.00 50.00] → c0

Cyclone IV E

### c0 - Core/External Output Clock

Able to implement the requested PLL

Use this clock

Clock Tap Settings

Enter output clock frequency: 148.5000000 MHz | Actual Settings: 148.437500

Enter output clock parameters:

Clock multiplication factor: 1 | 95

Clock division factor: 1 | 32

Clock phase shift: 0.00 deg | 0.00

Clock duty cycle (%): 50.00 | 50.00

Description	Val.
Primary clock VCO frequency (MHz)	5...
Modulus for M counter	95

Note: The displayed internal settings of the PLL is recommended for use by advanced users only

Per Clock Feasibility Indicators: c0 c1 c2 c3 c4

[Cancel] [**< Back**] [Next >] [Finish]