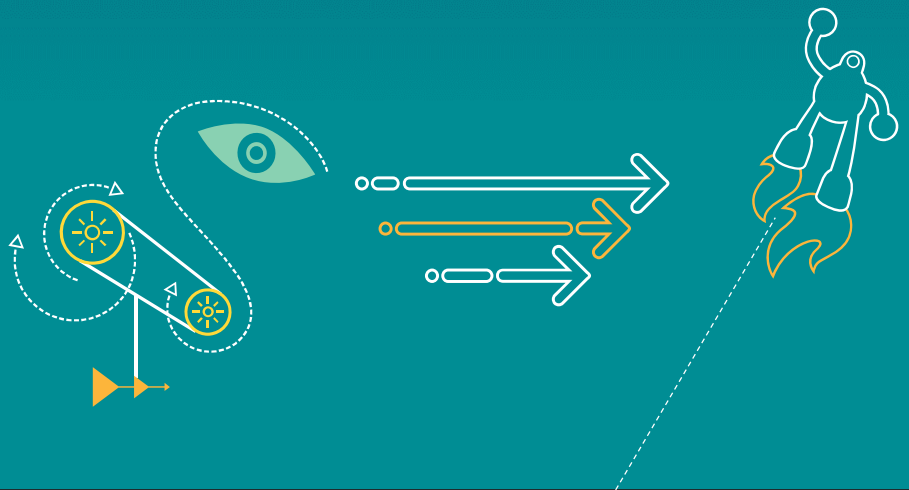
高通多媒体技术期刊　20160309

Qualcomm Technologies, Inc.



Confidential and Proprietary – Qualcomm Technologies, Inc.

Revision History

Revision	Date	Description
A	Mar. 2016	Initial release

Note: There is no Rev. I, O, Q, S, X, or Z per Mil. standards.

内容

- Display
  - 如何计算 HDMI Timing
  - 如何添加新的HDMI resolution
  - 在8937/8953/8917平台上，如何计算 DSI Timing

Display

如何计算HDMI Timing

- 对于OEM/ODM，有时候需要计算HDMI timing，举例如下:
  - Active Width = 1280
  - Active Height = 800
  - Blank_Horizontal = HFP + HPW + HBP = 48 + 32 + 80 = 160
  - Blank_Vertical = VFP + VPW + VBP = 3 + 6 + 14 = 23
  - Fps = 60
- 从而得到HDMI Timing为：
- 水平方向的frequency
  - freq_h= 1 / [(1/60Hz) / (Height + Blank_Vertical)] = 49383
- 垂直方向的frequency
  - freq_v = 60Hz x 1000 = 60000
- Pixel frequency
  - pixel_freq = (Active Width + Blank_Horizontal) * (Active Height + Blank_Vertical) * 60Hz = 1440 x 823 x 60 / 1000 = 71107
- Refresh rate
  - 60Hz x 1000 = 60000

如何添加新的HDMI resolution

- 首先介绍，如何通过命令方法来添加新的HDMI resolution，如下步骤：

- 1. 需要找到 HDMI driver 注册的 FB 节点，即fb1
  - adb shell cat /sys/class/graphics/fb1/msm_fb_type
  - 上面命令返回为 dtv panel
- 2. 确保 /sys/class/graphics/fb1/add_res存在，意味着动态添加新的HDMI resolution是被支持的。
- 3. 连接 HDMI cable 到Sink 端
- 4. 使能 HDMI EDID logs
  - adb wait-for-device root
  - adb wait-for-device shell “echo 8 > /proc/sys/kernel/printk”
  - adb shell "echo -n \"file mdss_hdmi_edid.c +tp\" > /sys/kernel/debug/dynamic_debug/control”
- 5. 查看kernel log 信息
  - adb logcat -b kernel
- 6. 查看EDID modes 信息
  - adb shell cat /sys/class/graphics/fb1/edid_modes
  - 显示结果为1,5,4,3,2,1,6,7,16,135,136

如何添加新的HDMI resolution –cont1

- 7. 使能resolution 信息
  - adb shell "echo 1 > /sys/class/graphics/fb1/res_info"
- 8. 通过cat 显示resolution 信息
  - adb shell cat /sys/class/graphics/fb1/res_info
  - 同样的信息可以在kernel log中看到，如下：

```
01-01 00:16:39.615 0 0 D : 1,640,16,96,48,1,480,10,2,33,1,25200,60000,0,1,1,1
01-01 00:16:39.620 0 0 D : 5,1920,88,44,148,0,540,2,5,5,0,74250,60000,0,1,3,1
01-01 00:16:39.626 0 0 D : 4,1280,110,40,220,0,720,5,5,20,0,74250,60000,0,1,3,1
01-01 00:16:39.632 0 0 D : 3,720,16,62,60,1,480,9,6,30,1,27027,60000,0,1,3,1
01-01 00:16:39.637 0 0 D : 2,720,16,62,60,1,480,9,6,30,1,27027,60000,0,1,1,1
01-01 00:16:39.643 0 0 D : 1,640,16,96,48,1,480,10,2,33,1,25200,60000,0,1,1,1
01-01 00:16:39.649 0 0 D : 6,1440,38,124,114,1,240,4,3,15,1,27000,60000,1,1,1,1
01-01 00:16:39.655 0 0 D : 7,1440,38,124,114,1,240,4,3,15,1,27000,60000,1,1,3,1
01-01 00:16:39.661 0 0 D : 16,1920,88,44,148,0,1080,4,5,36,0,148500,60000,0,1,3,1
01-01 00:16:39.667 0 0 D : 135,800,40,128,88,0,600,1,4,23,0,40000,60000,0,1,1,1
```

- 上面对应的结构体变量定义为 struct msm_hdmi_mode_timing_info
  - /include/uapi/video/msm_hdmi_modes.h

如何添加新的HDMI resolution –cont2

- 9. 获取新添加的HDMI resolution的信息，比如：
  - PCLK(MHZ) HPW(PCLK) HBRP(PCLK) Hactive(PCLK) HFPR(PCLK) VSW(line) VBPR(line) Vactive(line) VFPR(line)
  - 对应值为 256 12 12 2880 24 2 7 1440 8
- 10.添加新的resolution：
  - echo active_h hfp hpw hbp active_low_h(0/1) active_v vfp vpw vbp active_low_v(0/1) pclk(KHz) fps(x1000) 0 1 aspect_ratio(1->4/3, 2:5/4, 3:16/9) > add_res
- 11. 对应上述的HDMI resolution，如下操作：
  - adb shell
  - cd /sys/class/graphics/fb1/
  - echo 2880 24 12 12 0 1440 8 2 7 0 256000 60000 0 1 3 > add_res
- 12. Disconnect HDMI cable，然后重新连接
- 13. 运行下面命令查看新添加的EDID
  - cat edid_modes
  - 结果为：1,5,4,3,2,1,6,7,16,135,136,146
- 14. 对比步骤6，可以看到新添加的resolution为 146.

如何添加新的HDMI resolution –cont3

- 15. 再执行 “echo 1 > res_info”，然后 “cat res_info”，查看kernel log，会出现下面信息：
  - 01-01 00:38:17.045 0 0 D : 146,2880,24,12,12,0,1440,8,2,7,0,256000,60000,0,1,3,1
- 16. 如果新添加的resolution没有设置，需要执行下面的操作：
  - setprop hw.hdmi.resolution 146
  - stop
  - start
  - Disconnect HDMI cable
  - connect HDMI cable
- 可以看到新的resolution起作用，且可以通过下面command来查看当前HDMI resolution：
  - cat video_mode. 显示结果为146.

如何添加新的HDMI resolution –cont4

- 前面我们介绍了如何通过命令形式添加新的HDMI resolution，下面介绍具体代码的修改：
- 1： 在include/uapi/video/msm_hdmi_modes.h 文件中，修改如下：
- a)

```
@@ -137,7 +137,8 @@ struct msm_hdmi_mode_timing_info {

 /* WQXGA */
 #define HDMI_VFRMT_2560x1600p60_16_9   WQXGA_OFF(1)
-#define HDMI_WQXGAFRMT_END     HDMI_VFRMT_2560x1600p60_16_9
+#define HDMI_VFRMT_2880x1440p60_16_9   WQXGA_OFF(2)
+#define HDMI_WQXGAFRMT_END     HDMI_VFRMT_2880x1440p60_16_9

 #define WXGA_OFF(x)            (HDMI_WQXGAFRMT_END + x)

@@ -240,6 +241,9 @@ struct msm_hdmi_mode_timing_info {
 #define HDMI_VFRMT_2560x1600p60_16_9_TIMING                \
    {HDMI_VFRMT_2560x1600p60_16_9, 2560, 48, 32, 80, false,     \
     1600, 3, 6, 37, false, 268500, 60000, false, HDMI_RES_AR_16_9}
+#define HDMI_VFRMT_2880x1440p60_16_9_TIMING                \
+   {HDMI_VFRMT_2880x1440p60_16_9, 2880, 24, 12, 12, false,     \
+    1440, 8, 2, 7, false, 256000, 60000, false, HDMI_RES_AR_16_9}
 #define HDMI_VFRMT_3840x2160p30_16_9_TIMING                \
    {HDMI_VFRMT_3840x2160p30_16_9, 3840, 176, 88, 296, false,   \
     2160, 8, 10, 72, false, 297000, 30000, false, HDMI_RES_AR_16_9}
```

如何添加新的HDMI resolution –cont5

- b)

```
@@ -358,6 +362,8 @@ do {     \
         MSM_HDMI_MODES_SET_TIMING(__lut,     \
             HDMI_VFRMT_2560x1600p60_16_9);   \
         MSM_HDMI_MODES_SET_TIMING(__lut,     \
+            HDMI_VFRMT_2880x1440p60_16_9);   \
+        MSM_HDMI_MODES_SET_TIMING(__lut,     \
             HDMI_VFRMT_800x600p60_4_3); \
         MSM_HDMI_MODES_SET_TIMING(__lut,     \
             HDMI_VFRMT_848x480p60_16_9);     \
@@ -457,6 +463,7 @@ static inline const char *msm_hdmi_mode_2string(u
     case HDMI_VFRMT_1024x768p60_4_3:    return "1024x768 p60 4/3";
     case HDMI_VFRMT_1280x1024p60_5_4:   return "1280x1024 p60 5/4";
     case HDMI_VFRMT_2560x1600p60_16_9:  return "2560x1600 p60 16/9";
+    case HDMI_VFRMT_2880x1440p60_16_9:  return "2880x1440 p60 16/9";
     default:                    return "???";
     }
 }
```

- 其中：pixel frequency = { (2880+24+12+12) * (1440+8+2+7) * 60 } / 1000 = 256000

如何添加新的HDMI resolution –cont6

- 2: 在libhdmi/hdmi.cpp文件中，修改如下：

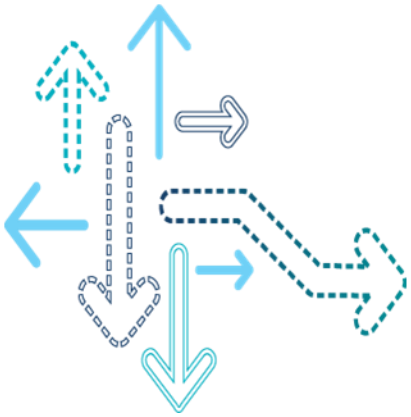
```
@@ -86,10 +86,11 @@ EDIDData gEDIDData [] = {
      EDIDData(HDMI_VFRMT_1920x1080p60_16_9, 1920, 1080, 60, 29),
      EDIDData(HDMI_VFRMT_1920x1200p60_16_10, 1920, 1200, 60, 30),
      EDIDData(HDMI_VFRMT_2560x1600p60_16_9, 2560, 1600, 60, 31),
-     EDIDData(HDMI_VFRMT_3840x2160p24_16_9, 3840, 2160, 24, 32),
-     EDIDData(HDMI_VFRMT_3840x2160p25_16_9, 3840, 2160, 25, 33),
-     EDIDData(HDMI_VFRMT_3840x2160p30_16_9, 3840, 2160, 30, 34),
-     EDIDData(HDMI_VFRMT_4096x2160p24_16_9, 4096, 2160, 24, 35),
+     EDIDData(HDMI_VFRMT_2880x1440p60_16_9, 2880, 1440, 60, 32),
+     EDIDData(HDMI_VFRMT_3840x2160p24_16_9, 3840, 2160, 24, 33),
+     EDIDData(HDMI_VFRMT_3840x2160p25_16_9, 3840, 2160, 25, 34),
+     EDIDData(HDMI_VFRMT_3840x2160p30_16_9, 3840, 2160, 30, 35),
+     EDIDData(HDMI_VFRMT_4096x2160p24_16_9, 4096, 2160, 24, 36),
  };
```

在8937/8953/8917平台上，如何计算DSI Timing

- 对于8937/8953/8917平台，请下载80-NH713-1_G_DSI_Timing_Parameters 文档。

- 在chip 选项中，可以选择8937, 8953或者8917，如下图所示：

Vert. Front Porch	1	*lines*
Escclk source (mxo = 27MHz, pxo = 24MHz, cxo = 19.2MHz)	19.2	*MHz*
MMSS_CC ESCCLK PREDIV	1	
Chip	8x37	
DSI PHY IP Catalog version (major)	1	
MDP REGISTER PROGRAMMING		
Hsync period	2060	dclks/line
Vsync period	1539	lines/frame
Dot clock overhead (blanking %)	1.01	



Questions?

https://support.cdmatech.com