U: Cb V: Cr

Y代表亮度信息，Cb，Cr为色度信号，用于描述图像的色彩以及饱和度，Cb为饱和度信息，对应为U, Cr为色度信号对应为V

I420: YYYYYYYY UU VV    =>YUV420P  
YV12: YYYYYYYY VV UU    =>YVU420P  
NV12: YYYYYYYY UVUV     =>YUV420SP  
NV21: YYYYYYYY VUVU     =>YVU420SP

YV12 ： 亮度（行×列） ＋ V（行×列/4) + U（行×列/4）

I420 ： 亮度（行×列） ＋ U（行×列/4) + V（行×列/4） YUV420p

可以看出，YV12和I420基本上是一样的，就是UV的顺序不同。

**Android OMX**中定义了 YUV420 YUV422 的打包格式(Packet/紧凑型/Interleaved/交错)，全平面格式(planer)，半平面格式(semi-planer),

基本都是U在前 V在后的

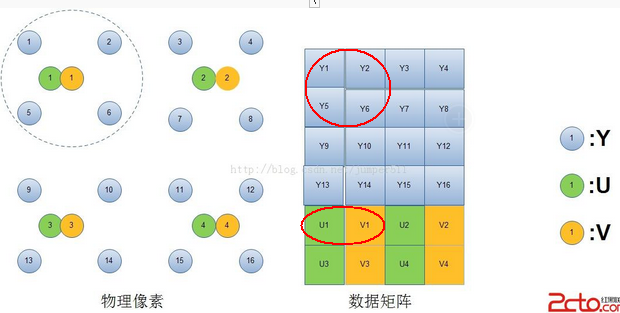
COLOR\_FormatYUV420Planar = 19; // Android采用I420/ YU12 还有一种是是 YV12 格式（属于YUV420）



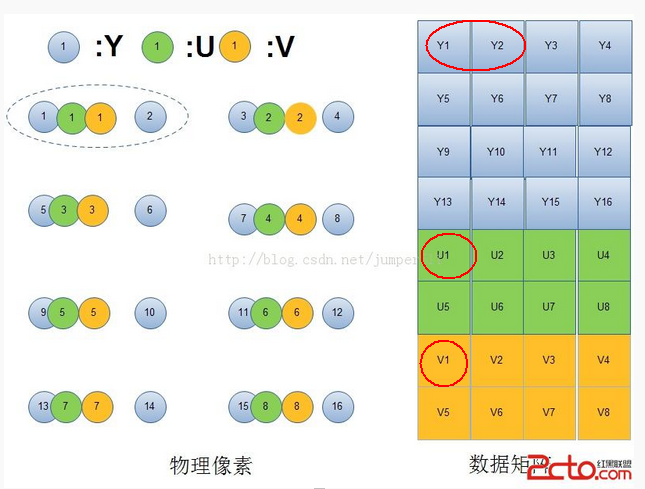
COLOR\_FormatYUV420PackedPlanar = 20;



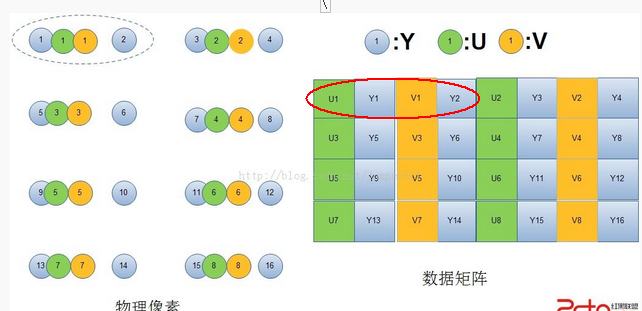
COLOR\_FormatYUV420SemiPlanar = 21; // Android是NV12，还有一种是NV21（属于YUV420）



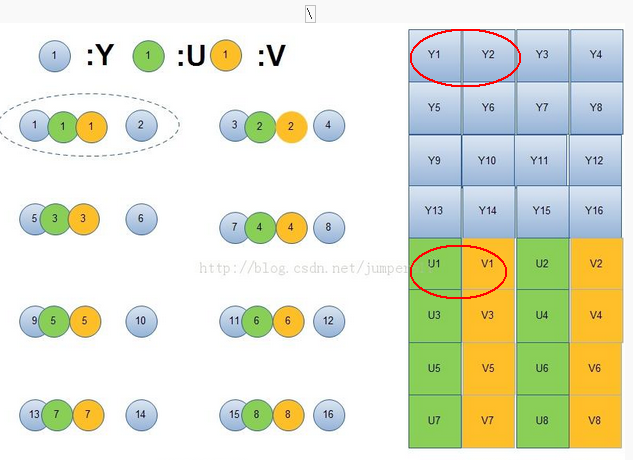
COLOR\_FormatYUV422Planar = 22; //  YUV422P（属于YUV422）



COLOR\_FormatYUV422PackedPlanar = 23;



COLOR\_FormatYUV422SemiPlanar = 24;



COLOR\_FormatYCbYCr = 25; YUV422 Y U Y V YUVY 格式 （属于YUV422）

COLOR\_FormatYCrYCb = 26; YUV422 Y V Y U

COLOR\_FormatCbYCrY = 27; YUV422 U Y V Y UYVY 格式 （属于YUV422）

COLOR\_FormatCrYCbY = 28; YUV422 V Y U Y

COLOR\_FormatYUV420PackedSemiPlanar = 39; // ???? 就是 COLOR\_FormatYUV420Planar ??

COLOR\_FormatYUV422PackedSemiPlanar = 40; // ??? 就是 COLOR\_FormatYUV422Planar ??

另外还有两种：

// COLOR\_FormatSurface indicates that the data will be a GraphicBuffer metadata reference.

// In OMX this is called OMX\_COLOR\_FormatAndroidOpaque. ？？？？？

public static final int COLOR\_FormatSurface = 0x7F000789;

// This corresponds to YUV\_420\_888 format ？？？？？

public static final int COLOR\_FormatYUV420Flexible = 0x7F420888;

参考 ： <http://android.xsoftlab.net/reference/android/graphics/ImageFormat.html> ？？？？？

**YUV 格式通常有两大类:打包(packed)格式和平面(planar)格式。**

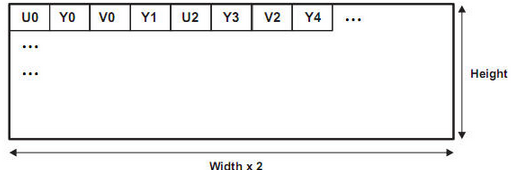
**打包(packed)格式**:将 YUV 分量存放在同一个数组中,通常是几个相邻的像素组成一个宏像素(macro-pixel)

**交错模式 Interleaved**

**UYVY422 (YUV交叉存放 顺序是 U Y V Y)**

ffmpeg  中  PIX\_FMT\_UYVY422   ///< packed YUV 4:2:2, 16bpp, Cb Y0 Cr Y1

android  OMX 中  OMX\_COLOR\_FormatCbYCrY    //CbYCrY  : Organized as 16bit UYVY (i.e. CbYCrY)



**平面(planar)格式**:使用三个数组分开存放 YUV 三个分量,就像是一个三维平面一样

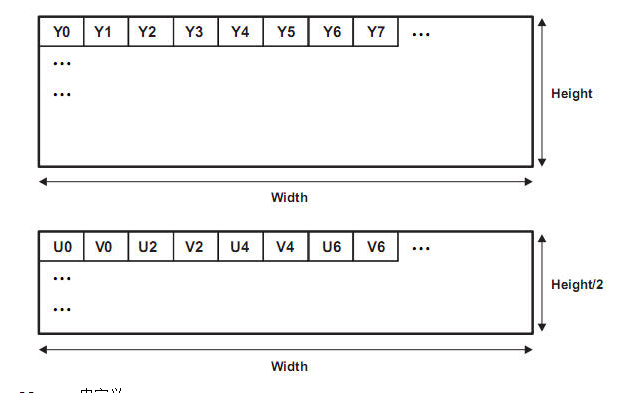
**半平面**Semi-Planar

**NV12**: YYYYYYYY UVUV     => **YUV420SP**

ffmpeg中定义PIX\_FMT\_NV12,

///< planar YUV 4:2:0, 12bpp, 1 plane for Y and 1 plane for the UV components, which are interleaved (first byte U and the following byte V)

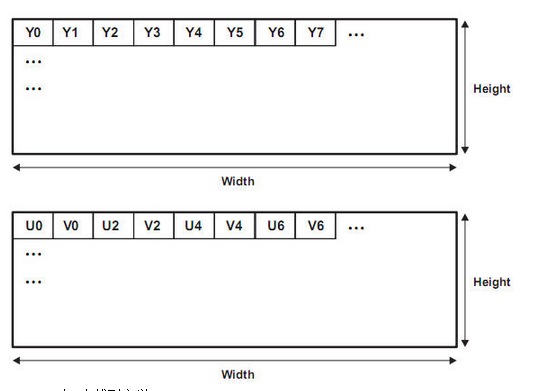
android OMX中定义 OMX\_COLOR\_FormatYUV420SemiPlanar    //YUV420SemiPlanar       : Two arrays, one is all Y, the other is U and V



YUV422SP??

ffmpeg 中 未找到定义

android  OMX  中 OMX\_COLOR\_FormatYUV422SemiPlanar    ////YUV422SemiPlanar       : Two arrays, one is all Y, the other is U and V

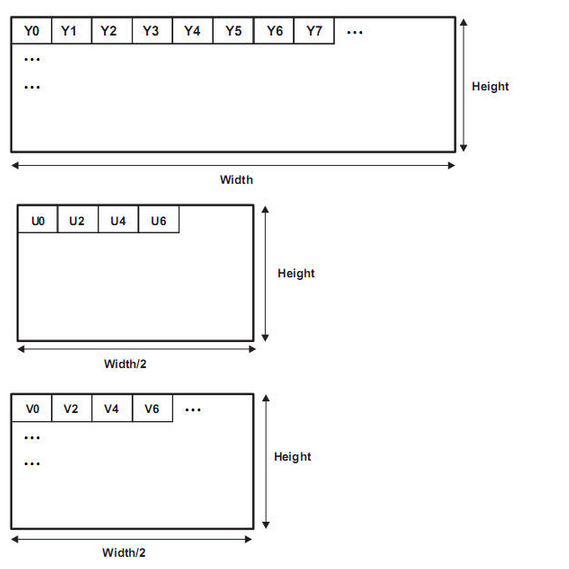


**平面** Planar

**YUV422P**

ffmpeg 中的定义  PIX\_FMT\_YUV422P,   ///< planar YUV 4:2:2, 16bpp, (1 Cr & Cb sample per 2x1 Y samples)

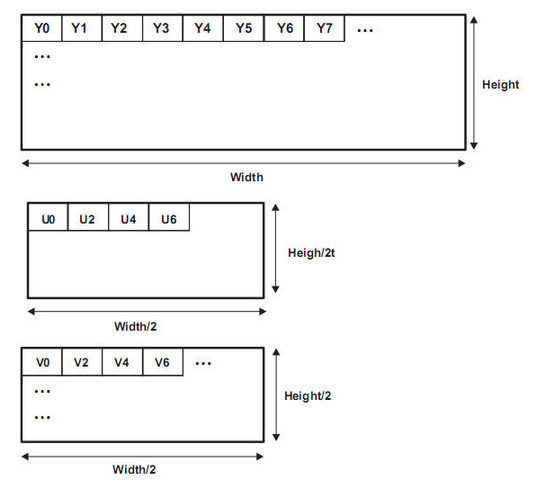
android  OMX  中的定义  OMX\_COLOR\_FormatYUV422Planar     //\*  YUV422Planar           : Three arrays Y,U,V.



**YUV420P I420**: YYYYYYYY UU VV    =>YUV420P

ffmpeg 中定义  PIX\_FMT\_YUV420P    //< planar YUV 4:2:0, 12bpp, (1 Cr & Cb sample per 2x2 Y samples)

android OMX 中定义  OMX\_COLOR\_FormatYUV420Planar     //  \*  YUV420Planar    : Three arrays Y,U,V



关于yuv 格式-Semi Planar和Planar

<http://blog.csdn.net/sunnylgz/article/details/7580628>

<http://www.2cto.com/kf/201403/287429.html>

详解YUV数据格式 --- 存储方式

<http://wenku.baidu.com/link?url=p7lynOB4Hyd4ErkZ5O1TU3xLQ5i0Qf2iqYdfXFMF1fYBeYqqF8jEMDgceJSFHtkjKlKcz-ByYbxcgxIyFc2qzQSIdPXLH0UG2Khz9-w7Bra>

Android MediaCodec stuff -- bigflake的博客

<http://bigflake.com/mediacodec/>

