

W3PDATAAB	0x77100334	读/写	索引 A,B 的窗口 3 调色板数据	0x0000_0000
W3PDATAACD	0x77100338	读/写	索引 C,D 的窗口 3 调色板数据	0x0000_0000
W3PDATAAEF	0x7710033C	读/写	索引 E,F 的窗口 3 调色板数据	0x0000_0000

W3PDATAxx	位	描述		初始状态
ODD_VAL	[31:16]	LUT 值寄存器		0
EVEN_VAL	[15:0]	Lut 值寄存器		0

14.4.61. 窗口 4 的调色板数据

寄存器	地址	读/写	描述	复位值
W2PDATA01	0x77100340	读/写	索引 0,1 的窗口 4 调色板数据	0x0000_0000
W2PDATA23	0x77100344	读/写	索引 2,3 的窗口 4 调色板数据	0x0000_0000

W4PDATAxx	位	描述		初始状态
ODD_VAL	[31:16]	LUT 值寄存器		0
EVEN_VAL	[15:0]	Lut 值寄存器		0

14.4.62. WIN0 调色板 RAM 访问地址（不是 SFR）

索引	地址	读/写	描述	复位值
WIN0_PAENTRY0	0x77100400	读/写	窗口 0 调色板进入 0 地址	未定义
WIN0_PAENTRY1	0x77100404	读/写	窗口 0 调色板进入 1 地址	未定义
-	-	-	-	-
WIN0_PAENTRY255	0x771007FC	读/写	窗口 0 调色板进入 255 地址	未定义

14.4.63. WIN1 调色板 RAM 访问地址（不是 SFR）

索引	地址	读/写	描述	复位值
WIN1_PAENTRY0	0x77100400	读/写	窗口 1 调色板进入 0 地址	未定义
WIN1_PAENTRY1	0x77100404	读/写	窗口 1 调色板进入 1 地址	未定义
-	-	-	-	-
WIN1_PAENTRY255	0x771007FC	读/写	窗口 1 调色板进入 255 地址	未定义

16 TV 定标器（后处理器 VER2.5）

这节主要描述S3C6410X中TV定标器的功能和使用方法。

16.1 TV 定标器概述

除了FIFO的大小和输入FIFO模式，TV定标器和后处理器相似。TV定标器执行视频/图像缩放、视频格式转换和色彩空间转换。如图16-1所示，它是由数据通路、DMA控制器和寄存器文件组成。

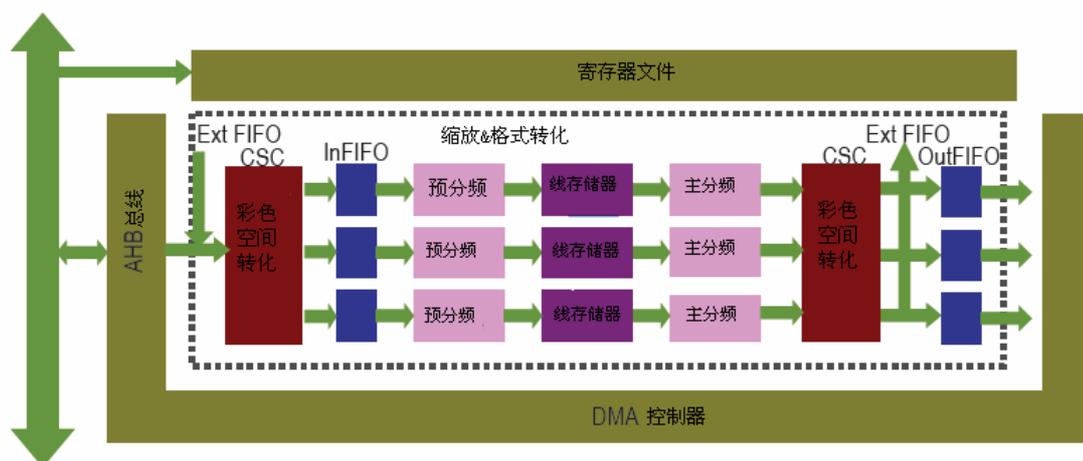


图 16-1 TV 定标器模块图

TV定标器包含下面特性：

- 兼容AMBA AHB v2.0的接口。
- 用于视频/图解比例尺放大/缩小或者物象像放大/缩小的3通道缩放管线。
- 视频输入格式：420、422格式。
- 图像输入格式：16位（565格式）或24位。
- 图像输出到存储器的格式：16位（565格式）/24位图形数据（仅逐行扫描）。
- 视频输出到存储器的格式：YCbCr420, YCbCr422。
- 输入到外部FIFO的格式：YCbCr444 / RGB（24位），用于隔行和逐行的扫描。
- 输出到外部FIFO的格式：YCbCr444 / RGB（30位），用于隔行和逐行的扫描。

- 自运行模式操作。
- 可设计的源图像和目标图像可达到1440x1024的分辨率。
- 可设计的缩放比率。
- 视频信号格式转换。
- 从YCbCr到RGB的色彩空间转换。
- 从RGB到YCbCr的色彩空间转换。
- 对于DMA，最大脉冲长度增加到16位。
- 带有AHB接口时钟的独立处理时钟。

16.2 源和目标图像数据格式

在FIMV TV定标器的顶部，有两种数据模式，DMA模式和FIFO模式，如图16-2所示。

在FIFO模式（如果LCDPathEnable = 1），目标图像传送到显示控制器中（或者其它带有FIFO接口的IP）的FIFO，没有附加的存储器带宽（比如TV定标器到存储器和存储器到显示控制器）。

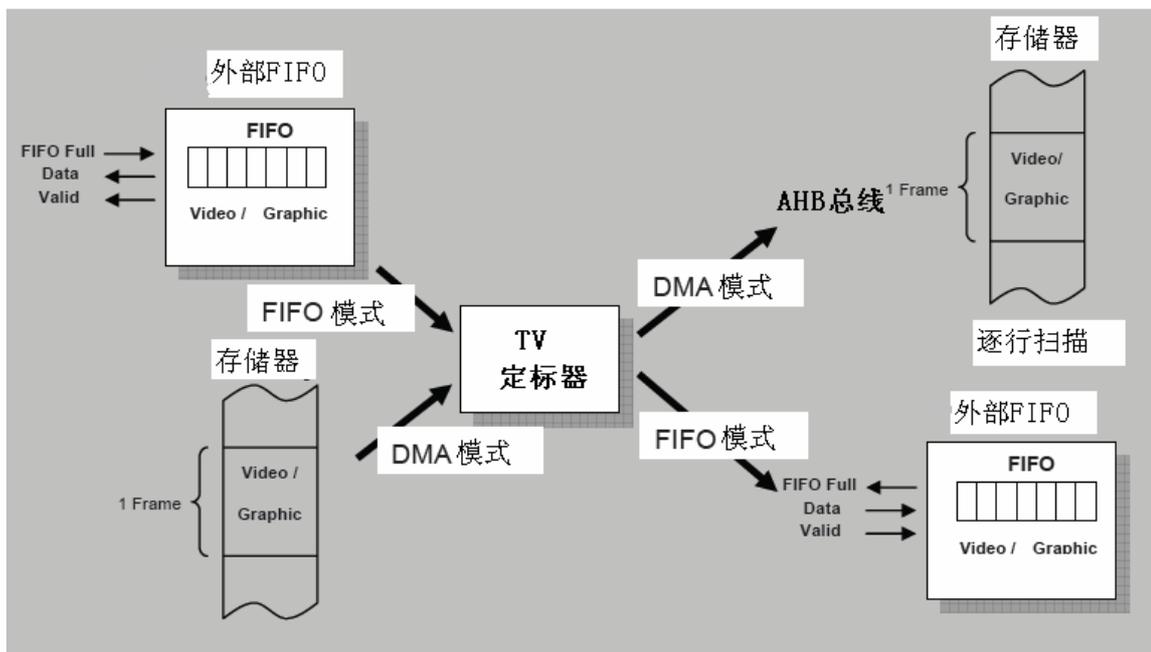


图 16-2 带有两个输入输出模式的 TV 定标器模块图

1 DMA模式操作

如表16-1a所示，通过模式配置能选择各种源和目标图像格式。源图像格式是YCbCr420、YCbCr422、RGB16位（565格式）和RGB 24位格式中的一种。目标图像格式同于源图像格式，是YCbCr420、YCbCr422、RGB 16位（565格式）和 RGB 24位格式中的一种。详细的控制信号如表16-1b所示。

表16-1a 对于视频/图像源格式和相应的数据格式的模式配置

MODE[8]	MODE[3]	MODE[2]	MODE[1]	MODE[15][0]	描述	
					视频/图像格式	图12-30和12-31中的数据格式
SRC420	InRGB	INTERLEAVE	InRGB格式	InYCbCr 格式		
1	0	0	1	×	420 YCbCr格式	A
0	0	1	1	00/10	422YCbYCr格式	B/B'
0	0	1	1	01/11	422CbYCrY格式	C/C'
0	1	1	1	×	RGB 24位真彩色	D
0	1	1	0	×	RGB 16位格式	E

表16-1b 视频/图像目标格式和相应的数据格式的模式配置

MODE[18]	MODE[17]	MODE[20][19]	MODE[4]	MODE[15][0]	描述	
					视频/图像格式	图12-30和12-31中的数据格式
OutRGB	DST420	OutYCbCr格式	OutRGB格式	InYCbCr 格式		
0	1	×	×	×	420 YCbCr格式	A
0	0	00/10	×	00/10	422YCbYCr格式	B/B'
0	0	01/11	×	01/11	422CbYCrY格式	C/C'
1	×	×	1	×	RGB 24位真彩色	D
1	×	×	0	×	RGB 16位格式	E

在YCbCr420源和目标图像格式的情况下，Y、Cb和Cr都存储在各自的独立地址空间，没有交叉，如图16-3（a）和图16-4的Case A所示。其它情况下，在统一的地址空间，使用字节或者半字交叉，如图16-3（b）。YCbCr422源图像的字节交叉顺序能从YCbYCr 或 CbYCrY中选择，如图16-3（b）和图16-4的case B和case C。RGB 24位的字节顺序和RGB位的半字顺序如图16-3（b）和16-4所示。

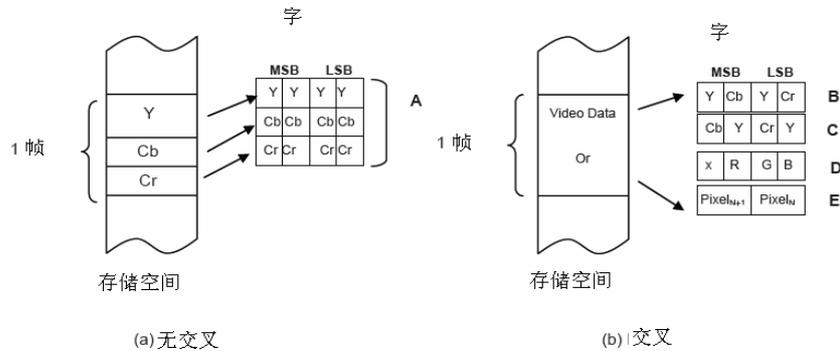


图16-3 数据格式外部存储

YCbCr420和YCbCr422源图像格式下，是选择MPEG4格式还是选择MPEG2/H.263格式，由色度信息的采样位置决定，如图16-4所示。当源和目标图像是YCbCr格式，目标图像和源图像有相同的采样位置。如果源图像是RGB格式，目标图像是YCbCr格式，色度信息的采样位置和亮度位置相同。

所有的源图像和目标图像数据必须存储在存储器系统，以字边界对齐。意思是既不支持字节也不支持半字大小的DMA操作。因此，源和目标图像的宽度必须选择满足字边界大小的。

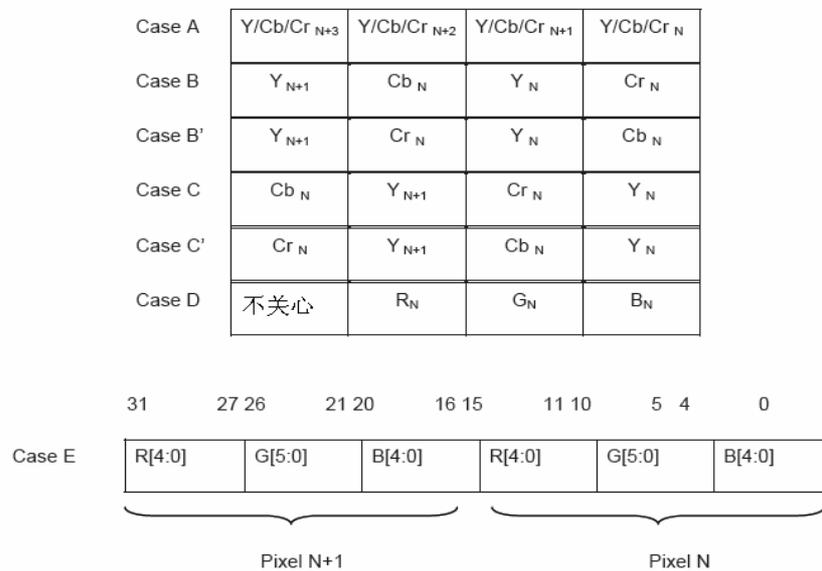


图16-4 字节或半字组织模块图

2. FIFO 模式操作

输出数据格式由MODE[18]决定，如表16-1b所示。数据格式被固定到30位数据，每十位构成RGB 或

YCbCr444。如果ExtFIFOIn或LCDPathEnable设置为“1”，表16-1 b中其它特殊模式配置信号被忽略。FIFO模式中，输入输出数据格式如表16-2所示。

表16-2 FIFO模式中，输入输出数据格式

OutRGB (MODE[18])	输出数据格式 (LCDPathEnable = 1) (隔行/逐行)
0	YCbCr 444
1	RGB 30位

输出FIFO模式中，无论是隔行扫描模式还是逐行扫描模式，都是通过“隔行”控制寄存器进行选择的。只有当LCDPathEnable =1 时，“隔行”控制位有效，否则支持逐行扫描模式。

隔行扫描模式有效 (LCDPathEnable = 1和Interlace = 1)，由奇数场和偶数场组成的帧管理自动运行。因此，该帧管理对于隔行扫描和逐行扫描模式是一样的。它也兼容FIMV TV定标器。

16.3 图像尺寸和比率

RGB图像源大小由沿水平方向和垂直方向上的像素数决定。YCbCr420 和 YCbCr422 源图像大小只由Y采样沿水平和垂直方向决定。目标图像大小由最终RGB图解的图像尺度决定。如果源图像是YCbCr 图像，经色度空间转换后，RGB图解的图像最终尺度定下来。

像前面所说的，SRC_Width和DST_Width受到字边界的限制，所以水平像素数能用kn表示，其中n=1、2、3、...，对于24bppRGB /16bppRGB / YCbCr420图像,k分别为1 / 2 / 8。而且SRC_Width必须是PreScale_H_Ratio的4倍，并且SRC_Height必须是PreScale_V_Ratio的倍数。源图像和目标图像模块图如图16-5所示。

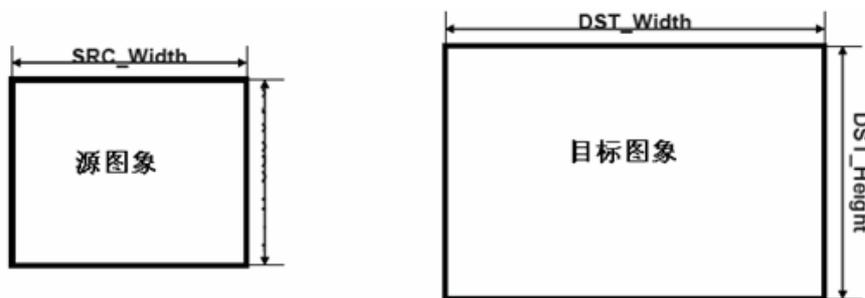


图16-5 源和目标图像大小模块图

其它的控制寄存器，预缩放图像大小、预缩放比率、预缩放移位比率和主要的预缩放比率由下面的等

式定义:

```
if ( SRC_Width >= 64 × DST_Width ) { Exit(-1); /*超出水平比例尺范围 */ }
else if ( SRC_Width >= 32 × DST_Width ) { PreScale_H_Ratio = 32; H_Shift = 5; }
else if ( SRC_Width >= 16 × DST_Width ) { PreScale_H_Ratio = 16; H_Shift = 4; }
else if ( SRC_Width >= 8 × DST_Width ) { PreScale_H_Ratio = 8; H_Shift = 3; }
else if ( SRC_Width >= 4 × DST_Width ) { PreScale_H_Ratio = 4; H_Shift = 2; }
else if ( SRC_Width >= 2 × DST_Width ) { PreScale_H_Ratio = 2; H_Shift = 1; }
else { PreScale_H_Ratio = 1; H_Shift = 0; }
```

```
PreScale_DSTWidth = SRC_Width / PreScale_H_Ratio;
```

```
dx = ( SRC_Width << 8 ) / ( DST_Width << H_Shift);
```

```
If ( SRC_Height >= 64 × DST_Height ) { Exit(-1); /* 超出垂直比例尺范围 */ }
else if ( SRC_Height >= 32 × DST_Height ) { PreScale_V_Ratio = 32; V_Shift = 5; }
else if ( SRC_Height >= 16 × DST_Height ) { PreScale_V_Ratio = 16; V_Shift = 4; }
else if ( SRC_Height >= 8 × DST_Height ) { PreScale_V_Ratio = 8; V_Shift = 3; }
else if ( SRC_Height >= 4 × DST_Height ) { PreScale_V_Ratio = 4; V_Shift = 2; }
else if ( SRC_Height >= 2 × DST_Height ) { PreScale_V_Ratio = 2; V_Shift = 1; }
else { PreScale_V_Ratio = 1; V_Shift = 0; }
```

```
PreScale_DSTHeight = SRC_Height / PreScale_V_Ratio;
```

```
dy = ( SRC_Height << 8 ) / ( DST_Height << V_Shift);
```

```
PreScale_SHFactor = 10 - ( H_Shift + V_Shift);
```

16.4 源和目标图像的 DMA 操作

DMA操作有三种地址，即起始地址、结束地址和偏移地址。每个地址由Y/Cb/Cr三个源地址和RGB/oCb/oCr三个目标地址组成。如果一个源图像以非交叉格式（像YCbCr420）存储，所有的源地址分量是有效的，如图16-6(a)所示。如果一个源图像以交叉格式（像RGB图像格式或YCbCr422格式）存储，三个

源分量中只有Y是有效的，两个色度地址分量是无效的，如图16-6 (b)所示。如果目标图像以非交叉格式（像YCbCr420）存储，所有的源地址分量RGB/oCb/oCr是有效的，如图16-6 (a)所示。如果一个源图像以交叉格式（像RGB图像格式或YCbCr422格式）存储，三个源分量中只有RGB是有效的，oCb/oCr两个色度地址分量是无效的，如图16-6 (b)所示。

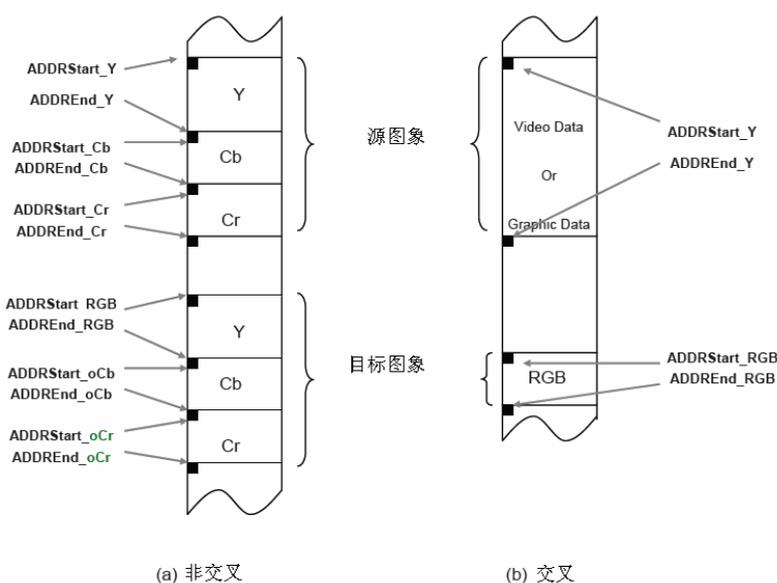


图16-6 根据存储起配置类型，起始和结束地址的设置

1. 起始地址

ADDRStart_Y/Cb/Cr/RGB/oCb/oCr 的起始地址指向第一个字的地址，Y/Cb/Cr/RGB/oCb/oCr 相应的分量被读取或者写入该地址。每一个必须以字(例如：ADDRStart_X[1:0] = 00)对齐。只有 YCbCr420 源图像格式，ADDRStart_Cb 和 ADDRStart_Cr 是有效的。只有对于 YCbCr420 目标图像格式，ADDRStart_oCb 和 ADDRStart_oCr 是有效的。

2. 结束地址

ADDRStart_Y

= ADDRStart_Y + 存储器大小 (对于 Y/RGB 分量)

= ADDRStart_Y + (SRC_Width × SRC_Height) × ByteSize_Per_Pixel + Offset_Y × (SRC_Height-1)

ADDREnd_Cb (对于 YCbCr420 源格式有效的)
= ADDRStart_Cb + 存储器大小 (Cb 分量)
= ADDRStart_Cb + (SRC_Width/2 × SRC_Height/2) × ByteSize_Per_Pixel + Offset_Cb ×
(SRC_Height/2-1)

ADDREnd_Cr (对于 YCbCr420 源格式有效的)
= ADDRStart_Cr + 存储器大小 (Cr 分量)
= ADDRStart_Cr + (SRC_Width/2 × SRC_Height/2) × ByteSize_Per_Pixel + Offset_Cr ×
(SRC_Height/2-1)

ADDREnd_RGB
= ADDRStart_RGB + 存储器大小 (RGB 数据或 Y 分量)
= ADDRStart_RGB + (DST_Width × DST_Height) × ByteSize_Per_Pixel + Offset_RGB ×
(DST_Height-1)

ADDREnd_oCb (对于 YCbCr420 目标格式有效的)
= ADDRStart_oCb + 存储器大小 (Cb 分量)
= ADDRStart_oCb + (SRC_Width/2 × SRC_Height/2) × ByteSize_Per_Pixel + Offset_Cb ×
(SRC_Height/2-1)

ADDREnd_oCr (有效的对于 YCbCr420 目标格式)
= ADDRStart_oCr + 存储器大小 (Cr 分量) for the component of
= ADDRStart_oCr + (SRC_Width/2 × SRC_Height/2) × ByteSize_Per_Pixel + Offset_Cr ×
(SRC_Height/2-1)

其中

Offset_Y/Cb/Cr/RGB

= 存储器大小 (每一条水平线偏移量)

= 水平偏移的像素数(采样) × ByteSize_Per_Pixel (或采样)