RGB Formats

These formats encode each pixel as a triplet of RGB values. They are packed formats, meaning that the RGB values for one pixel are stored consecutively in memory and each pixel consumes an integer number of bytes. When the number of bits required to store a pixel is not aligned to a byte boundary, the data is padded with additional bits to fill the remaining byte.

The formats differ by the number of bits per RGB component (typically but not always the same for all components), the order of components in memory, and the presence of an alpha component or additional padding bits.

The usage and value of the alpha bits in formats that support them (named ARGB or a permutation thereof, collectively referred to as alpha formats) depend on the device type and hardware operation. :ref:`Capture <capture>` devices (including capture queues of mem-to-mem devices) fill the alpha component in memory. When the device captures an alpha channel the alpha component will have a meaningful value. Otherwise, when the device doesn't capture an alpha channel but can set the alpha bit to a user-configurable value, the :ref:`V4L2_CID_ALPHA_COMPONENT <v4l2-alpha-component>` control is used to specify that alpha value, and the alpha component of all pixels will be set to the value specified by that control. Otherwise a corresponding format without an alpha component (XRGB or XBGR) must be used instead of an alpha format.

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\userspace-api\media\v4l\[linux-master][Documentation][userspace-api][media][v4l]pixfmt-rgb.rst`, **line 19**); ***backlink***

Unknown interpreted text role "ref".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\userspace-api\media\v4l\[linux-master][Documentation][userspace-api][media][v4l]pixfmt-rgb.rst`, **line 19**); ***backlink***

Unknown interpreted text role "ref".

:ref:`Output <output>` devices (including output queues of mem-to-mem devices and :ref:`video output overlay <osd>` devices) read the alpha component from memory. When the device processes the alpha channel the alpha component must be filled with meaningful values by applications. Otherwise a corresponding format without an alpha component (XRGB or XBGR) must be used instead of an alpha format.

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\userspace-api\media\v4l\[linux-master][Documentation][userspace-api][media][v4l]pixfmt-rgb.rst`, **line 31**); ***backlink***

Unknown interpreted text role "ref".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\userspace-api\media\v4l\[linux-master][Documentation][userspace-api][media][v4l]pixfmt-rgb.rst`, **line 31**); ***backlink***

Unknown interpreted text role "ref".

Formats that contain padding bits are named XRGB (or a permutation thereof). The padding bits contain undefined values and must be ignored by applications, devices and drivers, for both :ref:`capture` and :ref:`output` devices.

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\userspace-api\media\v4l\[linux-master][Documentation][userspace-api][media][v4l]pixfmt-rgb.rst`, **line 38**); ***backlink***

Unknown interpreted text role "ref".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\userspace-api\media\v4l\[linux-master][Documentation][userspace-api][media][v4l]pixfmt-rgb.rst`, **line 38**); ***backlink***

Unknown interpreted text role "ref".

Note

- In all the tables that follow, bit 7 is the most significant bit in a byte.
- 'r', 'g' and 'b' denote bits of the red, green and blue components respectively. 'a' denotes bits of the alpha component (if supported by the format), and 'x' denotes padding bits.

Less Than 8 Bits Per Component

These formats store an RGB triplet in one, two or four bytes. They are named based on the order of the RGB components as seen in a 8-, 16- or 32-bit word, which is then stored in memory in little endian byte order (unless otherwise noted by the presence of bit 31 in the 4CC value), and on the number of bits for each component. For instance, the RGB565 format stores a pixel in a 16-bit word [15:0] laid out at as [R_4 R_3 R_2 R_1 R_0 G_5 G_4 G_3 G_2 G_1 G_0 B_4 B_3 B_2 B_1 B_0], and stored in memory in two bytes, [R_4 R_3 R_2 R_1 R_0 G_5 G_4 G_3] followed by [G_2 G_1 G_0 B_4 B_3 B_2 B_1 B_0].

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\userspace-api\media\v4l\[linux-master][Documentation][userspace-api][media][v4l]pixfmt-rgb.rst`, **line 72**)

Unknown directive type "tabularcolumns".

```
.. tabularcolumns:: |p{2.8cm}|p{2.0cm}|p{0.22cm}|p{0.22cm}|p{0.22cm}|p{0.22cm}|p{0.22cm}|p{0.22cm}|p{0.22cm}|p{0.22cm}|p{0.22cm}|p
```

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\userspace-api\media\v4l\[linux-master][Documentation][userspace-api][media][v4l]pixfmt-rgb.rst`, **line 75**)

Unknown directive type "flat-table".

```
.. flat-table:: RGB Formats With Less Than 8 Bits Per Component
    :header-rows:  2
    :stub-columns: 0

    * - Identifier
      - Code
      - :cspan:`7` Byte 0 in memory
      - :cspan:`7` Byte 1
      - :cspan:`7` Byte 2
      - :cspan:`7` Byte 3
    * -
      -
      - 7
      - 6
      - 5
      - 4
      - 3
      - 2
      - 1
      - 0

      - 7
      - 6
      - 5
      - 4
      - 3
      - 2
      - 1
      - 0

      - 7
      - 6
      - 5
      - 4
      - 3
      - 2
      - 1
      - 0

      - 7
      - 6
      - 5
      - 4
      - 3
      - 2
      - 1
      - 0
    * .. _V4L2-PIX-FMT-RGB332:

      - ``V4L2_PIX_FMT_RGB332``
      - 'RGB1'

      - r\ :sub:`2`
      - r\ :sub:`1`
      - r\ :sub:`0`
      - g\ :sub:`2`
      - g\ :sub:`1`
      - g\ :sub:`0`
      - b\ :sub:`1`
      - b\ :sub:`0`
      -
    * .. _V4L2-PIX-FMT-ARGB444:

      - ``V4L2_PIX_FMT_ARGB444``
      - 'AR12'

      - g\ :sub:`3`
      - g\ :sub:`2`
      - g\ :sub:`1`
      - g\ :sub:`0`
      - b\ :sub:`3`
      - b\ :sub:`2`
      - b\ :sub:`1`
      - b\ :sub:`0`

      - a\ :sub:`3`
      - a\ :sub:`2`
      - a\ :sub:`1`
      - a\ :sub:`0`
      - r\ :sub:`3`
      - r\ :sub:`2`
      - r\ :sub:`1`
      - r\ :sub:`0`
      -
    * .. _V4L2-PIX-FMT-XRGB444:

      - ``V4L2_PIX_FMT_XRGB444``
      - 'XR12'

      - g\ :sub:`3`
      - g\ :sub:`2`
      - g\ :sub:`1`
      - g\ :sub:`0`
      - b\ :sub:`3`
      - b\ :sub:`2`
      - b\ :sub:`1`
      - b\ :sub:`0`

      - x
      - x
      - x
      - x
      - r\ :sub:`3`
      - r\ :sub:`2`
      - r\ :sub:`1`
      - r\ :sub:`0`
      -
    * .. _V4L2-PIX-FMT-RGBA444:

      - ``V4L2_PIX_FMT_RGBA444``
      - 'RA12'

      - b\ :sub:`3`
      - b\ :sub:`2`
      - b\ :sub:`1`
      - b\ :sub:`0`
      - a\ :sub:`3`
      - a\ :sub:`2`
      - a\ :sub:`1`
      - a\ :sub:`0`

      - r\ :sub:`3`
      - r\ :sub:`2`
      - r\ :sub:`1`
      - r\ :sub:`0`
      - g\ :sub:`3`
      - g\ :sub:`2`
      - g\ :sub:`1`
      - g\ :sub:`0`
      -
    * .. _V4L2-PIX-FMT-RGBX444:

      - ``V4L2_PIX_FMT_RGBX444``
      - 'RX12'

      - b\ :sub:`3`
      - b\ :sub:`2`
      - b\ :sub:`1`
      - b\ :sub:`0`
      - x
      - x
      - x
      - x

      - r\ :sub:`3`
      - r\ :sub:`2`
      - r\ :sub:`1`
      - r\ :sub:`0`
      - g\ :sub:`3`
      - g\ :sub:`2`
      - g\ :sub:`1`
      - g\ :sub:`0`
      -
    * .. _V4L2-PIX-FMT-ABGR444:

      - ``V4L2_PIX_FMT_ABGR444``
      - 'AB12'

      - g\ :sub:`3`
      - g\ :sub:`2`
      - g\ :sub:`1`
      - g\ :sub:`0`
      - r\ :sub:`3`
      - r\ :sub:`2`
      - r\ :sub:`1`
      - r\ :sub:`0`

      - a\ :sub:`3`
      - a\ :sub:`2`
      - a\ :sub:`1`
      - a\ :sub:`0`
      - b\ :sub:`3`
      - b\ :sub:`2`
      - b\ :sub:`1`
      - b\ :sub:`0`
      -
    * .. _V4L2-PIX-FMT-XBGR444:

      - ``V4L2_PIX_FMT_XBGR444``
      - 'XB12'

      - g\ :sub:`3`
      - g\ :sub:`2`
      - g\ :sub:`1`
      - g\ :sub:`0`
      - r\ :sub:`3`
      - r\ :sub:`2`
      - r\ :sub:`1`
      - r\ :sub:`0`

      - x
      - x
      - x
      - x
      - b\ :sub:`3`
      - b\ :sub:`2`
      - b\ :sub:`1`
      - b\ :sub:`0`
      -
    * .. _V4L2-PIX-FMT-BGRA444:

      - ``V4L2_PIX_FMT_BGRA444``
      - 'BA12'

      - r\ :sub:`3`
      - r\ :sub:`2`
      - r\ :sub:`1`
      - r\ :sub:`0`
      - a\ :sub:`3`
      - a\ :sub:`2`
      - a\ :sub:`1`
      - a\ :sub:`0`

      - b\ :sub:`3`
      - b\ :sub:`2`
      - b\ :sub:`1`
      - b\ :sub:`0`
      - g\ :sub:`3`
      - g\ :sub:`2`
      - g\ :sub:`1`
      - g\ :sub:`0`
      -
    * .. _V4L2-PIX-FMT-BGRX444:

      - ``V4L2_PIX_FMT_BGRX444``
      - 'BX12'

      - r\ :sub:`3`
      - r\ :sub:`2`
      - r\ :sub:`1`
      - r\ :sub:`0`
      - x
      - x
      - x
      - x

      - b\ :sub:`3`
      - b\ :sub:`2`
      - b\ :sub:`1`
      - b\ :sub:`0`
      - g\ :sub:`3`
      - g\ :sub:`2`
      - g\ :sub:`1`
      - g\ :sub:`0`
      -
    * .. _V4L2-PIX-FMT-ARGB555:

      - ``V4L2_PIX_FMT_ARGB555``
      - 'AR15'

      - g\ :sub:`2`
      - g\ :sub:`1`
      - g\ :sub:`0`
      - b\ :sub:`4`
      - b\ :sub:`3`
      - b\ :sub:`2`
      - b\ :sub:`1`
      - b\ :sub:`0`

      - a
      - r\ :sub:`4`
      - r\ :sub:`3`
      - r\ :sub:`2`
      - r\ :sub:`1`
      - r\ :sub:`0`
      - g\ :sub:`4`
      - g\ :sub:`3`
      -
    * .. _V4L2-PIX-FMT-XRGB555:

      - ``V4L2_PIX_FMT_XRGB555``
      - 'XR15'

      - g\ :sub:`2`
      - g\ :sub:`1`
      - g\ :sub:`0`
      - b\ :sub:`4`
      - b\ :sub:`3`
      - b\ :sub:`2`
      - b\ :sub:`1`
      - b\ :sub:`0`

      - x
      - r\ :sub:`4`
      - r\ :sub:`3`
      - r\ :sub:`2`
      - r\ :sub:`1`
      - r\ :sub:`0`
      - g\ :sub:`4`
      - g\ :sub:`3`
      -
    * .. _V4L2-PIX-FMT-RGBA555:

      - ``V4L2_PIX_FMT_RGBA555``
      - 'RA15'

      - g\ :sub:`1`
      - g\ :sub:`0`
      - b\ :sub:`4`
      - b\ :sub:`3`
      - b\ :sub:`2`
      - b\ :sub:`1`
      - b\ :sub:`0`
      - a

      - r\ :sub:`4`
      - r\ :sub:`3`
      - r\ :sub:`2`
      - r\ :sub:`1`
      - r\ :sub:`0`
      - g\ :sub:`4`
      - g\ :sub:`3`
      - g\ :sub:`2`
      -
    * .. _V4L2-PIX-FMT-RGBX555:

      - ``V4L2_PIX_FMT_RGBX555``
      - 'RX15'

      - g\ :sub:`1`
      - g\ :sub:`0`
      - b\ :sub:`4`
      - b\ :sub:`3`
      - b\ :sub:`2`
      - b\ :sub:`1`
      - b\ :sub:`0`
      - x

      - r\ :sub:`4`
      - r\ :sub:`3`
      - r\ :sub:`2`
      - r\ :sub:`1`
      - r\ :sub:`0`
      - g\ :sub:`4`
      - g\ :sub:`3`
      - g\ :sub:`2`
      -
    * .. _V4L2-PIX-FMT-ABGR555:

      - ``V4L2_PIX_FMT_ABGR555``
      - 'AB15'

      - g\ :sub:`2`
      - g\ :sub:`1`
      - g\ :sub:`0`
      - r\ :sub:`4`
      - r\ :sub:`3`
      - r\ :sub:`2`
      - r\ :sub:`1`
      - r\ :sub:`0`

      - a
      - b\ :sub:`4`
      - b\ :sub:`3`
      - b\ :sub:`2`
      - b\ :sub:`1`
      - b\ :sub:`0`
      - g\ :sub:`4`
      - g\ :sub:`3`
      -
    * .. _V4L2-PIX-FMT-XBGR555:

      - ``V4L2_PIX_FMT_XBGR555``
      - 'XB15'

      - g\ :sub:`2`
      - g\ :sub:`1`
      - g\ :sub:`0`
      - r\ :sub:`4`
      - r\ :sub:`3`
      - r\ :sub:`2`
      - r\ :sub:`1`
      - r\ :sub:`0`

      - x
      - b\ :sub:`4`
      - b\ :sub:`3`
      - b\ :sub:`2`
      - b\ :sub:`1`
      - b\ :sub:`0`
      - g\ :sub:`4`
      - g\ :sub:`3`
      -
    * .. _V4L2-PIX-FMT-BGRA555:

      - ``V4L2_PIX_FMT_BGRA555``
      - 'BA15'

      - g\ :sub:`1`
      - g\ :sub:`0`
      - r\ :sub:`4`
      - r\ :sub:`3`
      - r\ :sub:`2`
      - r\ :sub:`1`
      - r\ :sub:`0`
      - a

      - b\ :sub:`4`
      - b\ :sub:`3`
      - b\ :sub:`2`
      - b\ :sub:`1`
      - b\ :sub:`0`
      - g\ :sub:`4`
      - g\ :sub:`3`
      - g\ :sub:`2`
      -
```

```
* .. _V4L2-PIX-FMT-BGRX555:

  - ``V4L2_PIX_FMT_BGRX555``
  - 'BX15'

  - g\ :sub:`1`
  - g\ :sub:`0`
  - r\ :sub:`4`
  - r\ :sub:`3`
  - r\ :sub:`2`
  - r\ :sub:`1`
  - r\ :sub:`0`
  - x

  - b\ :sub:`4`
  - b\ :sub:`3`
  - b\ :sub:`2`
  - b\ :sub:`1`
  - b\ :sub:`0`
  - g\ :sub:`4`
  - g\ :sub:`3`
  - g\ :sub:`2`
  -
* .. _V4L2-PIX-FMT-RGB565:

  - ``V4L2_PIX_FMT_RGB565``
  - 'RGBP'

  - g\ :sub:`2`
  - g\ :sub:`1`
  - g\ :sub:`0`
  - b\ :sub:`4`
  - b\ :sub:`3`
  - b\ :sub:`2`
  - b\ :sub:`1`
  - b\ :sub:`0`

  - r\ :sub:`4`
  - r\ :sub:`3`
  - r\ :sub:`2`
  - r\ :sub:`1`
  - r\ :sub:`0`
  - g\ :sub:`5`
  - g\ :sub:`4`
  - g\ :sub:`3`
  -
* .. _V4L2-PIX-FMT-ARGB555X:

  - ``V4L2_PIX_FMT_ARGB555X``
  - 'AR15' | (1 << 31)

  - a
  - r\ :sub:`4`
  - r\ :sub:`3`
  - r\ :sub:`2`
  - r\ :sub:`1`
  - r\ :sub:`0`
  - g\ :sub:`4`
  - g\ :sub:`3`

  - g\ :sub:`2`
  - g\ :sub:`1`
  - g\ :sub:`0`
  - b\ :sub:`4`
  - b\ :sub:`3`
  - b\ :sub:`2`
  - b\ :sub:`1`
  - b\ :sub:`0`
  -
* .. _V4L2-PIX-FMT-XRGB555X:

  - ``V4L2_PIX_FMT_XRGB555X``
  - 'XR15' | (1 << 31)

  - x
  - r\ :sub:`4`
  - r\ :sub:`3`
  - r\ :sub:`2`
  - r\ :sub:`1`
  - r\ :sub:`0`
  - g\ :sub:`4`
  - g\ :sub:`3`

  - g\ :sub:`2`
  - g\ :sub:`1`
  - g\ :sub:`0`
  - b\ :sub:`4`
  - b\ :sub:`3`
  - b\ :sub:`2`
  - b\ :sub:`1`
  - b\ :sub:`0`
  -
* .. _V4L2-PIX-FMT-RGB565X:

  - ``V4L2_PIX_FMT_RGB565X``
  - 'RGBR'

  - r\ :sub:`4`
  - r\ :sub:`3`
  - r\ :sub:`2`
  - r\ :sub:`1`
  - r\ :sub:`0`
  - g\ :sub:`5`
  - g\ :sub:`4`
  - g\ :sub:`3`

  - g\ :sub:`2`
  - g\ :sub:`1`
  - g\ :sub:`0`
  - b\ :sub:`4`
  - b\ :sub:`3`
  - b\ :sub:`2`
  - b\ :sub:`1`
  - b\ :sub:`0`
  -
* .. _V4L2-PIX-FMT-BGR666:

  - ``V4L2_PIX_FMT_BGR666``
  - 'BGRH'

  - b\ :sub:`5`
  - b\ :sub:`4`
  - b\ :sub:`3`
  - b\ :sub:`2`
  - b\ :sub:`1`
  - b\ :sub:`0`
  - g\ :sub:`5`
  - g\ :sub:`4`

  - g\ :sub:`3`
```

```
      - g\ :sub:`2`
      - g\ :sub:`1`
      - g\ :sub:`0`
      - r\ :sub:`5`
      - r\ :sub:`4`
      - r\ :sub:`3`
      - r\ :sub:`2`

      - r\ :sub:`1`
      - r\ :sub:`0`
      - x
      - x
      - x
      - x
      - x
      - x

      - x
      - x
      - x
      - x
      - x
      - x
      - x
      - x
```

8 Bits Per Component

These formats store an RGB triplet in three or four bytes. They are named based on the order of the RGB components as stored in memory, and on the total number of bits per pixel. For instance, RGB24 format stores a pixel with [R_7 R_6 R_5 R_4 R_3 R_2 R_1 R_0] in the first byte, [G_7 G_6 G_5 G_4 G_3 G_2 G_1 G_0] in the second byte and [B_7 B_6 B_5 B_4 B_3 B_2 B_1 B_0] in the third byte. This differs from the DRM format nomenclature that instead use the order of components as seen in a 24- or 32-bit little endian word.

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\userspace-api\media\v4l\[linux-master] [Documentation] [userspace-api] [media] [v4l]pixfmt-rgb.rst`, line 660)

Unknown directive type "flat-table".

```
.. flat-table:: RGB Formats With 8 Bits Per Component
    :header-rows:  1
    :stub-columns: 0

    * - Identifier
      - Code
      - Byte 0 in memory
      - Byte 1
      - Byte 2
      - Byte 3
    * .. _V4L2-PIX-FMT-BGR24:

      - ``V4L2_PIX_FMT_BGR24``
      - 'BGR3'

      - B\ :sub:`7-0`
      - G\ :sub:`7-0`
      - R\ :sub:`7-0`
      -
    * .. _V4L2-PIX-FMT-RGB24:

      - ``V4L2_PIX_FMT_RGB24``
      - 'RGB3'

      - R\ :sub:`7-0`
      - G\ :sub:`7-0`
      - B\ :sub:`7-0`
      -
    * .. _V4L2-PIX-FMT-ABGR32:

      - ``V4L2_PIX_FMT_ABGR32``
      - 'AR24'

      - B\ :sub:`7-0`
      - G\ :sub:`7-0`
      - R\ :sub:`7-0`
      - A\ :sub:`7-0`
    * .. _V4L2-PIX-FMT-XBGR32:

      - ``V4L2_PIX_FMT_XBGR32``
      - 'XR24'

      - B\ :sub:`7-0`
      - G\ :sub:`7-0`
      - R\ :sub:`7-0`
      - X\ :sub:`7-0`
    * .. _V4L2-PIX-FMT-BGRA32:

      - ``V4L2_PIX_FMT_BGRA32``
      - 'RA24'

      - A\ :sub:`7-0`
      - B\ :sub:`7-0`
      - G\ :sub:`7-0`
      - R\ :sub:`7-0`
    * .. _V4L2-PIX-FMT-BGRX32:

      - ``V4L2_PIX_FMT_BGRX32``
      - 'RX24'

      - X\ :sub:`7-0`
      - B\ :sub:`7-0`
      - G\ :sub:`7-0`
      - R\ :sub:`7-0`
    * .. _V4L2-PIX-FMT-RGBA32:

      - ``V4L2_PIX_FMT_RGBA32``
      - 'AB24'

      - R\ :sub:`7-0`
      - G\ :sub:`7-0`
      - B\ :sub:`7-0`
      - A\ :sub:`7-0`
    * .. _V4L2-PIX-FMT-RGBX32:

      - ``V4L2_PIX_FMT_RGBX32``
      - 'XB24'

      - R\ :sub:`7-0`
      - G\ :sub:`7-0`
      - B\ :sub:`7-0`
      - X\ :sub:`7-0`
    * .. _V4L2-PIX-FMT-ARGB32:

      - ``V4L2_PIX_FMT_ARGB32``
      - 'BA24'
```

```
       - A\ :sub:`7-0`
       - R\ :sub:`7-0`
       - G\ :sub:`7-0`
       - B\ :sub:`7-0`
     * .. _V4L2-PIX-FMT-XRGB32:

       - ``V4L2_PIX_FMT_XRGB32``
       - 'BX24'

       - X\ :sub:`7-0`
       - R\ :sub:`7-0`
       - G\ :sub:`7-0`
       - B\ :sub:`7-0`
```

Deprecated RGB Formats

Formats defined in :ref:`pixfmt-rgb-deprecated` are deprecated and must not be used by new drivers. They are documented here for reference. The meaning of their alpha bits `(a)` is ill-defined and they are interpreted as in either the corresponding ARGB or XRGB format, depending on the driver.

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\userspace-api\media\v4l\[linux-master][Documentation][userspace-api][media][v4l]pixfmt-rgb.rst`, line 769); *backlink*

Unknown interpreted text role "ref".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\userspace-api\media\v4l\[linux-master][Documentation][userspace-api][media][v4l]pixfmt-rgb.rst`, line 780)

Unknown directive type "tabularcolumns".

```
.. tabularcolumns:: |p{2.6cm}|p{0.70cm}|p{0.22cm}|p{0.22cm}|p{0.22cm}|p{0.22cm}|p{0.22cm}|p{0.22cm}|p{0.22cm}|p{0.22cm}|p{0.22cm}|
```

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\linux-master\Documentation\userspace-api\media\v4l\[linux-master][Documentation][userspace-api][media][v4l]pixfmt-rgb.rst`, line 784)

Unknown directive type "flat-table".

```
.. flat-table:: Deprecated Packed RGB Image Formats
    :header-rows:  2
    :stub-columns: 0

    * - Identifier
      - Code
      - :cspan:`7` Byte 0 in memory

      - :cspan:`7` Byte 1

      - :cspan:`7` Byte 2

      - :cspan:`7` Byte 3
    * -
      -
      - 7
      - 6
      - 5
      - 4
      - 3
      - 2
      - 1
      - 0

      - 7
      - 6
      - 5
      - 4
      - 3
      - 2
      - 1
      - 0

      - 7
      - 6
      - 5
      - 4
      - 3
      - 2
      - 1
      - 0

      - 7
      - 6
      - 5
      - 4
      - 3
      - 2
      - 1
      - 0
    * .. _V4L2-PIX-FMT-RGB444:

      - ``V4L2_PIX_FMT_RGB444``
      - 'R444'

      - g\ :sub:`3`
      - g\ :sub:`2`
      - g\ :sub:`1`
      - g\ :sub:`0`
      - b\ :sub:`3`
      - b\ :sub:`2`
      - b\ :sub:`1`
      - b\ :sub:`0`

      - a\ :sub:`3`
      - a\ :sub:`2`
      - a\ :sub:`1`
      - a\ :sub:`0`
      - r\ :sub:`3`
      - r\ :sub:`2`
      - r\ :sub:`1`
      - r\ :sub:`0`
      -
    * .. _V4L2-PIX-FMT-RGB555:

      - ``V4L2_PIX_FMT_RGB555``
      - 'RGBO'

      - g\ :sub:`2`
      - g\ :sub:`1`
      - g\ :sub:`0`
      - b\ :sub:`4`
      - b\ :sub:`3`
      - b\ :sub:`2`
      - b\ :sub:`1`
      - b\ :sub:`0`

      - a
      - r\ :sub:`4`
      - r\ :sub:`3`
      - r\ :sub:`2`
      - r\ :sub:`1`
      - r\ :sub:`0`
      - g\ :sub:`4`
      - g\ :sub:`3`
      -
    * .. _V4L2-PIX-FMT-RGB555X:

      - ``V4L2_PIX_FMT_RGB555X``
      - 'RGBQ'

      - a
      - r\ :sub:`4`
      - r\ :sub:`3`
      - r\ :sub:`2`
      - r\ :sub:`1`
      - r\ :sub:`0`
      - g\ :sub:`4`
      - g\ :sub:`3`

      - g\ :sub:`2`
      - g\ :sub:`1`
      - g\ :sub:`0`
      - b\ :sub:`4`
      - b\ :sub:`3`
      - b\ :sub:`2`
      - b\ :sub:`1`
      - b\ :sub:`0`
      -
    * .. _V4L2-PIX-FMT-BGR32:

      - ``V4L2_PIX_FMT_BGR32``
      - 'BGR4'

      - b\ :sub:`7`
      - b\ :sub:`6`
      - b\ :sub:`5`
      - b\ :sub:`4`
      - b\ :sub:`3`
      - b\ :sub:`2`
      - b\ :sub:`1`
      - b\ :sub:`0`

      - g\ :sub:`7`
      - g\ :sub:`6`
      - g\ :sub:`5`
      - g\ :sub:`4`
      - g\ :sub:`3`
      - g\ :sub:`2`
      - g\ :sub:`1`
      - g\ :sub:`0`

      - r\ :sub:`7`
      - r\ :sub:`6`
      - r\ :sub:`5`
      - r\ :sub:`4`
      - r\ :sub:`3`
      - r\ :sub:`2`
      - r\ :sub:`1`
      - r\ :sub:`0`

      - a\ :sub:`7`
      - a\ :sub:`6`
      - a\ :sub:`5`
      - a\ :sub:`4`
      - a\ :sub:`3`
      - a\ :sub:`2`
      - a\ :sub:`1`
      - a\ :sub:`0`
    * .. _V4L2-PIX-FMT-RGB32:

      - ``V4L2_PIX_FMT_RGB32``
      - 'RGB4'

      - a\ :sub:`7`
      - a\ :sub:`6`
      - a\ :sub:`5`
      - a\ :sub:`4`
      - a\ :sub:`3`
      - a\ :sub:`2`
      - a\ :sub:`1`
      - a\ :sub:`0`

      - r\ :sub:`7`
      - r\ :sub:`6`
      - r\ :sub:`5`
      - r\ :sub:`4`
      - r\ :sub:`3`
      - r\ :sub:`2`
      - r\ :sub:`1`
      - r\ :sub:`0`

      - g\ :sub:`7`
      - g\ :sub:`6`
      - g\ :sub:`5`
      - g\ :sub:`4`
      - g\ :sub:`3`
      - g\ :sub:`2`
      - g\ :sub:`1`
      - g\ :sub:`0`

      - b\ :sub:`7`
      - b\ :sub:`6`
      - b\ :sub:`5`
      - b\ :sub:`4`
      - b\ :sub:`3`
      - b\ :sub:`2`
      - b\ :sub:`1`
      - b\ :sub:`0`
```

A test utility to determine which RGB formats a driver actually supports is available from the LinuxTV v4l-dvb repository. See https://linuxtv.org/repo/ for access instructions.