

VciFrameBuffer

1) Functional Description

This VCI target can be used to display RVB or YUV images. It contains a buffer whose size is defined as the product [number_of_lines * number_of_pixel_per_line * number_of_bytes_per_pixel]. It does not contain any memory mapped registers, and supports both read & write accesses (including bursts).

2) Component definition & usage

[source:trunk/soclib/soclib/module/connectivity_component/vci_framebuffer/caba/metadata/vci_framebuffer.sd?](#)

```
Uses( 'vci_framebuffer' )
```

3) CABA Implementation

CABA sources

- interface :
[source:trunk/soclib/soclib/module/connectivity_component/vci_framebuffer/caba/source/include/vci_framebuffer.h?](#)
- implementation :
[source:trunk/soclib/soclib/module/connectivity_component/vci_framebuffer/caba/source/src/vci_framebuffer.cpp?](#)

CABA Constructor parameters

```
VciFrameBuffer(  
    sc_module_name name,    // Instance name  
    const soclib::common::IntTab &index,    // Target index  
    const soclib::common::MappingTable &mt,    // Mapping Table  
    unsigned long width,    // number of pixel per line  
    unsigned long heighth,    // number of lines  
    int subsampling);    // optional argument : default value 420 corresponds to YUV420
```

CABA Ports

- **p_resetrn** : Global system reset
- **p_clk** : Global system clock
- **p_vci** : The VCI target port

4) TLM-DT implementation

Not available yet.

TLM-DT sources

TLM-DT Constructor parameters

TLM-DT Ports