downsampling

1) Functional Description

The downsampling operation consists in reducing the number of input (i.e. decreasing the sampling rate in our case). In our system, the downsampling factor is 4. This means that we reduce the number of input symbols by 4. The architecture of the downsampling component is presented in the figure 1. It is composed of a downsampling core and a MWMR wrapper. The wrapper is used to interface the core and the MWMR controller avalaible here VciMwmrController.

2) Component definition & usage

Component definition

- source:trunk/soclib/soclib/module/ofdm_chain_components/downsampling/caba/metadata/downsampling.sd?

Usage

Downsampling has a *fifo_depth* parameter, which defines the fifo depth for the input. For example with a FIFO depth equal to 16 :

```
Uses('Downsampling', fifo_depth = 16);
```

3) CABA Implementation

CABA sources

- interface : source:trunk/soclib/soclib/module/ofdm_chain_components/downsampling/caba/source/include/downsampling.h?
- implementation : source:trunk/soclib/soclib/module/ofdm_chain_components/downsampling/caba/source/src/downsampling.cpp?

CABA Constructor parameters

```
Downsampling(
     sc_module_name name,                    // Instance name
     int ncycles)  // Number of computation cycles
```

CABA Ports

- sc_in<bool> **p_resetn** : hardware reset
- sc_in<bool> **p_clk** : clock
- soclib::caba::FifoOutput<uint32_t> **p_to_ctrl** : interface from the downsampling to the MWMR controller
- soclib::caba::FifoInput<uint32_t> **p_from_ctrl** : interface from the MWMR controller to the downsampling

4) TLM-T Implementation

The TLM-T implementation is not yet available.