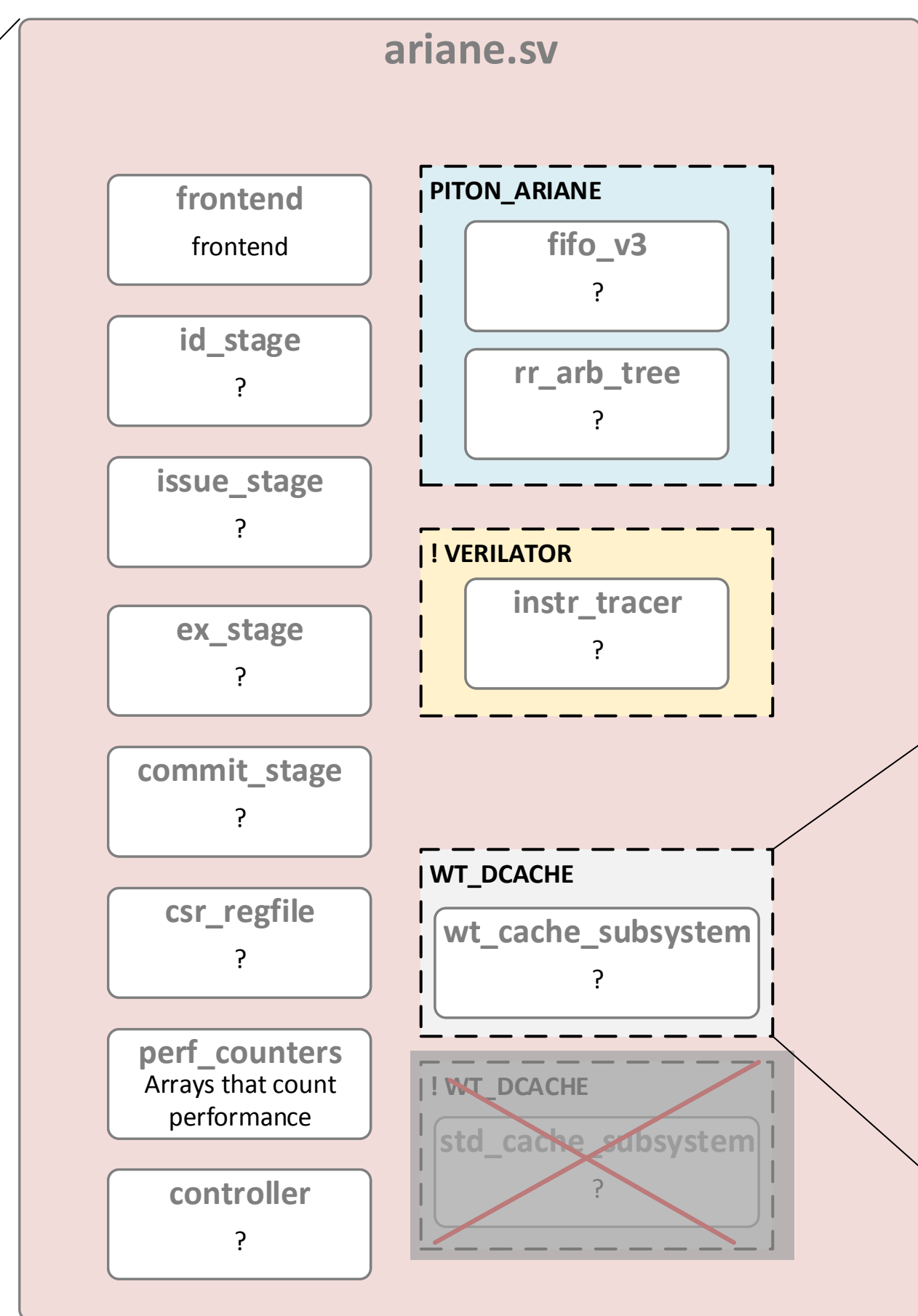
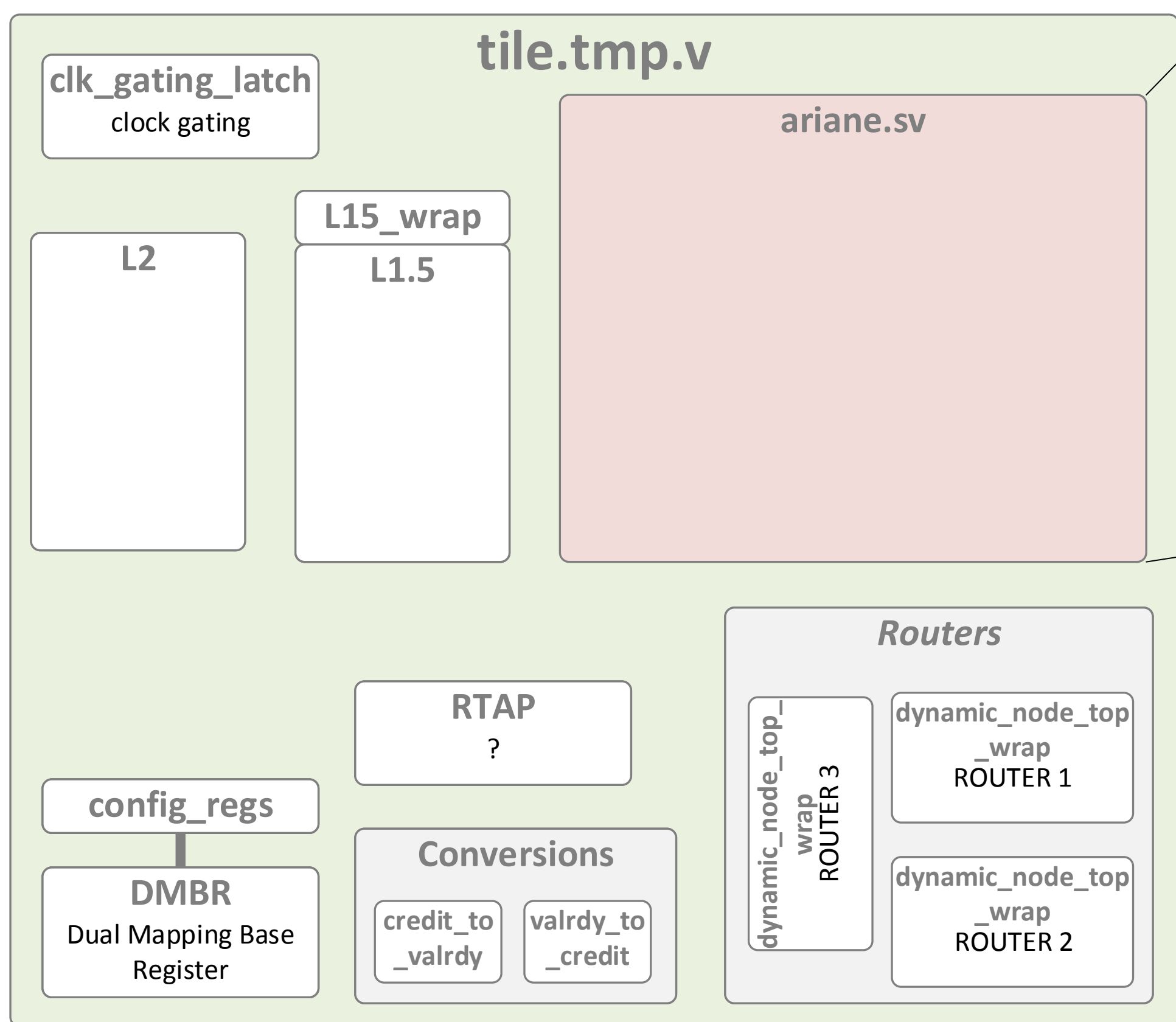
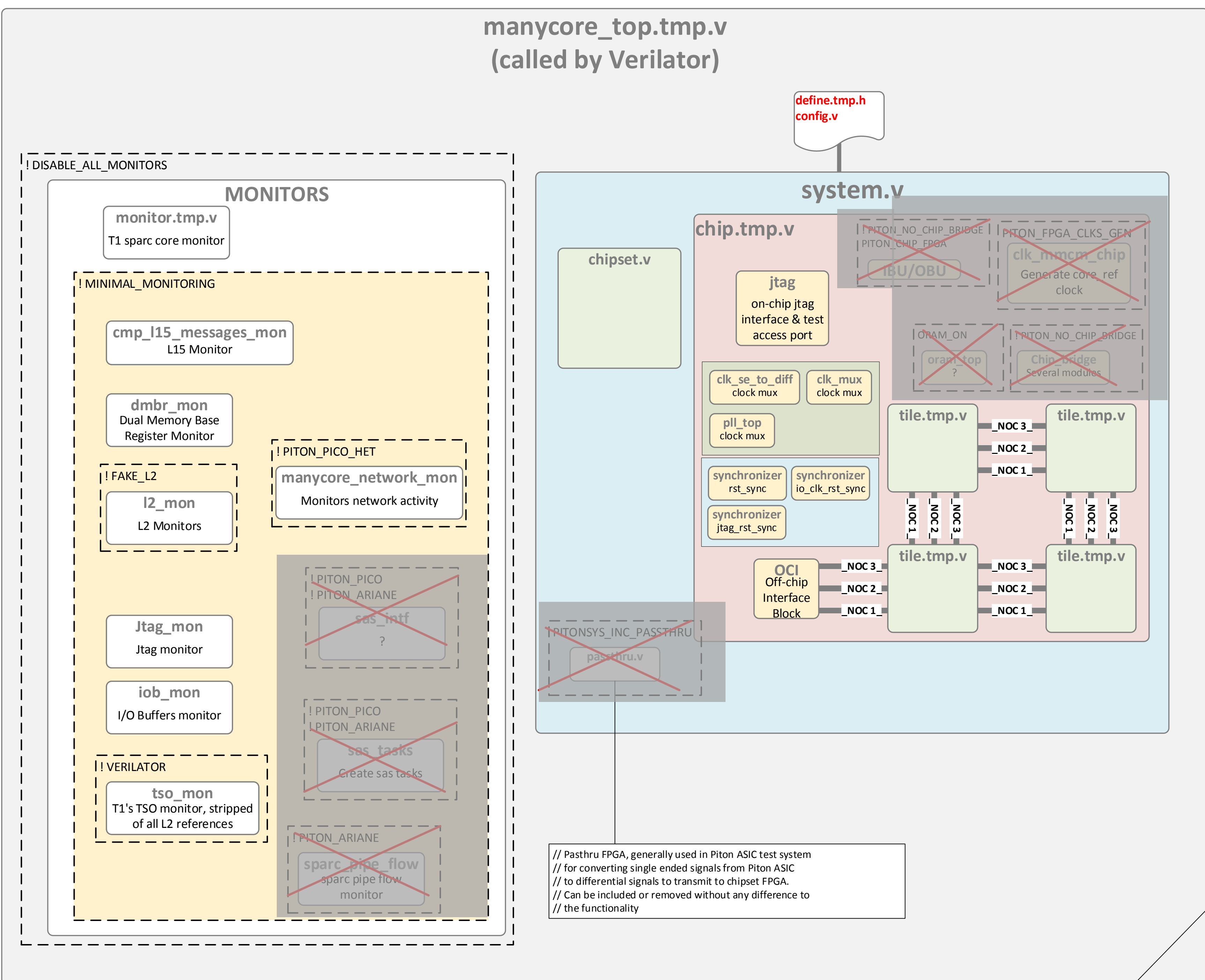
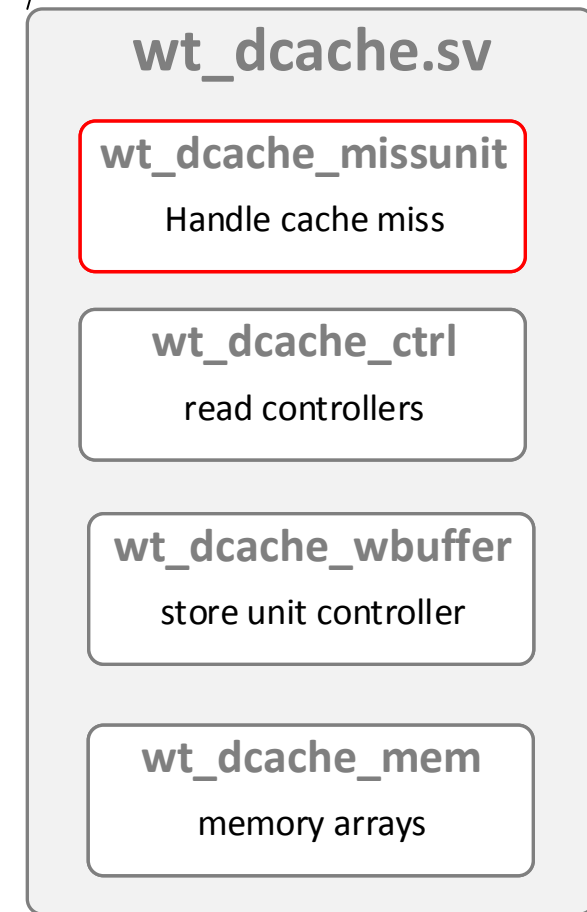
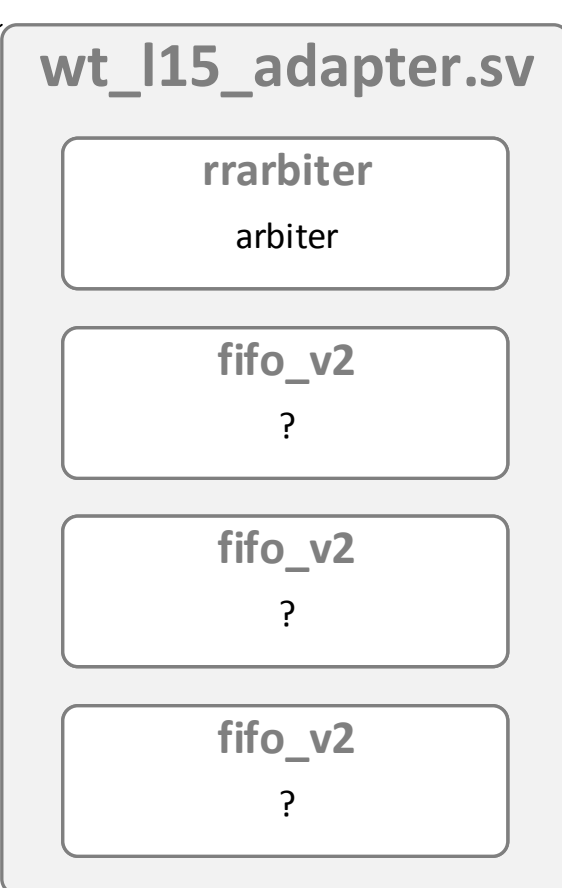
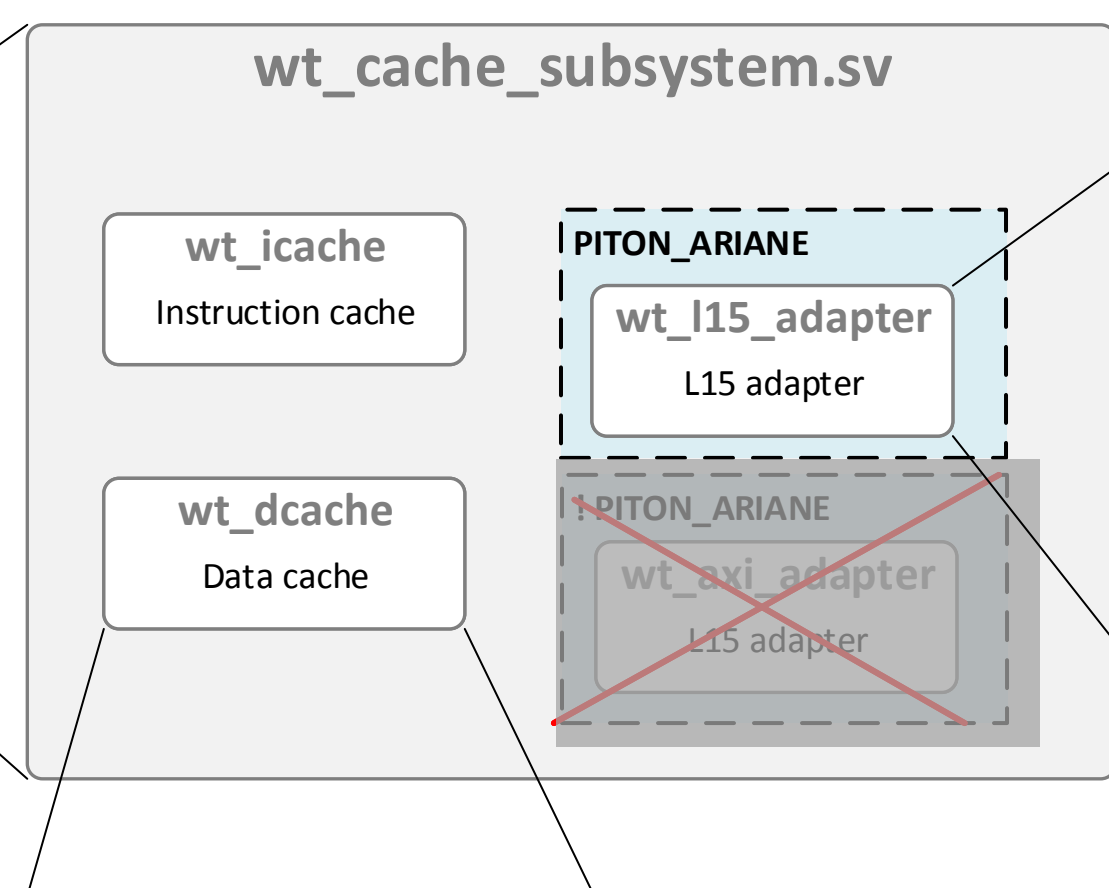


```
'timescale 1ps/1ps
'define NO_SCAN
'define PITON_NO_JTAG
'define PITON_NO_CHIP_BRIDGE
'define PITON_CLKS_CHIPSET
'define PITON_ARIANE
'define USE_FAKE_PLI_AND_CLKMUX
'define USE_FAKE_IOS
'define WT_DCACHE
'define NO_MRA_VAL
'define NO_SLAM_RANDOM
'define RTL_ARIANED
'define RTL_ARIANE1
'define NO_SLAM_RANDOM
'define NO_MRA_VAL
```

```
// Macros used in this file:
// PITON_FPGA_SYNTH      set to remove any RTL that is ASIC specific,
//                        such as clock gating, latches, etc. This also
//                        ties off any ASIC IP control signals that
//                        are not used in FPGA implementations
// PITON_NO_CHIP_BRIDGE   This indicates no chip bridge should be used on
//                        off chip link. The 3 NMs are exposed as credit
//                        based interfaces directly. This is mainly used for FPGA
//                        where there are no pin constraints. Cannot be used with
//                        PITONSYS_INC_PASSTHRU. Note that if PITON_NO_CHIP_BRIDGE
//                        is set, io_clk is not really used.
// PITON_NO_ITAG          set to remove ITAG support from Piton chip.
//                        Usually used for FPGA implementations, as ITAG
//                        is not needed.
// PITON_CLKS_CHIPSSET    indicates Piton clocks are to be generated
//                        by the chipsset. Requires
//                        PITON_CHIPSSET_CLKS_GEN
// PITON_CLKS_PASSTHRU    indicates Piton clocks are to be generated by
//                        the passthru FPGA. Requires PITONSYS_INC_PASSTHRU
//                        and PITON_PASSTHRU_CLKS_GEN
// PITON_CLKS_SIM         Piton clocks should be driven by simulated
//                        clocks from simulation testbench (input to
//                        this module). This is set by default
// PITON_CHIPSSET_DIFF_CLK Some chipsets use single ended clocks and some
//                        use differential clocks as input
// PITON_CHIPSSET_CLKS_GEN If this is set, the chipsset generates it own
//                        internal clocks. Otherwise, clocks are
//                        simulated and are inputs to this module
// PITON_FPGA_RST_ACT_HIGH This indicates we need to invert input reset signal
// VCT07_BOARD_GENESYS2_BOARD Used to indicate which board this code is being synthesized for
//
// There are more than just these
// PITONSYS_INC_PASSTHRU Set to this to include the passthrough FPGA
//                        (spartan) for real chip Piton testing. Note
//                        this macro is not compatible with
//                        PITON_NO_CHIP_BRIDGE, as it does not make
//                        sense to have the passthru FPGA if there is no
//                        chip bridge. The design will have compile
//                        errors if both are specified
// PITON_PASSTHRU_CLKS_GEN Set to have the passthrough generate its own
//                        internal clocks. Otherwise they are simulated
// PITONSYS_NO_MMC         If set, no memory controller is used. This is used
//                        in the testing of the Pico system, where a small test
//                        can be run on the chip with DRAM
//                        emulated in BRAMS
// PITON_FPGA_MMC_DDR3     Set to indicate an FPGA implementation will
//                        use a DDR3/3 memory controller. If
//                        this is not set, a default "fake"
//                        simulated DRAM is used.
```



OpenPiton is big endian



```
'timescale 1ps/1ps
'define NO_SCAN
'define PITON_NO_JTAG
'define PITON_NO_CHIP_BRIDGE
'define PITON_CLKS_CHIPSET
'define PITON_ARIANE
'define USE_FAKE_PLL_AND_CLKMUX
'define USE_FAKE_IOS
'define WT_DCACHE
'define NO_MRA_VAL
'define NO_SLAM_RANDOM
'define RTL_ARIANE0
'define RTL_ARIANE1
'define NO_SLAM_RANDOM
'define NO_MRA_VAL
```

