

# Complete Hierarchical Module Tree: hkspi\_tb.v

## Overview

This document provides a complete, level-by-level breakdown of all includes and instantiations starting from `hkspi_tb.v`, showing the entire design hierarchy in a clear, structured format.

## LEVEL 0: hkspi\_tb.v (Testbench Root)

**File Location:** Day4/vsdRiscvScl180/dv/hkspi/hkspi\_tb.v

```
hkspi_tb.v
|
└── includes:
    ├── __uprj_netlists.v
    ├── caravel_netlists.v
    ├── spiflash.v
    └── tbuart.v
|
└── instantiates:
    ├── vsdcaravel (instance: uut)
    ├── spiflash (instance: spiflash)
    └── tbuart (instance: tbuart)
```

## Testbench Functions:

- Clock generation (12.5 ns period = 40 MHz)
- Power sequencing (power1, power2)
- Reset sequence control (RSTB)
- SPI stimulus generation (SCK, CSB, SDI)
- VCD waveform capture

## **LEVEL 1A: Included File - \_\_uprj\_netlists.v**

**File Location:** Day4/vsdRiscvScl180/rtl/\_\_uprj\_netlists.v

```
__uprj_netlists.v
|
|   └── includes:
|       └── defines.v
|           └── __user_project_wrapper.v (conditional: RTL or GL version)
|               └── (GL version): gl/__user_project_wrapper.v
|
└── defines modules:
    └── user_project_wrapper (via include)
```

## **LEVEL 1B: Included File - caravel\_netlists.v**

**File Location:** Day4/vsdRiscvScl180/rtl/caravel\_netlists.v

```
caravel_netlists.v
|
|   └── includes:
|       ├── SECTION 1: Definitions & Pads
|       |   ├── defines.v
|       |   ├── userDefines.v
|       |   └── pads.v
|       |
|       └── SECTION 2: Core Components
|           ├── digital_pll.v
|           ├── caravel_clocking.v
|           ├── user_id_programming.v
|           ├── chip_io.v
|           ├── housekeeping.v
|           ├── mpnj_logic_high.v
|           ├── mpnj2_logic_high.v
|           ├── mgmt_protect.v
|           ├── constant_block.v
|           ├── gpio_control_block.v
|           ├── gpio_defaults_block.v
|           ├── gpio_logic_high.v
|           ├── xres_buf.v
|           └── spare_logic_block.v
|           |
|           └── SECTION 3: Management Core
|               ├── mgmt_core_wrapper.v
|               └── __user_project_wrapper.v
|               |
|               └── SECTION 4: Technology Libraries (SCL180)
|                   ├── pc3b03ed_wrapper.v
|                   ├── pc3d21.v
|                   ├── pc3d01_wrapper.v
|                   └── pt3b02_wrapper.v
|                   |
|                   └── SECTION 5: PLL & Clocking (duplicates)
|                       ├── digital_pll.v
|                       ├── digital_pll_controller.v
|                       ├── ring_osc2x13.v
|                       ├── caravel_clocking.v
|                       ├── user_id_programming.v
|                       ├── clock_div.v
|                       ├── mpnj_io.v
|                       └── chip_io.v
```

```
|  
|  
|   └── SECTION 6: Housekeeping & Management (duplicates)  
|       ├── housekeeping_spi.v  
|       ├── housekeeping.v  
|       ├── mprj_logic_high.v  
|       ├── mprj2_logic_high.v  
|       ├── mgmt_protect.v  
|       └── mgmt_protect_hv.v  
|  
└── defines modules: (50+ modules via includes)  
    ├── digital_pll  
    ├── caravel_clocking  
    ├── chip_io  
    ├── housekeeping  
    ├── mgmt_protect  
    ├── mgmt_core_wrapper  
    ├── user_project_wrapper  
    ├── gpio_control_block  
    ├── gpio_defaults_block  
    └── [many more...]
```

## LEVEL 1C: Included File - spiflash.v

**File Location:** Day4/vsdRiscvScl180/dv/spiflash.v

```
spiflash.v
|
├── includes: (none)
|
├── defines modules:
│   └── spiflash (behavioral SPI flash memory model)
|
└── functionality:
    ├── Reads firmware from: "hkspi.hex"
    ├── Ports:
    │   ├── csb (chip select, active low)
    │   ├── clk (SPI clock)
    │   ├── io0 (MOSI - bidirectional)
    │   ├── io1 (MISO - bidirectional)
    │   ├── io2 (Write Protect - bidirectional)
    │   └── io3 (Hold - bidirectional)
    └── Behavioral logic (no sub-modules)
```

## LEVEL 1D: Included File - tbuart.v

**File Location:** Day4/vsdRiscvScl180/dv/tbuart.v

```
tbuart.v
|
├── includes: (none)
|
├── defines modules:
│   └── tbuart (UART monitor utility)
|
└── functionality:
    ├── Port: ser_rx (UART receive input)
    ├── Monitors UART serial output
    └── Behavioral logic (no sub-modules)
```

# **LEVEL 1E: Instantiated Module - vsdcaravel**

**File Location:** Day4/vsdRiscvScl180/rtl/vsdcaravel.v

```
vsdcaravel (instance: uut)
|
└── includes:
    ├── copyright_block.v
    ├── caravel_logo.v
    ├── caravel_motto.v
    ├── open_source.v
    ├── user_id_textblock.v
    └── caravel_core.v
|
└── instantiates:
    └── INSTANCE 1: chip_io (instance: padframe)
        |
        └── connects:
            └── Package Pins Side:
                ├── Power pins (18 total):
                │   ├── vddio_pad, vddio_pad2 (3.3V I/O power)
                │   ├── vssio_pad, vssio_pad2 (I/O ground)
                │   ├── vccd_pad (1.8V digital power)
                │   ├── vssd_pad (digital ground)
                │   ├── vdda_pad (analog power)
                │   ├── vssa_pad (analog ground)
                │   ├── vdda1_pad, vdda1_pad2 (user area 1 analog)
                │   ├── vdda2_pad (user area 2 analog)
                │   ├── vssa1_pad, vssa1_pad2 (user area 1 ground)
                │   ├── vssa2_pad (user area 2 ground)
                │   ├── vccd1_pad, vccd2_pad (user area digital power)
                │   └── vssd1_pad, vssd2_pad (user area digital ground)
                |
                └── I/O Pads:
                    ├── clock (external clock input)
                    ├── resetb (external reset, active low)
                    ├── gpio (dedicated GPIO pad)
                    ├── mprj_io[37:0] (38 user I/O pads)
                    ├── flash_csb (SPI flash chip select)
                    ├── flash_clk (SPI flash clock)
                    ├── flash_io0 (SPI flash data 0)
                    └── flash_io1 (SPI flash data 1)
                |
                └── Core Side:
                    ├── Power distribution (18 signals)
                    ├── Clock output: clock_core
                    └── Reset outputs: porb_h, por_l, resetb_core_h
```

```
|   └── GPIO signals (6 signals)
|   └── Flash SPI signals (12 signals)
|   └── User I/O configuration (342+ signals)
|       └── Analog I/O access (28 signals)
|
|   └── INSTANCE 2: caravel_core (instance: chip_core)
|       |
|       └── [See LEVEL 2 for detailed breakdown]
|
└── INSTANCE 3: copyright_block (graphics only)
└── INSTANCE 4: caravel_logo (graphics only)
└── INSTANCE 5: caravel_motto (graphics only)
└── INSTANCE 6: open_source (graphics only)
└── INSTANCE 7: user_id_textblock (graphics only)
```

## LEVEL 2: chip\_io Module (Pad Frame)

**File Location:** Day4/vsdRiscvScl180/rtl/chip\_io.v

**Lines:** 1220

```
chip_io (instance: padframe)
|
|--- includes: (none - pure RTL module)
|
|--- instantiates:
|   |
|   |--- INSTANCE 1-7: constant_block[6:0] (7 instances)
|   |   |
|   |   |--- connects:
|   |   |   |
|   |   |   |--- Purpose: Generate constant logic 1 and 0 in 1.8V domain
|   |   |   |--- vccd <- internal 1.8V supply
|   |   |   |--- vssd <- internal ground
|   |   |   |--- one[6:0] → vccd_const_one (constant 1 values)
|   |   |   |--- zero[6:0] → vssd_const_zero (constant 0 values)
|
|   |--- INSTANCE 8: pc3d01_wrapper (instance: clock_pad)
|   |   |
|   |   |--- connects:
|   |   |   |
|   |   |   |--- Purpose: Input pad for system clock
|   |   |   |--- Type: 3.3V → 1.8V level shifter
|   |   |   |--- PAD <- clock (external pin)
|   |   |   |--- IN → clock_core (to caravel_core)
|   |   |   |--- VPWR <- vddio (3.3V)
|   |   |   |--- VGND <- vssio
|   |   |   |--- LVPWR <- vccd (1.8V)
|   |   |   |--- LVGND <- vssd
|
|   |--- INSTANCE 9: pc3b03ed_wrapper (instance: gpio_pad)
|   |   |
|   |   |--- connects:
|   |   |   |
|   |   |   |--- Purpose: Bidirectional GPIO pad
|   |   |   |--- Type: Configurable I/O with drive mode control
|   |   |   |--- PAD ↔ gpio (external pin)
|   |   |   |--- IN ← gpio_out_core (output driver)
|   |   |   |--- OUT → gpio_in_core (input receiver)
|   |   |   |--- INPUT_DIS <- gpio_inenb_core
|   |   |   |--- OUT_EN_N <- gpio_outenb_core
|   |   |   |--- dm[2:0] <- {gpio_mode1, gpio_mode1, gpio_mode0}
|   |   |   |--- VPWR <- vddio
|   |   |   |--- VGND <- vssio
|   |   |   |--- LVPWR <- vccd
|   |   |   |--- LVGND <- vssd
```

```
    └─ INSTANCE 10: pc3b03ed_wrapper (instance: flash_io0_pad)
        |
        └─ connects:
            └─ Purpose: SPI Flash data line 0 (MOSI)
            └─ PAD ↔ flash_io0 (external pin)
            └─ IN ← flash_io0_do_core
            └─ OUT → flash_io0_di_core
            └─ INPUT_DIS ← flash_io0_ieb_core
            └─ OUT_EN_N ← flash_io0_oeb_core
            └─ dm[2:0] ← flash_io0_mode
            └─ [power pins same as above]

    └─ INSTANCE 11: pc3b03ed_wrapper (instance: flash_io1_pad)
        |
        └─ connects:
            └─ Purpose: SPI Flash data line 1 (MISO)
            └─ PAD ↔ flash_io1 (external pin)
            └─ IN ← flash_io1_do_core
            └─ OUT → flash_io1_di_core
            └─ INPUT_DIS ← flash_io1_ieb_core
            └─ OUT_EN_N ← flash_io1_oeb_core
            └─ dm[2:0] ← flash_io1_mode
            └─ [power pins same as above]

    └─ INSTANCE 12: pt3b02_wrapper (instance: flash_csb_pad)
        |
        └─ connects:
            └─ Purpose: SPI Flash chip select (output only)
            └─ Type: Output pad with enable control
            └─ PAD → flash_csb (external pin)
            └─ IN ← flash_csb_core
            └─ OE_N ← flash_csb_oeb_core
            └─ [power pins same as above]

    └─ INSTANCE 13: pt3b02_wrapper (instance: flash_clk_pad)
        |
        └─ connects:
            └─ Purpose: SPI Flash clock (output only)
            └─ PAD → flash_clk (external pin)
            └─ IN ← flash_clk_core
            └─ OE_N ← flash_clk_oeb_core
            └─ [power pins same as above]
```

```
|── INSTANCE 14: pc3d21 (instance: resetb_pad)
|   |
|   └── connects:
|       ├── Purpose: Reset input (3.3V domain)
|       ├── Type: Input pad with level conversion
|       ├── PAD ← resetb (external pin)
|       ├── CIN → resetb_core_h (to xres_buf in caravel_core)
|       ├── VPWR ← vddio
|       ├── VGND ← vssio
|       ├── LVPWR ← vccd
|       └── LVGND ← vssd
|
|── INSTANCE 15: mprj_io (instance: mprj_pads)
|   |
|   └── connects:
|       ├── Purpose: 38-pin user I/O pad array
|       ├── Type: Fully configurable GPIO with analog access
|       |
|       ├── Power (12 connections):
|           ├── vddio ← vddio
|           ├── vssio ← vssio
|           ├── vccd ← vccd
|           ├── vssd ← vssd
|           ├── vdda1 ← vdda1
|           ├── vdda2 ← vdda2
|           ├── vssa1 ← vssa1
|           ├── vssa2 ← vssa2
|           ├── vddio_q ← vddio_q
|           ├── vssio_q ← vssio_q
|           ├── analog_a ← analog_a
|           └── analog_b ← analog_b
|       |
|       ├── Control (76 connections):
|           ├── porb_h ← porb_h
|           ├── enh[37:0] ← all driven by porb_h
|           └── vccd_conb[37:0] ← mprj_io_one (constant 1)
|       |
|       ├── I/O Pads (38 bidirectional):
|           └── io[37:0] ↔ mprj_io (external pins)
|       |
|       ├── Data Signals (114 connections):
|           ├── io_out[37:0] ← mprj_io_out (from core)
|           ├── oeb[37:0] ← mprj_io_oeb (output enable bar)
```

```

    |   └─ io_in[37:0] → mprj_io_in (to core)
    |
    └─ Input Buffer Config (114 connections):
        ├─ inp_dis[37:0] ← mprj_io_inp_dis
        ├─ ib_mode_sel[37:0] ← mprj_io_ib_mode_sel
        └─ vtrip_sel[37:0] ← mprj_io_vtrip_sel
    |
    └─ Slew/Hold Config (76 connections):
        ├─ slow_sel[37:0] ← mprj_io_slow_sel
        └─ holdover[37:0] ← mprj_io_holdover
    |
    └─ Analog Config (114 connections):
        ├─ analog_en[37:0] ← mprj_io_analog_en
        ├─ analog_sel[37:0] ← mprj_io_analog_sel
        └─ analog_pol[37:0] ← mprj_io_analog_pol
    |
    └─ Drive Mode (114 connections):
        └─ dm[113:0] ← mprj_io_dm (3 bits × 38 pads)
    |
    └─ Analog Access (28 connections):
        └─ analog_io[27:0] ↔ mprj_analog_io

```

## LEVEL 2: caravel\_core Module (Core Logic)

**File Location:** Day4/vsdRiscvScl180/rtl/caravel\_core.v

**Lines:** 1444

```
caravel_core (instance: chip_core)
|
|--- includes: (none - uses parent includes)
|
|--- instantiates:
|
|   --- INSTANCE 1: mgmt_core_wrapper (instance: soc)
|   |
|   |--- connects (70+ ports):
|   |
|   |   --- Clock & Reset:
|   |   |--- core_clk ← caravel_clk
|   |   |--- core_rstn ← caravel_rstn
|   |
|   |   --- GPIO (6 signals):
|   |   |--- gpio_out_pad ← gpio_out_core
|   |   |--- gpio_in_pad → gpio_in_core
|   |   |--- gpio_mode0_pad ← gpio_mode0_core
|   |   |--- gpio_mode1_pad ← gpio_mode1_core
|   |   |--- gpio_outenb_pad ← gpio_outenb_core
|   |   |--- gpio_inenb_pad ← gpio_inenb_core
|   |
|   |   --- Flash SPI (14 signals):
|   |   |--- flash_csb ← flash_csb_core
|   |   |--- flash_clk ← flash_clk_core
|   |   |--- flash_io0_oeb ← flash_io0_oeb_core
|   |   |--- flash_io0_di → flash_io0_di_core
|   |   |--- flash_io0_do ← flash_io0_do_core
|   |   |--- flash_io1_oeb ← flash_io1_oeb_core
|   |   |--- flash_io1_di → flash_io1_di_core
|   |   |--- flash_io1_do ← flash_io1_do_core
|   |   |--- flash_io2_oeb ← flash_io2_oeb_core
|   |   |--- flash_io2_di → flash_io2_di_core
|   |   |--- flash_io2_do ← flash_io2_do_core
|   |   |--- flash_io3_oeb ← flash_io3_oeb_core
|   |   |--- flash_io3_di → flash_io3_di_core
|   |   |--- flash_io3_do ← flash_io3_do_core
|   |
|   |   --- Wishbone Bus to User Project (10 signals):
|   |   |--- mprj_wb_iena (enable)
|   |   |--- mprj_cyc_o (cycle)
|   |   |--- mprj_stb_o (strobe)
|   |   |--- mprj_we_o (write enable)
```

```
|   |   |   mprj_sel_o[3:0] (byte select)
|   |   |   mprj_adr_o[31:0] (address)
|   |   |   mprj_dat_o[31:0] (data out)
|   |   |   mprj_ack_i (acknowledge)
|   |   |   mprj_dat_i[31:0] (data in)

|   |
|   |   └─ Housekeeping Wishbone (4 signals):
|   |       |   hk_stb_o
|   |       |   hk_cyc_o
|   |       |   hk_dat_i[31:0]
|   |       |   hk_ack_i

|   |
|   |   └─ Interrupts (9 signals):
|   |       |   irq[5:0]
|   |       |   user_irq_ena[2:0]

|   |
|   |   └─ Module Control (4 signals):
|   |       |   qspi_enabled
|   |       |   uart_enabled
|   |       |   spi_enabled
|   |       |   debug_mode

|   |
|   |   └─ Serial Interfaces (7 signals):
|   |       |   ser_tx (UART TX)
|   |       |   ser_rx (UART RX)
|   |       |   spi_sdi
|   |       |   spi_csb
|   |       |   spi_sck
|   |       |   spi_sdo
|   |       |   spi_sdoenb

|   |
|   |   └─ Debug Interface (3 signals):
|   |       |   debug_in
|   |       |   debug_out
|   |       |   debug_oeb

|   |
|   |   └─ Logic Analyzer (384 signals):
|   |       |   la_input[127:0]
|   |       |   la_output[127:0]
|   |       |   la_oenb[127:0]
|   |       |   la_iena[127:0]

|   |
|   |   └─ Status:
```

```
    └─ trap

 INSTANCE 2: mgmt_protect (instance: mgmt_buffers)
    |
    └─ connects (48+ ports):
        |
        └─ Clock & Reset Tri-state Buffers:
            |
            └─ caravel_clk ← caravel_clk
            └─ caravel_clk2 ← caravel_clk2
            └─ caravel_rstn ← caravel_rstn
            └─ user_clock → mprj_clock
            └─ user_clock2 → mprj_clock2
            └─ user_reset → mprj_reset

        |
        └─ Wishbone Bus Buffers (Management → User):
            |
            └─ Input from Management:
                |
                └─ mprj_cyc_o_core
                └─ mprj_stb_o_core
                └─ mprj_we_o_core
                └─ mprj_sel_o_core[3:0]
                └─ mprj_adr_o_core[31:0]
                └─ mprj_dat_o_core[31:0]

            |
            └─ Output to User:
                |
                └─ mprj_cyc_o_user
                └─ mprj_stb_o_user
                └─ mprj_we_o_user
                └─ mprj_sel_o_user[3:0]
                └─ mprj_adr_o_user[31:0]
                └─ mprj_dat_o_user[31:0]

            |
            └─ Return from User:
                |
                └─ mprj_ack_i_user
                └─ mprj_dat_i_user[31:0]

        |
        └─ Control:
            └─ mprj_iena_wb

    └─ Interrupt Buffers (6 signals):
        |
        └─ user_irq_core[2:0]
        └─ user_irq_ena[2:0]
        └─ user_irq[2:0]
```

```
|   └ Logic Analyzer Buffers (512 signals):
|       └ la_data_out_core[127:0]
|       └ la_data_out_mprj[127:0]
|       └ la_data_in_core[127:0]
|       └ la_data_in_mprj[127:0]
|       └ la_oenb_mprj[127:0]
|       └ la_oenb_core[127:0]
|       └ la_iena_mprj[127:0]
|
|   └ Power Good Monitoring (4 signals):
|       └ user1_vcc_powergood
|       └ user2_vcc_powergood
|       └ user1_vdd_powergood
|       └ user2_vdd_powergood
|
└ INSTANCE 3: user_project_wrapper (instance: mprj)
|
|   └ connects (12+ port groups):
|
|       └ Wishbone Slave Interface (8 signals):
|           └ wbs_cyc_i ← mprj_cyc_o_user
|           └ wbs_stb_i ← mprj_stb_o_user
|           └ wbs_we_i ← mprj_we_o_user
|           └ wbs_sel_i[3:0] ← mprj_sel_o_user
|           └ wbs_addr_i[31:0] ← mprj_addr_o_user
|           └ wbs_dat_i[31:0] ← mprj_dat_o_user
|           └ wbs_ack_o → mprj_ack_i_user
|           └ wbs_dat_o[31:0] → mprj_dat_i_user
|
|       └ GPIO Pads (114 signals):
|           └ io_in[37:0] ← user_io_in
|           └ io_out[37:0] → user_io_out
|           └ io_oeb[37:0] → user_io_oeb
|
|       └ Analog I/O (28 signals):
|           └ analog_io[27:0] ← mprj_analog_io
|
|       └ Logic Analyzer (384 signals):
|           └ la_data_in[127:0] ← la_data_in_user
|           └ la_data_out[127:0] → la_data_out_user
|           └ la_oenb[127:0] → la_oenb_user
|
|       └ Clock & Reset (3 signals):
```

```
|   |   └─ wb_clk_i <- mprj_clock
|   |   └─ wb_rst_i <- mprj_reset
|   |   └─ user_clock2 <- mprj_clock2
|
|   └─ Interrupts (3 signals):
|       └─ user_irq[2:0] <- user_irq_core
|
└─ INSTANCE 4: caravel_clocking (instance: clock_ctrl)
    |
    └─ connects (13 ports):
        |
        └─ Inputs:
            |
            └─ porb <- porb_1
            └─ ext_clk_sel <- ext_clk_sel
|
|   └─ ext_clk <- clock_core
|   └─ pll_clk <- pll_clk
|   └─ pll_clk90 <- pll_clk90
|   └─ resetb <- rstb_1
|   └─ sel[2:0] <- spi_pll_sel
|   └─ sel2[2:0] <- spi_pll90_sel
|   └─ ext_reset <- ext_reset
|
|   └─ Outputs:
|       └─ core_clk → caravel_clk
|       └─ user_clk → caravel_clk2
|       └─ resetb_sync → caravel_rstn
|
└─ INSTANCE 5: digital_pll (instance: pll)
    |
    └─ connects (8 ports):
        |
        └─ Inputs:
            |
            └─ resetb <- rstb_1
            └─ enable <- spi_pll_ena
            └─ osc <- clock_core
            └─ div[4:0] <- spi_pll_div
|
|   └─ Control:
|       └─ dco <- spi_pll_dco_ena
|       └─ ext_trim[25:0] <- spi_pll_trim
|
|   └─ Outputs:
```

```
    └─ clockp[1:0]:
        ├─ [0] → pll_clk (0° phase)
        └─ [1] → pll_clk90 (90° phase)

    └─ INSTANCES 6-43: gpio_defaults_block[37:0] (38 instances)
        |
        └─ connects (per instance):
            ├─ Purpose: Default GPIO configuration storage
            ├─ Parameter: GPIO_CONFIG_INIT (13 bits each)
            └─ gpio_defaults[493:0] (output, 38×13 bits total)

    └─ INSTANCES 44-45: gpio_control_block (bidir_1[1:0])
        |
        └─ connects:
            ├─ Purpose: Bidirectional GPIO control (JTAG, SDO)
            ├─ Pads: mprj_io[1:0]
            ├─ Management control signals
            └─ Serial configuration chain

    └─ INSTANCES 46-51: gpio_control_block (in_1a[5:0])
        |
        └─ connects:
            ├─ Purpose: Input-only GPIO (section 1a)
            ├─ Pads: mprj_io[7:2]
            └─ Default input behavior

    └─ INSTANCES 52-70: gpio_control_block (in_1[18:0])
        |
        └─ connects:
            ├─ Purpose: Input-only GPIO (section 1)
            ├─ Pads: mprj_io[26:8]
            └─ Default input behavior

    └─ INSTANCES 71-73: gpio_control_block (bidir_2[2:0])
        |
        └─ connects:
            ├─ Purpose: Bidirectional GPIO (Flash I02/3, SPI SDO)
            ├─ Pads: mprj_io[37:35]
            └─ Management-controlled

    └─ INSTANCES 74-108: gpio_control_block (in_2[34:0])
        |
        └─ connects:
```

- └─ Purpose: Input-only GPIO (section 2)
- └─ Pads: mprj\_io[34:27] (partial, note overlap in doc)
- └─ Default input behavior

  

- └─ INSTANCE 109: housekeeping (instance: housekeeping)
  - |
  - └─ connects (50+ ports):
    - |
    - └─ Wishbone Bus Interface (10 signals):
      - |─ wb\_clk\_i ← caravel\_clk
      - |─ wb\_rstn\_i ← caravel\_rstn
      - |─ wb\_adr\_i[31:0] ← mprj\_adr\_o\_core
      - |─ wb\_dat\_i[31:0] ← mprj\_dat\_o\_core
      - |─ wb\_sel\_i[3:0] ← mprj\_sel\_o\_core
      - |─ wb\_we\_i ← mprj\_we\_o\_core
      - |─ wb\_cyc\_i ← hk\_cyc\_o
      - |─ wb\_stb\_i ← hk\_stb\_o
      - |─ wb\_ack\_o → hk\_ack\_i
      - └─ wb\_dat\_o[31:0] → hk\_dat\_i
    - |
    - └─ Power-on-Reset:
      - └─ porb ← porb\_l
    - |
    - └─ PLL Configuration (40 signals):
      - |─ pll\_ena → spi\_pll\_ena
      - |─ pll\_dco\_ena → spi\_pll\_dco\_ena
      - |─ pll\_div[4:0] → spi\_pll\_div
      - |─ pll\_sel[2:0] → spi\_pll\_sel
      - |─ pll90\_sel[2:0] → spi\_pll90\_sel
      - |─ pll\_trim[25:0] → spi\_pll\_trim
      - └─ pll\_bypass → ext\_clk\_sel
    - |
    - └─ Module Enable Flags (4 signals):
      - |─ qspi\_enabled
      - |─ uart\_enabled
      - |─ spi\_enabled
      - └─ debug\_mode
    - |
    - └─ UART Interface (2 signals):
      - |─ ser\_tx → mprj\_io[6]
      - └─ ser\_rx ← mprj\_io[5]
    - |
    - └─ SPI Master Interface (5 signals):

```
|   |   └── spi_sdi ← mprj_io[2]
|   |   └── spi_csb ← mprj_io[3]
|   |   └── spi_sck ← mprj_io[4]
|   |   └── spi_sdo → mprj_io[1]
|   └── spi_sdoenb

|   └── Debug Interface (3 signals):
|       ├── debug_in ← mprj_io[0]
|       ├── debug_out → mprj_io[0]
|       └── debug_oeb

|   └── GPIO Serial Configuration (4 signals):
|       ├── serial_clock → mprj_io_loader_clock
|       ├── serial_load → mprj_io_loader_strobe
|       ├── serial_resetn → mprj_io_loader_resetn
|       ├── serial_data_1 → mprj_io_loader_data_1
|       └── serial_data_2 → mprj_io_loader_data_2

|   └── GPIO Pad Configuration (114 signals):
|       ├── mgmt_gpio_in[37:0] ← mgmt_io_in_hk
|       ├── mgmt_gpio_out[37:0] → mgmt_io_out_hk
|       └── mgmt_gpio_oeb[37:0] → mgmt_io_oeb_hk

|   └── Flash SPI to Core (28 signals):
|       ├── spimemio_flash_csb ← flash_csb_core
|       ├── spimemio_flash_clk ← flash_clk_core
|       ├── spimemio_flash_io0_oeb ← flash_io0_oeb_core
|       ├── spimemio_flash_io1_oeb ← flash_io1_oeb_core
|       ├── spimemio_flash_io2_oeb ← flash_io2_oeb_core
|       ├── spimemio_flash_io3_oeb ← flash_io3_oeb_core
|       ├── spimemio_flash_io0_do ← flash_io0_do_core
|       ├── spimemio_flash_io1_do ← flash_io1_do_core
|       ├── spimemio_flash_io2_do ← flash_io2_do_core
|       ├── spimemio_flash_io3_do ← flash_io3_do_core
|       ├── spimemio_flash_io0_di → flash_io0_di_core
|       ├── spimemio_flash_io1_di → flash_io1_di_core
|       ├── spimemio_flash_io2_di → flash_io2_di_core
|       └── spimemio_flash_io3_di → flash_io3_di_core

|   └── Flash SPI to Padframe (12 signals):
|       ├── pad_flash_csb → flash_csb_frame
|       ├── pad_flash_csb_oeb → flash_csb_oeb
|       └── pad_flash_clk → flash_clk_frame
```

```
|   |   |   pad_flash_clk_oeb → flash_clk_oeb
|   |   |   pad_flash_io0_oeb → flash_io0_oeb
|   |   |   pad_flash_io1_oeb → flash_io1_oeb
|   |   |   pad_flash_io0_ieb → flash_io0_ieb
|   |   |   pad_flash_io1_ieb → flash_io1_ieb
|   |   |   pad_flash_io0_do → flash_io0_do
|   |   |   pad_flash_io1_do → flash_io1_do
|   |   |   pad_flash_io0_di ← flash_io0_di
|   |   |   pad_flash_io1_di ← flash_io1_di

|   |   | SRAM Read-Only Interface (4 signals):
|   |   |   sram_ro_clk ← hkspi_sram_clk
|   |   |   sram_ro_csb ← hkspi_sram_csb
|   |   |   sram_ro_addr[7:0] ← hkspi_sram_addr
|   |   |   sram_ro_data[31:0] → hkspi_sram_data

|   |   | Power Monitoring (4 signals):
|   |   |   usr1_vcc_powergood ← mprj_vcc_pwrgood
|   |   |   usr2_vcc_powergood ← mprj2_vcc_pwrgood
|   |   |   usr1_vdd_powergood ← mprj_vdd_pwrgood
|   |   |   usr2_vdd_powergood ← mprj2_vdd_pwrgood

|   |   | Status & Control (39 signals):
|   |   |   irq[5:0] → to mgmt_core_wrapper
|   |   |   reset → ext_reset
|   |   |   mask_rev_in[31:0] ← from user_id_programming
|   |   |   trap ← trap

|   | INSTANCE 110: user_id_programming (instance: user_id_value)
|   |
|   |   connects:
|   |       | Purpose: User ID storage (read-only)
|   |       | mask_rev[31:0] → mask_rev_in

|   | INSTANCE 111: dummy_por (instance: por)
|   |
|   |   connects:
|   |       | Purpose: Power-on-reset generation
|   |       | Inputs:
|   |           |   vdd3v3 ← vddio
|   |           |   vdd1v8 ← vccd
|   |           |   vss3v3 ← vssio
|   |           |   vss1v8 ← vssd
```

```
|   |
|   └── Outputs:
|       ├── porb_h → porb_h (high domain)
|       ├── porb_l → porb_l (low domain)
|       └── por_l → por_l
|
|   └── INSTANCE 112: xres_buf (instance: rstb_level)
|   |
|   └── connects:
|       ├── Purpose: Reset level converter (3.3V → 1.8V)
|       ├── A ← rstb_h (3.3V input)
|       ├── VPWR ← vddio (3.3V)
|       ├── VGND ← vssio
|       ├── LVPWR ← vccd (1.8V)
|       ├── LVGND ← vssd
|       └── X → rstb_l (1.8V output)
|
|   └── INSTANCES 113-116: spare_logic_block[3:0] (4 instances)
|   |
|   └── connects:
|       ├── Purpose: Spare logic for metal mask fixes
|       └── All tied to no-connect nets (unused)
|
└── INSTANCES 117+: Layout-only blocks
    ├── empty_macro[1:0] (2 instances)
    └── manual_power_connections (1 instance)
        └── Purpose: Layout/power routing only
```

## LEVEL 3: mgmt\_core\_wrapper Module

**File Location:** Day4/vsdRiscvScl180/rtl/mgmt\_core\_wrapper.v

```
mgmt_core_wrapper (instance: soc)
|
|--- includes:
|     mgmt_core.v
|
|--- instantiates:
|     INSTANCE 1: mgmt_core (instance: soc)
|         |
|         connects:
|             All signals pass-through from wrapper
|             [See LEVEL 4 for mgmt_core details]
```

## LEVEL 4: mgmt\_core Module (Management SoC)

**File Location:** Day4/vsdRiscvScl180/rtl/mgmt\_core.v

```
mgmt_core (instance: soc)
|
|--- includes:
|     |--- RAM256.v
|     |--- RAM128.v
|     |--- VexRiscv_MinDebugCache.v
|
|--- instantiates:
|
|     |--- INSTANCE 1: VexRiscv_MinDebugCache (instance: vexriscv_cpu)
|             |
|             |--- connects (40+ ports):
|                     |
|                     |--- Clock & Reset:
|                             |--- clk <- core_clk
|                             |--- reset <- core_rstn (inverted)
|
|                     |--- Debug Interface (6 signals):
|                             |--- debug_bus_cmd_valid
|                             |--- debug_bus_cmd_ready
|                             |--- debug_bus_cmd_payload_wr
|                             |--- debug_bus_cmd_payload_address[7:0]
|                             |--- debug_bus_cmd_payload_data[31:0]
|                             |--- debug_resetOut
|
|                     |--- Instruction Bus (iBus - 13 signals):
|                             |--- iBus_cmd_valid (request)
|                             |--- iBus_cmd_ready (acknowledge)
|                             |--- iBus_cmd_payload_pc[31:0] (program counter)
|                             |--- iBus_rsp_valid (response valid)
|                             |--- iBus_rsp_payload_error (error flag)
|                             |--- iBus_rsp_payload_inst[31:0] (instruction)
|                             |--- Cache control signals
|
|                     |--- Data Bus (dBus - 21 signals):
|                             |--- dBus_cmd_valid (request)
|                             |--- dBus_cmd_ready (acknowledge)
|                             |--- dBus_cmd_payload_wr (write enable)
|                             |--- dBus_cmd_payload_address[31:0]
|                             |--- dBus_cmd_payload_data[31:0]
|                             |--- dBus_cmd_payload_size[1:0] (transfer size)
|                             |--- dBus_rsp_ready
|                             |--- dBus_rsp_error
```

```
|   |   └─ dBus_rsp_data[31:0]
|   └─ Cache control signals
|
|   └─ Interrupts (2 signals):
|       ├─ timerInterrupt
|       └─ externalInterrupt
|
└─ INSTANCE 2: RAM256 (instance: RAM256)
    |
    └─ connects:
        ├─ Purpose: 256x32-bit instruction memory
        ├─ CLK ← core_clk
        ├─ EN0 ← chip enable (from address decoder)
        ├─ WE0 ← write enable (from iBus)
        ├─ A0[7:0] ← address (from iBus)
        ├─ Di0[31:0] ← write data (from iBus)
        └─ Do0[31:0] → read data (to iBus)
|
└─ INSTANCE 3: RAM128 (instance: RAM128)
    |
    └─ connects:
        ├─ Purpose: 128x32-bit data memory
        ├─ CLK ← core_clk
        ├─ EN0 ← chip enable (from address decoder)
        ├─ WE0 ← write enable (from dBus)
        ├─ A0[6:0] ← address (from dBus)
        ├─ Di0[31:0] ← write data (from dBus)
        └─ Do0[31:0] → read data (to dBus)
|
└─ Pure RTL Logic (no sub-modules):
    |
    └─ Housekeeping SPI Slave Interface:
        ├─ SPI state machine
        ├─ Configuration register access
        └─ Command/data processing
    |
    └─ UART Controller:
        ├─ TX/RX state machines
        ├─ Baud rate generator
        └─ FIFO buffers
    |
    └─ SPI Master Controller (Flash):
        ├─ Flash command processor
```

```
|   └── Quad SPI support
|   └── Address translation
|
|   └── GPIO Controller:
|       ├── Pin configuration
|       ├── Direction control
|       └── Data registers
|
|   └── Interrupt Controller:
|       ├── IRQ prioritization
|       ├── Interrupt enable/mask
|       └── Vector generation
|
|   └── Wishbone Bus Components:
|       ├── Bus arbiter (multi-master)
|       ├── Address decoder
|       ├── Bus multiplexer
|       └── Timeout/error handler
|
|   └── Clock & Reset Logic:
|       ├── Clock divider
|       ├── Reset synchronizer
|       └── Clock domain crossing
|
|   └── Debug Interface Handler:
|       ├── Debug command decoder
|       ├── Breakpoint logic
|       └── Register access
|
└── Control Registers:
    ├── System configuration
    ├── Status registers
    ├── Timer/counter
    └── Module enable/disable
```

## LEVEL 5: VexRiscv\_MinDebugCache (RISC-V Core)

**File Location:** Day4/vsdRiscvScl180/rtl/VexRiscv\_MinDebugCache.v

```
VexRiscv_MinDebugCache (instance: vexriscv_cpu)
|
|--- includes: (none)
|
└--- Pure RTL Implementation (no sub-module instantiations):
    |
    |--- Pipeline Stages:
    |   |
    |   |--- Stage 1: Instruction Fetch (IF)
    |   |   |
    |   |   |--- Program counter (PC) logic
    |   |   |--- Branch prediction
    |   |   |--- Instruction cache interface
    |   |   |--- Fetch queue
    |   |
    |   |--- Stage 2: Instruction Decode (ID)
    |   |   |
    |   |   |--- Instruction decoder
    |   |   |--- Register file read
    |   |   |--- Immediate generation
    |   |   |--- Control signal generation
    |   |   |--- Hazard detection
    |   |
    |   |--- Stage 3: Execute (EX)
    |   |   |
    |   |   |--- ALU (Arithmetic Logic Unit)
    |   |   |--- Branch/jump calculation
    |   |   |--- Comparison operations
    |   |   |--- Shift operations
    |   |   |--- Forwarding logic
    |   |
    |   |--- Stage 4: Memory Access (MEM)
    |   |   |
    |   |   |--- Data cache interface
    |   |   |--- Load/store unit
    |   |   |--- Memory alignment
    |   |   |--- Bus interface logic
    |   |
    |   |--- Stage 5: Writeback (WB)
    |   |   |
    |   |   |--- Register file write
    |   |   |--- Result multiplexer
    |   |   |--- Writeback enable logic
    |
    |--- Register File:
    |   |
    |   |--- 32 general-purpose registers (x0-x31)
    |   |
    |   |--- x0 hardwired to zero
    |   |
    |   |--- Dual-port read
```

- └ Single-port write
- Cache Subsystem:
  - | Instruction Cache (I-Cache):
    - | Tag array
    - | Data array
    - | Valid bits
    - └ Cache controller
  - | Data Cache (D-Cache):
    - | Tag array
    - | Data array
    - | Valid/dirty bits
    - └ Cache controller
- Debug Interface:
  - | Debug bus decoder
  - | Debug register access
  - | Single-step logic
  - | Breakpoint/watchpoint
  - └ Debug state machine
- CSR (Control & Status Registers):
  - | Machine mode CSRs
  - | Exception handling
  - | Interrupt control
  - └ Performance counters
- Exception & Interrupt Handling:
  - | Exception detection
  - | Interrupt arbitration
  - | Trap vector logic
  - └ Return from exception
- Pipeline Control:
  - | Stall logic
  - | Flush logic
  - | Hazard resolution
  - └ Pipeline registers
- Instruction Set Support:
  - | RV32I (Base Integer)

```
|── M Extension (Multiply/Divide)
|── C Extension (Compressed - optional)
└── Zicsr (CSR instructions)
```

## LEVEL 5: RAM256 & RAM128 (Memory Modules)

**File Location:** Day4/vsdRiscvScl180/rtl/RAM256.v and RAM128.v

```
RAM256 (instance: RAM256)
|
|--- includes: (none)
|
└--- Pure Memory Logic:
      |--- Size: 256 words × 32 bits = 8 KB
      |--- Address width: 8 bits [7:0]
      |--- Data width: 32 bits
      |--- Ports:
      |     |--- CLK (clock input)
      |     |--- EN0 (chip enable)
      |     |--- WE0 (write enable)
      |     |--- A0[7:0] (address)
      |     |--- Di0[31:0] (data input)
      |     |--- Do0[31:0] (data output)
      |
      |--- Behavioral Model:
      |     |--- Synchronous write (on clock edge)
      |     |--- Synchronous read (on clock edge)
      |     |--- Single-port access
```

---

```
RAM128 (instance: RAM128)
|
|--- includes: (none)
|
└--- Pure Memory Logic:
      |--- Size: 128 words × 32 bits = 4 KB
      |--- Address width: 7 bits [6:0]
      |--- Data width: 32 bits
      |--- Ports:
      |     |--- CLK (clock input)
      |     |--- EN0 (chip enable)
      |     |--- WE0 (write enable)
      |     |--- A0[6:0] (address)
      |     |--- Di0[31:0] (data input)
      |     |--- Do0[31:0] (data output)
      |
      |--- Behavioral Model:
      |     |--- Synchronous write (on clock edge)
```

- |— Synchronous read (on clock edge)
- |— Single-port access

# Complete Hierarchical Summary Tree

```
LEVEL 0: hkspi_tb.v
|
|--- includes:
|   |--- __uprj_netlists.v
|   |   |--- includes: __user_project_wrapper.v
|   |--- caravel_netlists.v
|   |   |--- includes: 50+ support files
|   |--- spiflash.v
|   |--- tuart.v
|
|--- instantiates:
|
|   |--- LEVEL 1: vsdcaravel (instance: uut)
|   |
|   |--- includes:
|   |   |--- copyright_block.v
|   |   |--- caravel_logo.v
|   |   |--- caravel_motto.v
|   |   |--- open_source.v
|   |   |--- user_id_textblock.v
|   |   |--- caravel_core.v
|
|   |--- instantiates:
|   |
|   |   |--- LEVEL 2A: chip_io (instance: padframe)
|   |
|   |   |--- instantiates:
|   |       |--- constant_block[6:0] (7 instances)
|   |       |--- pc3d01_wrapper (clock_pad)
|   |       |--- pc3b03ed_wrapper (gpio_pad)
|   |       |--- pc3b03ed_wrapper (flash_io0_pad)
|   |       |--- pc3b03ed_wrapper (flash_io1_pad)
|   |       |--- pt3b02_wrapper (flash_csb_pad)
|   |       |--- pt3b02_wrapper (flash_clk_pad)
|   |       |--- pc3d21 (resetb_pad)
|   |       |--- mprj_io (mprj_pads - 38 user I/O)
|
|   |--- LEVEL 2B: caravel_core (instance: chip_core)
|   |
|   |   |--- instantiates:
```

```
    └─ LEVEL 3A: mgmt_core_wrapper (instance: soc)
        |
        |   └─ includes: mgmt_core.v
        |
        |   └─ instantiates:
        |
        |       └─ LEVEL 4: mgmt_core (instance: soc)
        |           |
        |           └─ includes:
        |               └─ RAM256.v
        |               └─ RAM128.v
        |               └─ VexRiscv_MinDebugCache.v
        |
        |   └─ instantiates:
        |
        |       └─ LEVEL 5A: VexRiscv_MinDebugCache
        |           |
        |           (instance: vexriscv_cpu)
        |
        |           └─ Pure RTL:
        |               └─ IF stage
        |               └─ ID stage
        |               └─ EX stage
        |               └─ MEM stage
        |               └─ WB stage
        |               └─ Register file (32x32)
        |               └─ I-Cache
        |               └─ D-Cache
        |               └─ Debug interface
        |               └─ CSR registers
        |               └─ Exception handling
        |
        |   └─ LEVEL 5B: RAM256 (instance: RAM256)
        |       |
        |       └─ Behavioral Memory:
        |           └─ 256x32-bit SRAM
        |
        |   └─ LEVEL 5C: RAM128 (instance: RAM128)
        |       |
        |       └─ Behavioral Memory:
        |           └─ 128x32-bit SRAM
        |
        |   └─ Pure RTL Peripherals:
```

```
    |           └─ Housekeeping SPI
    |           └─ UART controller
    |           └─ SPI master
    |           └─ GPIO controller
    |           └─ Interrupt controller
    |           └─ Wishbone arbiter
    |           └─ Control registers

    └─ LEVEL 3B: mgmt_protect (instance: mgmt_buffers)
        └─ Pure RTL: Tri-state domain crossing buffers

    └─ LEVEL 3C: user_project_wrapper (instance: mprj)
        └─ includes: defines.v, debug_regs.v
        └─ instantiates: (user-defined logic)

    └─ LEVEL 3D: caravel_clocking (instance: clock_ctrl)
        └─ Pure RTL: Clock mux & synchronization

    └─ LEVEL 3E: digital_pll (instance: pll)
        └─ Pure RTL: PLL/DCO logic

    └─ LEVEL 3F: gpio_defaults_block[37:0] (38 instances)
        └─ Configuration storage

    └─ LEVEL 3G: gpio_control_block arrays (multiple)
        └─ GPIO pad control logic

    └─ LEVEL 3H: housekeeping (instance: housekeeping)
        └─ Pure RTL: SPI slave + config controller

    └─ LEVEL 3I: user_id_programming
        └─ User ID storage

    └─ LEVEL 3J: dummy_por (instance: por)
        └─ Power-on-reset generation

    └─ LEVEL 3K: xres_buf (instance: rstb_level)
        └─ Reset level converter

    └─ LEVEL 3L: spare_logic_block[3:0] + layout blocks
        └─ Spare gates & layout placeholders

└─ LEVEL 1: spiflash (instance: spiflash)
```

```

    └── Behavioral SPI flash memory simulator
    |
└── LEVEL 1: tbuart (instance: tbuart)
    └── UART monitor utility

```

## Module Instantiation Count Summary

Level	Module Type	Instances	Notes
0	hkspi_tb	1	Testbench root
1	vsdcaravel	1	Top-level SoC
1	spiflash	1	Behavioral model
1	tbuart	1	Monitor utility
2	chip_io	1	Pad frame
2	caravel_core	1	Core logic
2	constant_block	7	In chip_io
2	Pad wrappers	8	Various types
3	mgmt_core_wrapper	1	SoC wrapper
3	mgmt_protect	1	Domain crossing
3	user_project_wrapper	1	User logic
3	caravel_clocking	1	Clock control
3	digital_pll	1	PLL
3	gpio_defaults_block	38	GPIO config
3	gpio_control_block	65+	GPIO control (arrays)
3	housekeeping	1	SPI slave
3	Support modules	7	POR, level shift, etc.
4	mgmt_core	1	RISC-V SoC

Level	Module Type	Instances	Notes
5	VexRiscv_MinDebugCache	1	RISC-V CPU
5	RAM256	1	Instruction memory
5	RAM128	1	Data memory
<b>TOTAL</b>	<b>All modules</b>	<b>~145</b>	<b>Including arrays</b>

## Maximum Nesting Depth

Deepest path: 6 levels

```

hkspi_tb (Level 0)
└── vsdcaravel (Level 1)
    └── caravel_core (Level 2)
        └── mgmt_core_wrapper (Level 3)
            └── mgmt_core (Level 4)
                └── VexRiscv_MinDebugCache (Level 5)
                    └── RISC-V pipeline internals (Level 6)

```

## Key Signal Flow Paths

### 1. Clock Path

```

External clock pin
→ chip_io (clock_pad)
→ clock_core
→ caravel_clocking
→ {caravel_clk, caravel_clk2}
→ mgmt_core / user_project_wrapper

```

## 2. Reset Path

```
External resetb pin
→ chip_io (resetb_pad)
→ resetb_core_h
→ xres_buf (level converter)
→ rstb_l
→ caravel_clocking
→ caravel_rstn
→ mgmt_core / user_project_wrapper
```

## 3. SPI Flash Path

```
External SPI flash
↔ chip_io (flash pads)
↔ {flash_csb_core, flash_clk_core, flash_io[0:1]_core}
↔ housekeeping
↔ mgmt_core (SPI master controller)
↔ VexRiscv (instruction fetch)
```

## 4. Wishbone Bus Path

```
mgmt_core (master)
→ Wishbone signals
→ mgmt_protect (domain crossing)
→ user_project_wrapper (slave)
```

## 5. GPIO Path

```
External mprj_io pins
↔ chip_io (mprj_pads)
↔ gpio_control_blocks
↔ housekeeping / user_project_wrapper
```

# **End of Document**

This comprehensive document provides a complete, level-by-level breakdown of all includes and instantiations in the hkspi\_tb design hierarchy, showing exactly how each module connects and what it contains.