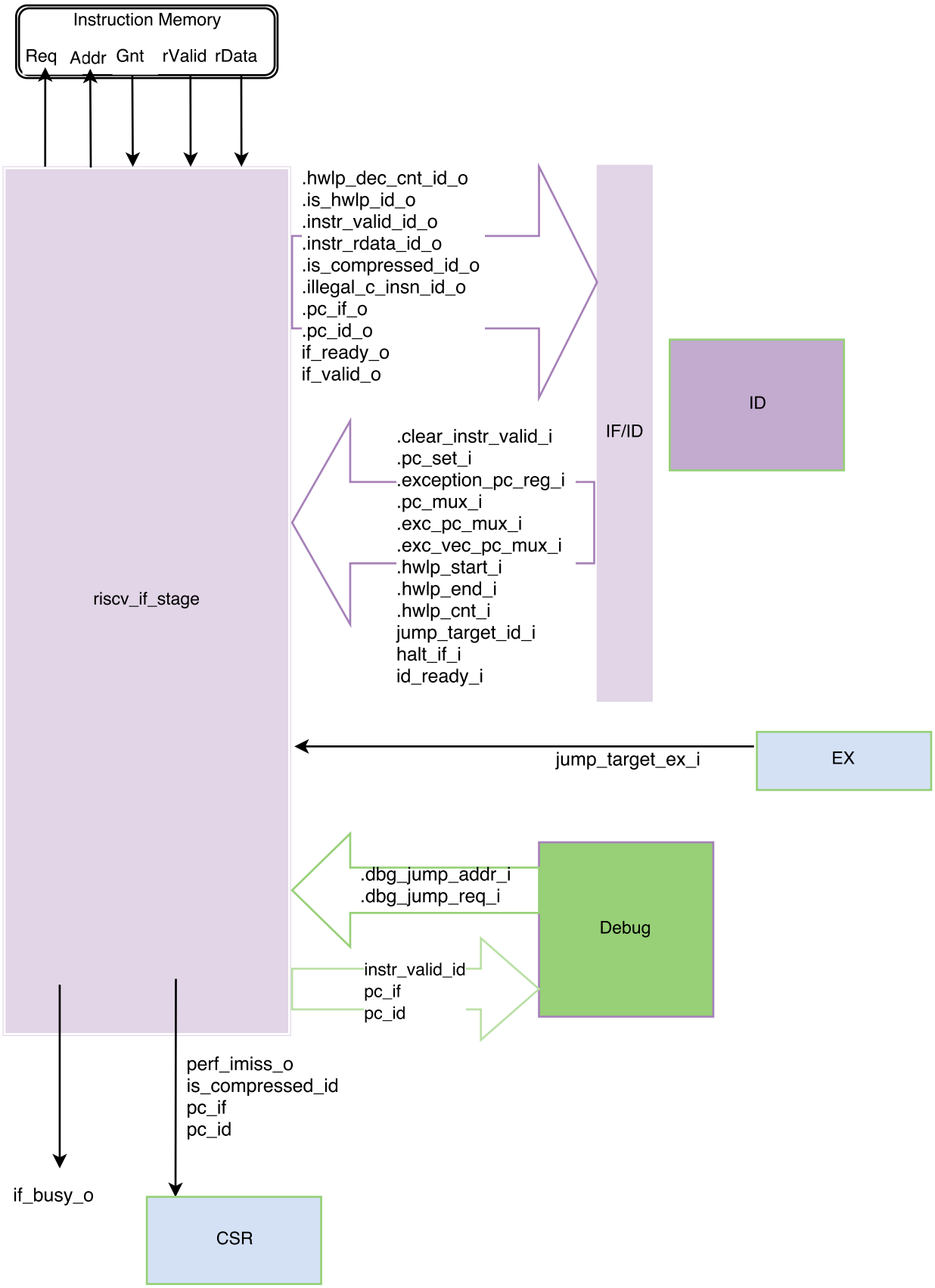
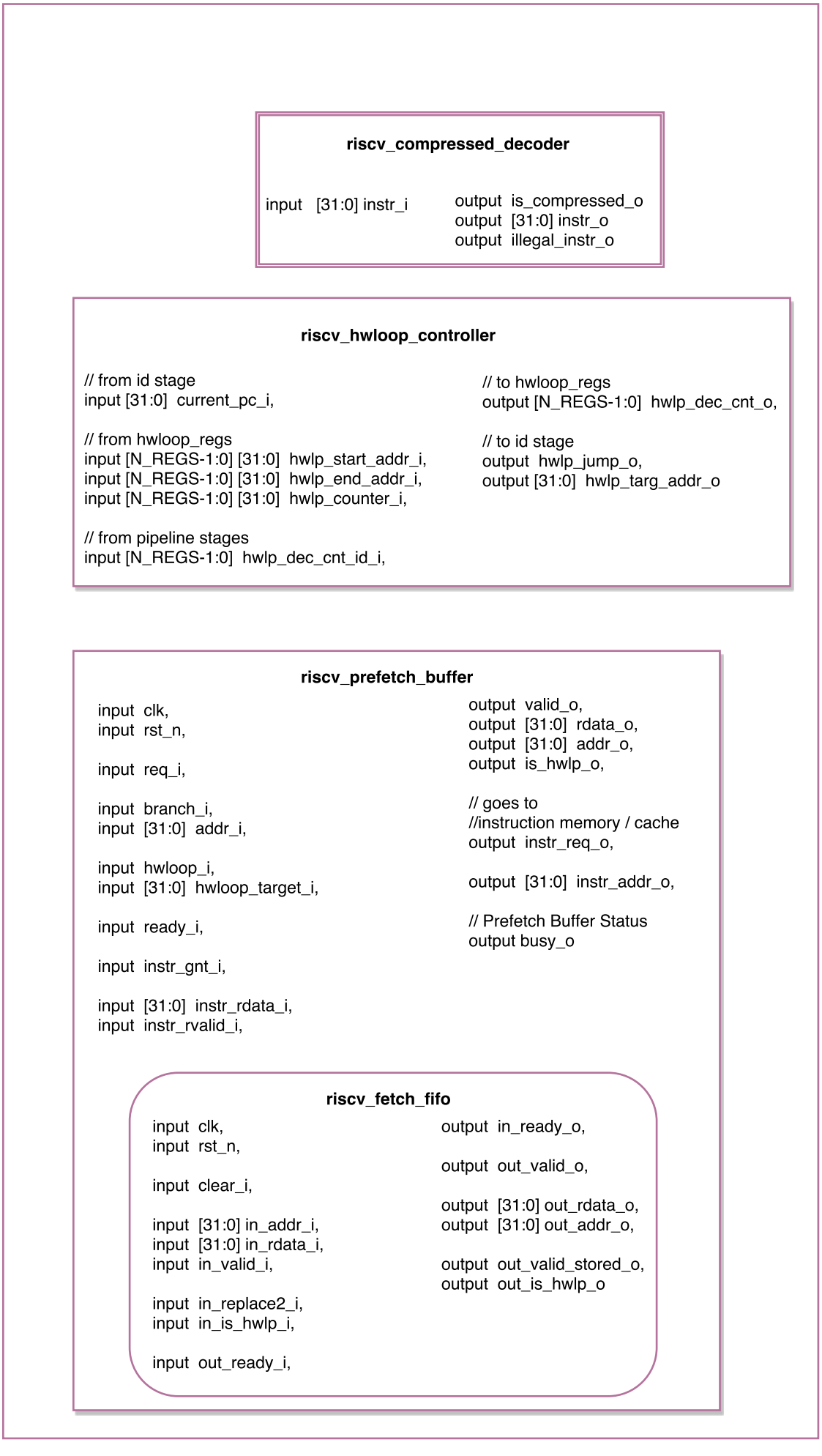

Instruction Memory
Req
Addr
Gnt
rValid
rData
riscv_if_stage
.hwlp_dec_cnt_id_o
.is_hwlp_id_o
.instr_valid_id_o
.instr_rdata_id_o
.is_compressed_id_o
.illegal_c_insn_id_o
.pc_if_o
.pc_id_o
if_ready_o
if_valid_o
IF/ID
ID
.clear_instr_valid_i
.pc_set_i
.exception_pc_reg_i
.pc_mux_i
.exc_pc_mux_i
.exc_vec_pc_mux_i
.hwlp_start_i
.hwlp_end_i
.hwlp_cnt_i
jump_target_id_i
halt_if_i
id_ready_i
jump_target_ex_i
EX
.dbg_jump_addr_i
.dbg_jump_req_i
Debug
instr_valid_id
pc_if
pc_id
perf_imiss_o
is_compressed_id
pc_if
pc_id
if_busy_o
CSR


riscv_compressed_decoder
input [31:0] instr_i
output is_compressed_o
output [31:0] instr_o
output illegal_instr_o
riscv_hwloop_controller
// from id stage
input [31:0] current_pc_i,
// from hwloop_regs
input [N_REGS-1:0] [31:0] hwlp_start_addr_i,
input [N_REGS-1:0] [31:0] hwlp_end_addr_i,
input [N_REGS-1:0] [31:0] hwlp_counter_i,
// from pipeline stages
input [N_REGS-1:0] hwlp_dec_cnt_id_i,
// to hwloop_regs
output [N_REGS-1:0] hwlp_dec_cnt_o,
// to id stage
output hwlp_jump_o,
output [31:0] hwlp_targ_addr_o
riscv_prefetch_buffer
input clk,
input rst_n,
input req_i,
input branch_i,
input [31:0] addr_i,
input hwloop_i,
input [31:0] hwloop_target_i,
input ready_i,
input instr_gnt_i,
input [31:0] instr_rdata_i,
input instr_rvalid_i,
output valid_o,
output [31:0] rdata_o,
output [31:0] addr_o,
output is_hwlp_o,
// goes to
//instruction memory / cache
output instr_req_o,
output [31:0] instr_addr_o,
// Prefetch Buffer Status
output busy_o
riscv_fetch_fifo
input clk,
input rst_n,
input clear_i,
input [31:0] in_addr_i,
input [31:0] in_rdata_i,
input in_valid_i,
input in_replace2_i,
input in_is_hwlp_i,
input out_ready_i,
output in_ready_o,
output out_valid_o,
output [31:0] out_rdata_o,
output [31:0] out_addr_o,
output out_valid_stored_o,
output out_is_hwlp_o