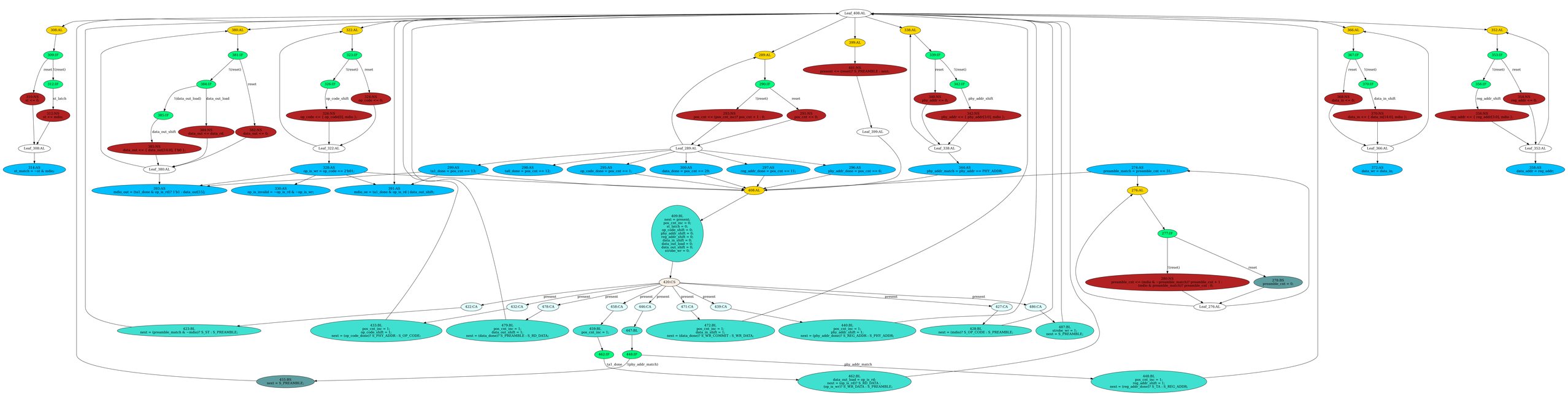
Leaf_408:AL
308:AL
309:IF
reset
!(reset)
310:NS
st <=
312:IF
st_latch
312:NS
st <= mdio;
Leaf_308:AL
114:AS
st_match = ~st & mdio;
380:AL
381:IF
!(reset)
reset
384:IF
!(data_out_load)
data_out_load
385:IF
data_out_shift
384:NS
data_out <= data_rd;
382:NS
data_out <= 0;
383:NS
data_out <= { data_out[14:0], 1'b0 };
Leaf_380:AL
322:AL
323:IF
!(reset)
reset
324:IF
op_code_shift
324:NS
op_code <= 0;
326:NS
op_code <= { op_code[0], mdio };
Leaf_322:AL
328:AS
op_is_wr = op_code == 2'b01;
359:AS
mdio_out = (ta1_done & op_is_rd)? 1'b1 : data_out[15];
330:AS
op_is_invalid = ~op_is_rd & ~op_is_wr;
291:AS
mdio_oe = ta1_done & op_is_rd | data_out_shift;
289:AL
290:IF
!(reset)
reset
293:NS
pos_cnt <= (pos_cnt_inc)? pos_cnt + 1 : 0;
291:NS
pos_cnt <= 0;
Leaf_289:AL
299:AS
ta1_done = pos_cnt == 13;
298:AS
ta0_done = pos_cnt == 12;
295:AS
op_code_done = pos_cnt == 1;
300:AS
data_done = pos_cnt == 29;
297:AS
reg_addr_done = pos_cnt == 11;
296:AS
phy_addr_done = pos_cnt == 6;
408:AL
399:AL
400:NS
present <= (reset)? S_PREAMBLE : next;
Leaf_399:AL
338:AL
339:IF
reset
!(reset)
340:NS
phy_addr <= 0;
342:IF
phy_addr_shift
342:NS
phy_addr <= { phy_addr[3:0], mdio };
Leaf_338:AL
344:AS
phy_addr_match = (phy_addr == PHY_ADDR);
366:AL
367:IF
reset
!(reset)
368:NS
data_in <= 0;
370:IF
data_in_shift
370:NS
data_in <= { data_in[14:0], mdio };
Leaf_366:AL
372:AS
data_wr = data_in;
352:AL
353:IF
!(reset)
reset
354:IF
reg_addr_shift
354:NS
reg_addr <= 0;
356:NS
reg_addr <= { reg_addr[3:0], mdio };
Leaf_352:AL
358:AS
data_addr = reg_addr;
274:AS
preamble_match = preamble_cnt == 31;
276:AL
277:IF
!(reset)
reset
280:NS
preamble_cnt <= (mdio & ~preamble_match)? preamble_cnt + 1 : (mdio & preamble_match)? preamble_cnt : 0;
278:BS
preamble_cnt = 0;
Leaf_276:AL
409:BL
next = present;
pos_cnt = 0;
ta_done = 0;
op_code_shift = 0;
phy_addr_shift = 0;
reg_addr_shift = 0;
data_in_shift = 0;
data_out_load = 0;
data_out_shift = 0;
strobe_wr = 0;
420:CS
present
422:CA
432:CA
478:CA
458:CA
466:CA
471:CA
439:CA
427:CA
486:CA
423:BL
next = (preamble_match & ~mdio)? S_ST : S_PREAMBLE;
433:BL
pos_cnt_inc = 1;
op_code_shift = 1;
next = (op_code_done)? S_PHY_ADDR : S_OP_CODE;
479:BL
pos_cnt_inc = 1;
data_out_shift = 1;
next = (data_done)? S_PREAMBLE : S_RD_DATA;
459:BL
pos_cnt_inc = 1;
447:BL
472:BL
pos_cnt_inc = 1;
data_in_shift = 1;
next = (data_done)? S_WR_COMMIT : S_WR_DATA;
440:BL
pos_cnt_inc = 1;
phy_addr_shift = 1;
next = (phy_addr_done)? S_REG_ADDR : S_PHY_ADDR;
428:BL
next = (mdio)? S_OP_CODE : S_PREAMBLE;
487:BL
strobe_wr = 1;
next = S_PREAMBLE;
462:IF
ta1_done
448:IF
!(phy_addr_match)
phy_addr_match
455:BS
next = S_PREAMBLE;
462:BL
data_out_load = op_is_rd;
next = (op_is_rd)? S_RD_DATA :
(op_is_wr)? S_WR_DATA : S_PREAMBLE;
448:BL
pos_cnt_inc = 1;
reg_addr_shift = 1;
next = (reg_addr_done)? S_TA : S_REG_ADDR;