FPGA Physical Design Rule Check (DRC) Desk Reference

Overview

Physical Design Rule Check (DRC) consists of a series of tests used to discover physical errors and some logic errors in the design. Three modules use physical DRC. They are:

- EPIC Device Editor

  You can run a DRC check with the **DRC > DRC** menu command. For more information, see the EPIC Help topic **Concepts > Physical Design Rule Check (DRC)**.

- The Generate Bitstream Data process in Project Navigator or bitgen

  A DRC check is run automatically when you generate your bitstream. For more information, see the topics in the ispLEVER Help system in **Design Flow User Guide > Device Programming > Bit Generation**.

- The drc program (see following section)

Running DRC from the Command Line

To run the DRC independently, enter the following on a command line:

drc [-s] [-e] [-nr] [-np] [-v] [-z] [-o <*outfile_name*>] <*file_name*>

-s Output a summary report only.

-e Report errors only.

-nr No routing checks.

-np No placement checks.

-v Report all warnings and errors (default).

-z Report incomplete programming as errors.

-o <outfile_name> Override the default output report file <*file_name*>.tdr with <*outfile_name*>.tdr.

<file_name> The name of the file on which DRC is to be run.

Physical DRC Error and Warning Messages

Messages are in the format:

SEVERITY - check type: Message.

Where SEVERITY stands for ERROR, WARNING, or ERRWARN; and check type indicates the type of DRC check that generated the message. The severity levels are:

- ERROR—Indicates a condition in which the routing or component logic will not operate correctly.
- WARNING—Indicates a condition where the routing or logic is incomplete, or where the condition is incorrect but not considered serious.
- ERRWARN—Certain messages may appear as either warnings or errors, depending on the application and signal connections. An application instructs DRC to treat all violations of this type as errors or warnings. for example, BITGEN treats them as errors, whereas EPIC treats them as warnings.
- CATASTROPHIC

The tables that follow are organized in this order:

- netcheck: ERROR, ERRWARN, WARNING (alphabetically within sections)
- blockcheck: ERROR, ERRWARN, WARNING (alphabetically within sections)
- macrocheck: ERRWARN (alphabetically within sections)
- macrodefcheck: ERROR, WARNING (alphabetically within sections)
- chipcheck: ERROR, ERRWARN, WARNING (alphabetically within LatticeSC section)
- duplicate name check: WARNING (alphabetically within sections)
- netsanitycheck: CATASTROPHIC (alphabetically within sections)

DRC Warnings and Errors (General)

General Message	Explanation
ERROR - netcheck: A TBUF group has both a route through and a placed component <*comp_name*>.	
ERROR - netcheck: Conflicting buffer directions on signal <*signal_name*> between comp <*comp_name*> pin <*pin_name*> and comp <*comp_name*> pin <*pin_name*>.	

General Message	Explanation
ERROR - netcheck: Conflicting pin inversions for signal <*signal_name*> found on T pins in the same tbuf group.	
ERROR - netcheck: Conflicting switch directions found on signal <*signal_name*>.	Net <*signal_name*> has been routed in a way that moves current through a programmable switch opposite to the allowed direction.
ERROR - netcheck: Conflicting switch directions on signal <*signal_name*> between comp <*comp_name*> pin <*pin_name*> and comp <*comp_name*> pin <*pin_name*>.	Net <*signal_name*> has been routed in a way that moves current between comp pins opposite to the allowed direction.
ERROR - netcheck: Differing nets <*net_name*> and <*net_name*> found on T pins in the same tbuf group.	
ERROR - netcheck: Loop found on signal <*signal_name*>.	Net <*signal_name*> net contains a routed loop.
ERROR - netcheck: More than one active source pin found on signal <*signal_name*>.	A net (not a tristate buffer net) has more than one driver.
ERROR - netcheck: Node <*node_name*> is used as a bounceback point in routing, but one of its connections is missing.	When a node is used as a bounceback point in routing, the arc from this bounceback point to the node outward should also be used in routing. This includes connections from the bounceback points to PLC, PIC or SWITCHBOX. The arc is a pinwire in case of PLC and PIC, a jumpwire in case of SWITCHBOX.
ERROR - netcheck: Not all source pins on signal <*signal_name*> are type TRI.	Net <*signal_name*> contains both tristate buffer sources and non-tristate buffer sources.
ERROR - netcheck: Signal <*signal_name*> connected to an external pin of comp <*comp_name*> must have more than one load or source pin on it.	With EPIC in macro mode, you have defined an external pin that is not connected to any other signal.
ERROR - netcheck: Unidirectional buffer or route through between 2 TBUF drivers found on signal <*signal_name*> between comp <*comp_name*> pin <*pin_name*> and comp <*comp_name*> pin <*pin_name*>.	
ERRWARN - netcheck: Design is completely unrouted, <*number*> warnings not reported below.	The design is not routed. The physical DRC has not reported each component with a separate warning message.
ERRWARN - netcheck: Signal <*signal_name*> is unrouted.	Net <*signal_name*> is completely unrouted.
WARNING - netcheck: Antenna found on signal <*signal_name*>.	Net <*signal_name*> has an antenna (a route that is connected to only one pin or a net segment that does not serve to connect two pins).
WARNING - netcheck: Duplicate signal name <*signal_name*> found.	Two nets in the database are both named <*signal_name*>.
WARNING - netcheck: Empty signal name found.	The database contains a net with no name.

General Message	Explanation
WARNING - netcheck: Floating net segments found for signal <*signal_name*>.	Net <*signal_name*> has a floating segment (a route which is not connected to a pin at either end).
WARNING - netcheck: No load pins found on signal <*signal_name*>.	Net <*signal_name*> does not have any load pins.
WARNING - netcheck: No pins found on signal <*signal_name*>.	Net <*signal_name*> does not have any pins.
WARNING - netcheck: No source pins found on signal <*signal_name*>.	Net <*signal_name*> does not have any source pins.
WARNING - netcheck: Signal <*signal_name*> is partially routed.	Net <*signal_name*> is only partially routed.
WARNING - netcheck: <*number*> unrouted net warnings not reported.	The physical DRC has not reported each unrouted net with a separate warning message.
ERROR - blockcheck: Attempting to use unbonded PIC <*pic_name*> as bonded.	
ERROR - blockcheck: Found <*number*> tbufs in a group instead of 4!	
ERROR - blockcheck: PFU <*pfu_name*> is using an active low write enable on WENA with pre-initialized RAM. This is not supported in the hardware.	
ERROR - blockcheck: PFU <*pfu_name*> is using an active low write enable on WENB with pre-initialized RAM. This is not supported in the hardware.	
ERROR - blockcheck: PFU <*pfu_name*> LUT <*lut_name*> has an invalid equation.	
ERROR - blockcheck: PFU <*pfu_name*> must have the LSRMUX programmed when using the SET_RST_LOGIC_BLOCK in FE_SELECT mode.	
ERROR - blockcheck: PFU <*pfu_name*> pin <*pin_name*> on LATCH/FF <*name*> is not connected.	
ERROR - blockcheck: PFU <*pfu_name*> pin <*pin_name*> on primitive <*name*> is not connected.	
ERROR - blockcheck: PFU <*pfu_name*> pin <*pin_name*> on primitive <*name*> is not used.	
ERROR - blockcheck: PFU <*pfu_name*> programming of QLUT0 and QLUT1 must both be set to LUT, RAM, or ROM.	
ERROR - blockcheck: PFU <*pfu_name*> programming of QLUT2 and QLUT3 must both be set to LUT, RAM or ROM.	

General Message	Explanation
ERROR - blockcheck: The signal on pin <*pin_name*> on component <*comp_name*> is routed using a direct-in (DIN) wire. This is illegal while the comp is in FE_SELECT mode.	
ERROR - blockcheck: The signals connected to unused TRI <*name*> have no effect.	
ERRWARN - blockcheck: Component <*comp_name*> is not placed.	
ERRWARN - blockcheck: PIC <*pic_name*> pin D is enabled but has no signal.	
ERRWARN - blockcheck: PIC <*pic_name*> pin O is enabled but has no signal.	
ERRWARN - blockcheck: PIC <*pic_name*> pin T is enabled but has no signal.	
ERRWARN - blockcheck: PFU <*pfu_name*> pin A4 needs a signal when in RIPPLE ADDSUB/ UPDN mode.	
ERRWARN - blockcheck: PFU <*pfu_name*> pin <*pin_name*> on RAM <*name*> has no signal. A WE and a WD signal are required.	
ERRWARN - blockcheck: TRI <*name*> is used but has no signal on pin I.	
ERRWARN - blockcheck: TRI <*name*> is used but has no signal on pin T.	
WARNING - blockcheck: Component <*comp_name*> has input signals but no output signals.	
WARNING - blockcheck: Design is completely unrouted, <*number*> warnings not reported below.	The design is not routed. The physical DRC has not reported each component with a separate warning message.
WARNING - blockcheck: Duplicate component name <*comp_name*> found.	
WARNING - blockcheck: Empty component name found for site <*site_name*>.	A component is placed in site <*site_name*>, but the database does not have a name for the component.
WARNING - blockcheck: No signals are connected to component <*comp_name*>.	
WARNING - blockcheck: PFU <*pfu_name*> active pin <*pin_name*> on MUX <*mux_name*> is not used.	
WARNING - blockcheck: PFU <*pfu_name*> has the mux <*mux_name*> enabled but the register <*name*> is not enabled.	

General Message	Explanation
WARNING - blockcheck: PFU <*pfu_name*> has the register <*name*> enabled but the mux <*mux_name*> is not enabled.	
WARNING - blockcheck: PFU <*pfu_name*> input pin <*pin_name*> on ROM/RAM <*rom_name*> has no signal.	
WARNING - blockcheck: PFU <*pfu_name*> must have an input specified on LSRMUX when in SYNC mode.	
WARNING - blockcheck: PFU <*pfu_name*> must have a signal on pin <*pin_name*> when using FFLATCH <*name*> in FE_SELECT mode.	
WARNING - blockcheck: PFU <*pfu_name*> must have a signal on pin LSR when using the SET_RST_LOGIC_BLOCK in FE_SELECT mode.	
WARNING - blockcheck: PFU <*pfu_name*> non-active pin <*pin_name*> on MUX <*mux_name*> is used.	
WARNING - blockcheck: PFU <*pfu_name*> output of used LUT <*lut_name*> drives nothing.	
WARNING - blockcheck: PFU <*pfu_name*> output of used ROM/RAM <*rom_name*> drives nothing.	
WARNING - blockcheck: PFU <*pfu_name*> pin <*pin_name*> on LATCH/FF <*name*> is not connected.	
WARNING - blockcheck: PFU <*pfu_name*> pin <*pin_name*> on LUT <*lut_name*> has a signal but its term is not used.	
WARNING - blockcheck: PFU <*pfu_name*> pin <*pin_name*> on LUT <*lut_name*> has no effect in non-RAM mode.	
WARNING - blockcheck: PFU <*pfu_name*> pin <*pin_name*> on primitive <*name*> is not connected.	
WARNING - blockcheck: PFU <*pfu_name*> pin <*pin_name*> on primitive <*name*> is not used.	
WARNING - blockcheck: PFU <*pfu_name*> pin <*pin_name*> on ROM <*rom_name*> has no effect.	
WARNING - blockcheck: PFU <*pfu_name*> RIPPLE_LOGIC_BLOCK output pins drive nothing.	

General Message	Explanation
WARNING - blockcheck: PFU <*pfu_name*> signal on pin A4 when not in RIPPLE ADDSUB/UPDN mode has no effect.	
WARNING - blockcheck: PFU <*pfu_name*> term <*term_name*> in LUT <*lut_name*> is used but its pin has no signal.	
WARNING - blockcheck: PFU <*pfu_name*> use of the internal signal F1 at the output of the XOR-MUX when in F4A-F5B mode is dangerous.	
WARNING - blockcheck: PFU <*pfu_name*> use of the internal signal F2 at the output of the 3-input AND gate when in F5A-F4B mode is not recommended.	
WARNING - blockcheck: PIC <*pic_name*> is not programmed.	
WARNING - blockcheck: PIC <*pic_name*> has the OD mux enabled but the tristate mux is not enabled.	
WARNING - blockcheck: PIC <*pic_name*> has the tristate control set high which negates the effect of the pullup.	
WARNING - blockcheck: PIC <*pic_name*> has the tristate control set low effectively masking the input on the O or D pins.	
WARNING - blockcheck: PIC <*pic_name*> has the tristate mux enabled but the OD mux is not enabled.	
WARNING - blockcheck: PIC <*pic_name*> pin D is not enabled but has a signal.	
WARNING - blockcheck: PIC <*pic_name*> pin O is not enabled but has a signal.	
WARNING - blockcheck: PIC <*pic_name*> pin T is not enabled but has a signal.	
WARNING - blockcheck: Pin <*pin_name*> on component <*comp_name*> has a signal but has no effect in counter mode.	
WARNING - blockcheck: Pin <*pin_name*> on component <*comp_name*> has a signal but is unconnected inside the comp.	
WARNING - blockcheck: Pin <*pin_name*> on component <*comp_name*> is programmed but has no signal connected to it.	

General Message	Explanation
WARNING - blockcheck: Pin A4 on component <*comp_name*> is programmed but requires a signal connected to port A0.	
WARNING - blockcheck: Pin B4 on component <*comp_name*> is programmed but requires a signal connected to port B0.	
WARNING - blockcheck: The clock enable mux in PFU <*pfu_name*> is set to zero and registers are being used.	
WARNING - blockcheck: The FFLATCH<*number*> prim in PFU <*pfu_name*> does not have its SET/RST demux configured.	
WARNING - blockcheck: The signal on pin <*pin_name*> on component <*comp_name*> is unrouted.	
WARNING - blockcheck: TRI <*name*> is used but has no signal on pin O.	
WARNING - blockcheck: TRI <*name*> the signal on pin T has no effect since T is not programmed.	
WARNING - blockcheck: TRI <*name*> the signal on pin T has no effect.	
WARNING - blockcheck: <*number*> unplaced comp warnings not reported.	The design has <*number*> unplaced components. The physical DRC has not reported each component with a separate warning message.
WARNING - blockcheck: <*number*> unrouted net warnings not reported.	The physical DRC has not reported each unrouted net with a separate warning message.
ERRWARN - macrocheck: Macro <*macro_name*> has no external pins.	
ERRWARN - macrocheck: No external pins on macro <*macro_name*> have signals.	
ERROR - macrodefcheck: There are no components used for this macro.	A macro definition file does not contain any components.
WARNING - macrodefcheck: Reference component for this macro undefined or unplaced.	With EPIC in macro mode, you have not defined a reference component for the macro library file.
WARNING - macrodefcheck: There are no external pins defined for this macro.	A macro definition file does not contain any external pins.
WARNING - macrodefcheck: The reference component for this macro is not placed.	
WARNING - duplicate name check: a comp and a signal both have the same name <name>. This will be a problem when generating HDL code.	Although it is no problem for the NC database to have a signal and a pad share a name, a problem does arise if the backannotater is run and Verilog generated.

General Message	Explanation
WARNING - duplicate name check: empty name found on component.	Although it is no problem for the NC database to have a signal and a pad share a name, a problem does arise if the backannotater is run and Verilog generated.
WARNING - duplicate name check: empty name found on signal.	Although it is no problem for the NC database to have a signal and a pad share a name, a problem does arise if the backannotater is run and Verilog generated.
WARNING - duplicate name check: more than one comp has name <*name*>.	Although it is no problem for the NC database to have a signal and a pad share a name, a problem does arise if the backannotater is run and Verilog generated.
WARNING - duplicate name check: more than one signal has name <*name*>.	Although it is no problem for the NC database to have a signal and a pad share a name, a problem does arise if the backannotater is run and Verilog generated.
CATASTROPHIC - netsanitycheck: A CIP is used by more than one signal: '%s' and '%s'.	
CATASTROPHIC - netsanitycheck: A CIP is used more than once on signal '%s'.	
CATASTROPHIC - netsanitycheck: A CIP on signal '%s' isn't marked as used.	
CATASTROPHIC - netsanitycheck: Node %ld is used by more than one signal: '%s', and '%s'.	
CATASTROPHIC - netsanitycheck: Node %ld on signal '%s' isn't marked as used.	
CATASTROPHIC - netsanitycheck: Number of subnets not correct on signal '%s'.	
CATASTROPHIC - netsanitycheck: Sigpin on placed comp %s and signal '%s' has no node	
CATASTROPHIC - netsanitycheck: Sigpin 0x%lx on signal '<*signal_name*>', id %d, is crossed with signal '%s', id %d.	
CATASTROPHIC - netsanitycheck: Sigpin 0x%lx on signal '%s', id %d, should be both used and perm but isn't.	

LatticeSC DRC Warnings and Errors

LatticeSC Message	Explanation
BLOCK CHECKS	
All Components	
WARNING - blockcheck: No signals are connected to component <*name*>.	Checked for all components.

LatticeSC Message	Explanation
WARNING - blockcheck: The signal on pin <*name*> on component <*name*> is unrouted.	Checked for all components.
Slice	
ERROR - blockcheck: in SLICE <*slice_name*>, <*register_#*> is used but clock enable is not asserted to it.	CEMUX not used but register is used; Checked for LOGIC and RIPPLE modes.
ERROR - blockcheck: in SLICE <*slice_name*>, <*register_#*> is used but there is no active clock in the slice.	CLKMUX doesn't select CLK. It is an error unless CLKMUX tied "Hi" and REGMODE is LATCH. Checked for LOGIC and RIPPLE modes.
ERROR - blockcheck: In SLICE <*comp_name*>, RAM address pin <*pin_name*> is neither connected to signal nor programmed to constant. Doing so may disable RAM initial value.	A floating RAM address pin will cause the wrong RAM initial value to be returned.
ERROR - In SLICE <*slice_name*>, RCLKMUX selects CLKMUX, but there is no active CLK.	
ERROR - In SLICE <*slice_name*>, RCLKMUX selects CLK2MUX, but there is no active CLK2.	
ERROR - blockcheck: In SLICE <*slice_name*>, WCLKMUX and RCLKMUX should not select the same CLKMUX.	The SLICE in DPRAM requires write clock and read clock to be different. Checked for DPRAM mode.
ERROR - blockcheck: In SLICE <*slice_name*>, WCLKMUX and RCLKMUX should not select the same CLK2MUX.	The SLICE in DPRAM requires write clock and read clock to be different. Checked for DPRAM mode.
ERROR - In SLICE <*slice_name*>, WCLKMUX selects CLKMUX, but there is no active CLK.	
ERROR - In SLICE <*slice_name*>, WCLKMUX selects CLK2MUX, but there is no active CLK2.	
ERROR - blockcheck: more than one signal is connected to <*pin_name*> on <*prim_name*>. While in SPRAM mode, all ports connected to <*port_name*> must source from the same signal. Failing port is <*port_name*>.	Address pin pair (A0, A1), (B0, B1), (C0, C1), (D0, D1) must be connected to the same signal in SPRAM. Every pair is shorted together inside the SLICE. Checked for SPRAM mode.
ERROR - blockcheck: more than one signal is connected to <*primpin_name*> on <*comp_name*>. While in SPRAM mode, all ports connected to <*primpin_name*> must source from the same signal. Failing port is <*port_name*>.	SPRAM's address pin pair (A0, A1), (B0, B1), (C0, C1), (D0, D1) is shorted together. Applies to SPRAM.

LatticeSC Message	Explanation
ERROR - blockcheck: port <*port_name*> on comp <*comp_name*> has no signal on it. While in SPRAM mode, all ports connected to <*port_name*> must source from the same signal.	The port has no source. Address pin pair (A0, A1), (B0, B1), (C0, C1), (D0, D1) must be connected to the same signal in SPRAM. Every pair is shorted together inside the SLICE. Checked for SPRAM mode.
ERRWARN - blockcheck: In SLICE <*comp_name*>, CI is hanging. In ripple mode CI must either connect to a signal or tie to 0 through M0MUX.	The CI in ripple mode introduces a 1 input if not tied down.
ERRWARN - blockcheck: in SLICE <*slice_name*>, port OFX1 is used but M1MUX is not used.	Checked for LOGIC mode.
ERRWARN - blockcheck: slice <*slice_name*> is in DPRAM mode, so WCLKMUX or RCLKMUX must be used to select clock.	Checked for DPRAM mode.
WARNING - blockcheck: Comp <*comp_name*> has input signals but no output signals.	Checked for LOGIC, RIPPLE, SPRAM, and DPRAM modes.
WARNING - blockcheck: in <*comp_name*>, active pin <*pin_name*> of unknown direction on <*mux_name*> is not connected to anything within the comp.	Checked for LOGIC, RIPPLE, SPRAM, and DPRAM modes.
WARNING - blockcheck: in <*comp_name*>, input pin <*pin_name*> on <*mux_name*> is not selected but is driven within the comp.	Checked for LOGIC, RIPPLE, SPRAM, and DPRAM modes.
WARNING - blockcheck: in <*comp_name*>, input pin <*pin_name*> on <*mux_name*> is selected but not driven by anything within the comp.	Checked for LOGIC, RIPPLE, SPRAM, and DPRAM modes.
WARNING - blockcheck: in <*comp_name*>, output pin <*pin_name*> on <*mux_name*> is active but not connected to a load pin within the comp.	Checked for LOGIC, RIPPLE, SPRAM, and DPRAM modes.
WARNING - blockcheck: in <*comp_name*>, output pin <*pin_name*> on <*mux_name*> is inactive but has a load within the comp.	Checked for LOGIC, RIPPLE, SPRAM, and DPRAM modes.
WARNING - blockcheck: in SLICE <*slice_name*>, at least one register is used, so REGMODE must be set to LATCH or FF	Checked for LOGIC and RIPPLE modes.
WARNING - blockcheck: in SLICE <*slice_name*>, CEMUX is tied to low in LOGIC/RIPPLE mode.	Checked for LOGIC and RIPPLE modes.
WARNING - blockcheck: in SLICE <*slice_name*>, CEMUX must be used to select CE or be programmed to a 1	CEMUX is not used. It should be in SPRAM/DPRAM modes. Checked for SPRAM and DPRAM modes.

LatticeSC Message	Explanation
WARNING - blockcheck: in SLICE <*slice_name*>, LSRONMUX is on but LSRMUX is tied to high.	Checked for LOGIC and RIPPLE modes.
WARNING - blockcheck: in SLICE <*slice_name*>, LUT <*name*> is not programmed but may be selected by K0K1MUX, which is used.	Checked for LOGIC mode.
WARNING - blockcheck: in SLICE <*slice_name*>, M0MUX is not programmed but K0K1MUX is used	Checked for LOGIC mode.
WARNING - blockcheck: in SLICE <*slice_name*>, REGx has an input signal on D0, but the registerís output is not used. Check MxMUX(input) or Qx(output) programming.	Regx can only be on as Qx is selected. If MxMux has signal to D0 pin of Regx but Qx is not selected, there is a warning. Checked for LOGIC and RIPPLE modes.
WARNING - blockcheck: in SLICE <*slice_name*>, REG <*reg_num*> is used, so REGMODE must be set to LATCH or FF	Checked for SPRAM and DPRAM modes.
WARNING - blockcheck: in SLICE <*slice_name*>, ripple logic is in CNTDN mode, so pin Bx should tie to low.	Checked for RIPPLE mode.
WARNING - blockcheck: in SLICE <*slice_name*>, ripple logic is in CNTUP mode. Port <*port_name*> should not have a signal on it.	Port Bo/Bi's signal will be ignored in CNTUP/CNTDN mode. Checked for RIPPLE mode.
WARNING - blockcheck: in SLICE <*comp_name*>, ripple logic is in CNTDN mode, so pin B<*int*> should tie to low.	Applies to SPRAM.
WARNING - blockcheck: In SLICE <*slice_name*>, short box <*short_name*> has input signal(s), but its output <*pin_name*> is tied to constant.	A short box in SPRAM mode is shorting signals to hi or low, which is only allowed for debugging. Checked for SPRAM mode.
WARNING - blockcheck: Pin <*name*> on component <*name*> has a signal but is not used inside the comp.	Checked for LOGIC, RIPPLE, SPRAM, and DPRAM modes.
WARNING - blockcheck: Pin <*name*> on component <*name*> has no signal but is used inside the comp.	Checked for LOGIC, RIPPLE, SPRAM, and DPRAM modes.
WARNING - blockcheck: SLICE <*slice_name*>, LSRMUX input must not be programmed to a 0. Doing so disables writing to the RAM.	Checked for SPRAM and DPRAM modes.
WARNING - blockcheck: SLICE <*slice_name*>, CEMUX input must not be programmed to a 0. Doing so disables writing to the RAM.	CEMUX needs to provide clock enable signal or set to 1 for clock always enabled. Checked for SPRAM and DPRAM modes.

LatticeSC Message	Explanation
WARNING - blockcheck: SLICE <*slice_name*> in SPRAM/DPRAM mode, LSR is held hi, CE is held hi. Doing so enables writes continuously to the RAM.	Checked for SPRAM and DPRAM modes.
WARNING - blockcheck: SLICE <*slice_name*> is in SPRAM mode, but there is no active clock.	Either no signal is on CLK port or CLKMUX is not set to CLK. Checked for SPRAM mode.
EBR	
WARNING - blockcheck: in EBR <*comp_name*>, FULLPOINTER should be a value other than 0 to make use of FIFO.	
PCS	
ERROR - blockcheck: In PCS <*comp_name*>, <*prop_name*> set to <*v1_name*> conflicts with <*prop2_name*> set to <*v2_name*>. The rates must be the same or be 1/2 of the full rate if using half rate.	The PCS comp is setting different TX/RX rates between channels. All channels must use the same TX/RX rate or 1/2 of that rate.
PIO	
ERROR - blockcheck: In PIO <*comp_name*>, ODMUX selects the DATAMUX pin but OPENDRAIN is not ON.	ODMUX can select the DATAMUX pin only when OPENDRAIN is ON.
ERROR - blockcheck: In PIO <*comp_name*>, OPENDRAIN is ON but ODMUX is not connecting to a live signal from DATAMUX or TRIMUX.	ODMUX must have a live signal, either from DATAMUX or TRIMUX when OPENDRAIN is ON.
ERROR - blockcheck: PIO <*pio_name*> does not match any of the <*number*> combination(s) for IO_TYPE <*io_type*> Combination <*number*>: <*property_name*> needs to be <*property_value*>, but <*comp_name*> is different.	This PIO has an invalid property combination. If there is more than one valid combination under a certain <*io_type*>, violations of each possible combination will be listed by properties (attributes).
WARNING - blockcheck: in PIO <*comp_name*>, data is sent to output buffer but the buffer is tri-stated, please change odmux programming to 0 or signal(from trimux).	Output buffer is tristated when ODMUX is 1.
WARNING - blockcheck: in PIO <*pio_name*> input, ODMUX should be tied to high.	
WARNING - blockcheck: in PIO <*pio_name*>, ODMUX selects DATAMUX but DATAMUX is not active.	
WARNING - blockcheck: in PIO <*pio_name*>, ODMUX selects TRIMUX but TRIMUX is not active.	

LatticeSC Message	Explanation
WARNING - blockcheck: In PIO <*comp_name*>, when OPENDRAIN is ON and ODMUX is connecting to a live signal through TRIMUX, DATAMUX should be tied low, make sure the signal going into DATAMUX is grounded at certain location.	DATAMUX should be tied to 0 on its PADDO leg when OPENDRAIN is ON and DATAMUX connects to a live signal through TRIMUX.
WARNING - blockcheck: Pin <*name*> on component <*name*> has a signal but is not used inside the comp.	
WARNING - blockcheck: Pin <*name*> on component <*name*> has no signal but is used inside the comp.	
IOLOGIC/IOLOGICE	
ERROR - blockcheck: In IOLOGICE <*comp_name*>, AIL is ON but UPDT is not set to VALID.	UPDT must be set to VALID when AIL is ON.
ERROR - blockcheck: In IOLOGICE <*comp_name*>, UPDT is set to VALID but AIL is not ON.	UPDT can only be set to VALID when AIL is ON.
ERROR - blockcheck: in IOLOGICE/IOLOGIC <*iol_name*>, IDDR is in SHX2/SHX4/DDRX2/ DDRX4 mode but LSRMUX is tied to constant.	MLSRMUX selects LSRMUX but LSRMUX is tied: IDDR should have LSR live signal if in X2/X4 mode.
ERROR - blockcheck: in IOLOGICE/IOLOGIC <*iol_name*> in <*mode*>, FF is in LATCHFF mode but DEL is not used.	DEL must always be used in IREG_OREG/IREG_ODDR mode if LATCHFF is used.
ERROR - blockcheck: in IOLOGICE/IOLOGIC <*iol_name*>, ODDR is in SHX2/SHX4/DDRX2/ DDRX4 mode but LSRMUX is tied to constant.	OLSRMUX selects LSRMUX but LSRMUX is tied: ODDR should have LSR live signal if in X2/X4 mode.
ERROR - blockcheck: in IOLOGICE/IOLOGIC <*iol_name*>, TRIDDR is in DDR but ODDR is in DDRX/DDRX2/DDRX4/SHX2/SHX4.	When a pad is tristate-controlled, it cannot be in teose ODDR modes.
ERROR - blockcheck: in IOLOGICE/IOLOGIC <*iol_name*>, TRIDDR is in DDRX but ODDR is in DDR/DDRX2/DDRX4/SHX1/SHX2/SHX4.	When a pad is tristate-controlled, it cannot be in these ODDR modes.
ERROR - blockcheck: in <*comptype_name*> <*comp_name*>, the setting of REGSET on ODDR and TRIDDR must be the same.	
ERROR - blockcheck: <*comptype_name*> <*comp_name*> has signal on RUNAIL pin but AIL is not used.	
WARNING - blockcheck: in <*comptype_name*> <*comp_name*>, IDDR is programmed to be SHX2/SHX4/DDRX1/DDRX2/DDRX4 but HCLKIMUX is not selecting ECLK.	Both CLK and ECLK are required for these DDR modes.

LatticeSC Message	Explanation
WARNING - blockcheck: in <*comptype_name*> <*comp_name*>, ODDR is programmed to be SHX2/SHX4/DDRX2/DDRX4 but HCLKOMUX is not selecting ECLK.	Both CLK and ECLK are required for these DDR modes.
WARNING - blockcheck: In IOLOGICE <*comp_name*>, AIL is used but SRMODE is not set to ASYNC.	AIL can be used in both ASYNC and LSR_OVER_CE mode, but ASYNC is preferred.
WARNING - blockcheck: in IOLOGICE <*iol_name*>, IDDR is using RUNAIL pin but there is no AIL.	Only checked for to IOLOGICE.
WARNING - blockcheck: in IOLOGICE/IOLOGIC <*iol_name*> at IDDR_ODDR mode, CLKMUX is used but neither IDDR/ODDR's ICLK nor HCLKIMUX/HCLKOMUX is using it.	CLKMUX's output should be used by at least one of these. Checked for IDDR_ODDR mode.
WARNING - blockcheck: in IOLOGICE/IOLOGIC <*iol_name*> at IREG_ODDR mode, CLKMUX is used but none of ODDR's LCLK/FF's CLK/ HCLKOMUX is using it.	CLKMUX's output should be used by at least one of these. Checked for IREG_ODDR mode.
WARNING - blockcheck: in IOLOGICE/IOLOGIC <*iol_name*> at IREG_OREG mode, CLKMUX is used but neither FF's CLK nor HCLKOMUX is using it.	CLKMUX's output should be used by at least one of these. Checked for IREG_OREG mode.
WARNING - blockcheck: in IOLOGICE/IOLOGIC <*iol_name*> at IDDR_OREG mode, CLKMUX is used but neither IDDR's LCLK nor HCLKIMUX/ HCLKOMUX is using it.	CLKMUX's output should be used by at least one of these. Checked for IDDR_OREG mode.
WARNING - blockcheck: in IOLOGICE/IOLOGIC <*iol_name*>, DEL cell's current COARSE/FINE setting <*delay_value1*> cannot ensure zero hold time. To achieve zero hold time for PCLK/ECLK, COARSE/FINE delay must be <*delay_value2*>.	
WARNING - blockcheck: in IOLOGICE/IOLOGIC <*iol_name*>, FF is used but there is no active CE.	Both the port and multiplexer are not active.
WARNING - blockcheck: in IOLOGICE/IOLOGIC <*iol_name*>, FF is used but there is no active CLK.	Both the port and multiplexer are not active.
WARNING - blockcheck: in IOLOGICE/IOLOGIC <*iol_name*>, FF is in LATCHFF mode but there is no active ECLK.	Both the port and multiplexer on the path to LATCH's CLK are not active.
WARNING - blockcheck: in IOLOGICE/IOLOGIC <*iol_name*>, FF or LATCH is used but has no D input.	A signal should input to DELMUX and output to the FF or LATCH.
WARNING - blockcheck: in IOLOGICE/IOLOGIC <*iol_name*>, HCLKIMUX selects ECLK/CLK but there is no active ECLK/CLK.	

LatticeSC Message	Explanation
WARNING - blockcheck: in IOLOGICE/IOLOGIC <*iol_name*>, HCLKOMUX selects ECLK/CLK but there is no active ECLK/CLK.	Both the port and multiplexer on the path to HCLKOMUX are not active.
WARNING - blockcheck: in IOLOGICE/IOLOGIC <*iol_name*>, IDDR is not used but HCLKIMUX selects CLK clock.	
WARNING - blockcheck: in IOLOGICE/IOLOGIC <*iol_name*>, IDDR is not used but HCLKIMUX selects ECLK clock.	
WARNING - blockcheck: in IOLOGICE/IOLOGIC <*iol_name*>, IDDR is used but HCLKIMUX is not used to select clock.	
WARNING - blockcheck: in IOLOGICE/IOLOGIC <*iol_name*>, IDDR is programmed to be SHX2/SHX4/DDRX1/DDRX2/DDRX4, but CLKMUX is not used.	ICLK pin should be used for these IDDR modes.
WARNING - blockcheck: in IOLOGICE/IOLOGIC <*iol_name*>, IDDR is not used but MLSRMUX selects LSRMUX.	IDDR should be used if MLSRMUX is used
WARNING - blockcheck: in IOLOGICE/IOLOGIC <*iol_name*>, IDDR is not used but MLSRMUX selects ELSRMUX.	IDDR should be used if MLSRMUX is used.
WARNING - blockcheck: in IOLOGICE/IOLOGIC <*iol_name*>, IDDR is used but MLSRMUX is not used.	IDDR should have LSR
WARNING - blockcheck: in IOLOGICE/IOLOGIC <*iol_name*>, LATCHFF is used but ECLK and CLK are not in the same phase.	In LATCHFF mode, the clocks to LATCH and FF should be both inverted or neither inverted.
WARNING - blockcheck: in IOLOGICE/IOLOGIC <*iol_name*>, ODDR is programmed to be SHX2/SHX4/DDRX1/DDRX2/DDRX4, but CLKMUX is not used.	The LCLK pin should be used for these ODDR modes.
WARNING - blockcheck: in IOLOGICE/IOLOGIC <*iol_name*>, MLSRMUX selects LSRMUX, but there is no signal on LSRMUX.	LSR should have a signal if used.
WARNING - blockcheck: in IOLOGICE/IOLOGIC <*iol_name*>, MLSRMUX selects ELSRMUX, but there is no signal on ELSRMUX.	LSR should have a signal if used.
WARNING - blockcheck: in IOLOGICE/IOLOGIC <*iol_name*>, ODDR and TRIDDR are not used but OLSRMUX selects LSRMUX.	If OLSRMUX is used, then TRIDDR or ODDR should be used.
WARNING - blockcheck: in IOLOGICE/IOLOGIC <*iol_name*>, ODDR and TRIDDR are not used but OLSRMUX selects ELSRMUX.	If OLSRMUX is used, then TRIDDR or ODDR should be used.

LatticeSC Message	Explanation
WARNING - blockcheck: in IOLOGICE/IOLOGIC <*iol_name*>, ODDR or TRIDDR is used but OLSRMUX is not used.	ODDR/TRIDDR should have LSR.
WARNING - blockcheck: in IOLOGICE/IOLOGIC <*iol_name*>, none of ODDR/TRIDDR/OUTREG/TSREG is used but HCLKOMUX selects CLK clock.	
WARNING - blockcheck: in IOLOGICE/IOLOGIC <*iol_name*>, none of ODDR/TRIDDR/OUTREG/TSREG is used but HCLKOMUX selects ECLK clock.	
WARNING - blockcheck: in IOLOGICE/IOLOGIC <*iol_name*>, ODDR, TRIDDR, OUTREG or TSREG used but HCLKOMUX is not used to select clock.	HCLKOMUX should be active and output to OUTREG or TSREG.
WARNING - blockcheck: in IOLOGICE/IOLOGIC <*iol_name*>, OLSRMUX selects LSRMUX, but there is no signal on LSRMUX.	LSR should have a signal if used.
WARNING - blockcheck: in IOLOGICE/IOLOGIC <*iol_name*>, OLSRMUX selects ELSRMUX, but there is no signal on ELSRMUX.	LSR should have a signal if used.
WARNING - blockcheck: in IOLOGICE/IOLOGIC <*iol_name*>, OUTREG is used but CEOMUX is not used.	
WARNING - blockcheck: in IOLOGICE/IOLOGIC <*iol_name*>, OUTREG/TSREG/FF is used, but LSRMUX is not used.	
WARNING - blockcheck: in IOLOGICE/IOLOGIC <*iol_name*>, UP pin is used by IDDR or ODDR, but UPDT is not used.	UP pin and UPDT should always be used together.
WARNING - blockcheck: in IOLOGICE/IOLOGIC <*iol_name*>, UPDT is used, but UP pin is not used by either IDDR or ODDR.	UP pin and UPDT should always be used together.
WARNING - blockcheck: Pin <*name*> on component <*name*> has a signal but is not used inside the comp.	
WARNING - blockcheck: Pin <*name*> on component <*name*> has no signal but is used inside the comp.	
CLKDIV	
ERROR - blockcheck: in CLKDIV <*comp_name*> if ELSR port has a signal, there must be signals on both CLK and LSR input pins.	

LatticeSC Message	Explanation
WARNING - blockcheck: in CLKDIV <*comp_name*> if ELSR port has a signal, the DIV property of CLKDIV should not be set to 1.	
DLL / PLL	
ERROR - blockcheck: DLL <*comp_name*> is in <*mode_name*> mode and its CLKOS is fed back to CLKFB, but their clock divider setting CLKOB_DIV and CLKI_DIV are different.	
ERROR - blockcheck: DLL <*comp_name*> is in <*mode_name*> mode, so it cannot enable CLKOS_DUTY50 and use CLKOS_FPHASE at the same time.	
ERROR - blockcheck: In PLL <*comp_name*>, CLKOS_FDEL is used but CLKOS_MODE is not set to FDEL or DIV.	When CLKOS_FDEL is set to a non-zero value, CLKOS_MODE must be set to FDEL or DIV.
WARNING - blockcheck: DLL <*comp_name*> is in <*mode_name*> mode, so it cannot have different setting on CLKOS_DUTY50 and CLKOP_DUTY50.	
WARNING - blockcheck: DLL <*comp_name*> is in <*mode_name*> mode and is using CLKOS_DUTY50 so it should set CLKOS_PHASE to 270.	
WARNING - blockcheck: DLL <*comp_name*> is in <*mode_name*> mode and is using CLKOS_DUTY50 so it should have the saming CLKOS_PHASE and CLKOP_PHASE setting.	
WARNING - blockcheck: in <*comp_name*>, CLKOS_FDEL_ADJ is enabled, so CLKOS_FDEL_ADJVAL is used for fine adjustment, the CLKOS_FPHASE value will be disregarded.	
NET CHECKS	
ERROR - netcheck: in comp <*comp_name*>, port <*port_name*> is used in NOREG mode, but it is also being connected to maco wire.	When the EBR port is used in NOREG mode, it cannot also be used as an output to MACO.
ERROR - netcheck: in comp <*comp_name*>, port <*port_name*> is set to constant, but its cib resource is being used for routing.	Cib routing resource is shared with the comp in this case.
ERROR - netcheck: in comp <*comp_name*>, port <*port_name*> is inverted or used in cycle boost, but its cib resource is being fanned out or not being used.	

LatticeSC Message	Explanation
ERROR - netcheck: DLL comp <*comp_name*>'s pin DCNTL<*int*> is competing routing resources with CIP routing at node <*node_name*>.	The DCNTL pins of a pair of DLL comps share routing resources such that they go in the same direction. The signals must be from the same source among the three options: DLL comp in the pair, another DLL comp in the pair, or the CIP.
ERROR - netcheck: DLL comp <*comp_name*>'s pin DCNTL<*int*> is competing routing resources with pin DCNTL<*int*> on comp <*comp_name*>.	
ERROR - netcheck: Node <*node_name*> is used as a bounceback point in routing, but one of its connections is missing.	When a node is used as a bounceback point in routing, the arc from this bounceback point to the node outward should also be used in routing. This includes connections from the bounceback points to PLC, PIC, or SWITCHBOX. The arc is a pinwire in the case of PLC and PIC, a jumpwire in the case of SWITCHBOX.
CHIP CHECKS	
ERROR - chipcheck: An IOLOGIC(E)'s DLL bus must be driven by one DLL, but in comp <*comp_name*>, DCNTL<*int*> is driven by <*comp_name*> while DCNTL<*int*> is driven by <*comp_name*>.	
ERROR - chipcheck: comp <*comp_name*> and <*comp_name*> are occupying the same SMI address <*offset_name*>, please check their SMI_OFFSET settings.	Different models cannot use the same SMI_OFFSET address. Note that the FLXMC model takes four continuous addresses.
ERROR - chipcheck: comp <*comp_name*> at site(bank) <*site_name*> is using PCICLAMP that is not supported by the site(bank).	PCICLAMP can only be used at the top or bottom edge of the chip.
ERROR - chipcheck: Comp <*comp_name*> at site <*site_name*> in <*mode_name*> mode is using VCCIO 3.3 that is not supported by the site.	A PIO with VCCIO 3.3 or LVDS/RSDS/HYPT must be placed in a site that supports those features.
ERROR - chipcheck: Comp <*comp_name*> at site <*site_name*> in <*mode_name*> mode is using LVDS/RSDS/HYPT that is not supported by the site.	A PIO with VCCIO 3.3 or LVDS/RSDS/HYPT must be placed in a site supports those features.
ERROR - chipcheck: Comp <*comp_name*> has SMI_ADDR_DIS setting of \"<*setting_name*>\" that is different with comp <*comp_name*>'s SMI_ADDR_DIS \"<*setting_name*>\".	All PLL or DLL components at the same corner (such as upper left) must have the same SMI_ADDR_DIS setting.
ERROR - chipcheck: Complementary PIO <*comp_name*> is placed on site <*site_name*> with its pairing site <*site_name*> taken by PIO <*comp_name*>.	Pairing PIC rule: All complementary modes' (HYPT, LVPECL, LVDS, RSDS, BLVDS25, MLVDS25, and all modes with letter "D") PIOs take over their neighboring PIO (in each half PIC). Only one set of PIO IOLOGIC sites can have comps placed in them.

LatticeSC Message	Explanation
ERROR - chipcheck: Complementary PIO <*comp_name*> is placed on pad B/D's site <*site_name*>.	Pairing PIC rule: Complementary modes (HYPT, LVPECL, LVDS, RSDS, BLVDS25, MLVDS25, and all modes with letter "D") can only be placed on TRUE sites. Only A and C pads are TRUE sites.
ERROR - chipcheck: Differential output PIO <*comp_name*> is placed on pad C/D's site <*site_name*>.	All differential outputs can only be placed on pad A or B sites.
ERROR - chipcheck: DPRAM SLICE <*comp_name*> is not placed in pair.	Two DPRAM sites can only be placed in pairing SLICE sites. (In one PFU, site A and B, C and D are pairs).
ERROR - chipcheck: DPRAM SLICE <*comp_name*> is not connected together by DPI/DPO ports with SLICE <*comp_name*>.	Two DPRAM sites can only be placed in pairing SLICE sites. (In one PFU, site A and B, C and D are pairs).
ERROR - chipcheck: DPRAM SLICE <*comp_name*>'s <CLK/CLK2> port is not connected to its pair SLICE <*comp_name*>'s <CLK2/CLK> port.	In a pair of DPRAMs, if one's CLK2(CLK) is used, it must be connected to the other's CLK(CLK2).
ERROR - chipcheck: HYPT buffer is used together with LVDS/RSDS in the same bank. Comp <*comp_name*> is in conflict with other PIOs in bank <*int*> refer to comp <*comp_name*>).	I/O BANK rule: There can be only one type of differential output driver in one bank. Current PIO types that use output driver bias: LVDS, RSDS, or HyperTransport. The exception to this rule is that LVDS and RSDS can be mixed in a bank.
ERROR - chipcheck: In bank <*int*>, Comp <*comp_name*> using VTT or DDR_II is in conflict with <*comp_name*> that uses VCMT.	I/O BANK rule: VTT/DDR_II and VCMT are mutually exclusive in a bank.
ERROR - chipcheck: In bank <*int*>, comp <*comp_name*> using external reference circuit is competing for the VREF1 rail with comp <*comp_name*> or <*comp_name*>.	When differential output is using external reference circuit, it takes VREF1 rail of the bank, so other comps cannot use VREF1.
ERROR - chipcheck: In bank <*int*>, differential output comp <*comp_name*> using external reference circuit is in conflict with comp <*comp_name*> using VREF1_DRIVER.	
ERROR - chipcheck: In bank <*int*>, differential output comp <*comp_name*> using external reference circuit is in conflict with comp <*comp_name*> using VREF1_LOAD.	
ERROR - chipcheck: In bank <*int*>, two differential output comps, <*comp_name*> using external reference circuit and <*comp_name*> using internal reference circuit are in conflict.	All differential outputs in a bank must use the same reference circuits, either external or internal.
ERROR - chipcheck: In bank <*int*>, VCCIO for <*comp_name*>, <*VCCIO_VALUE*> is incompatible with the other PIO that uses VCCIO <*VCCIO_VALUE*> (refer to comp <*comp_name*>).	I/O BANK rule: All PIOs in a bank must agree on the VCCIO value.

LatticeSC Message	Explanation
ERROR - chipcheck: In bank <*int*>, VREF for <*comp_name*> is different with the other two VREFs at comp <*comp1*> and <*comp2*>.	I/O BANK rule: All VREFs used in a bank must agree with either of the VREF1_LOAD or VREF2_LOAD's VREF values.
ERROR - chipcheck: In bank <*int*>, VTT for <*comp_name*>, <*VTT_VALUE*> is incompatible with the other PIO that uses VTT <*VTT_VALUE*> (refer to comp <*comp_name*>).	I/O BANK rule: If there are more than one VTT pads in a bank, they must agree on the VTT value.
ERROR - chipcheck: IOLOGIC(E) <*comp_name*> at pad <A|B|C|D> and IOLOGIC(E) <*comp_name*> at pad <A|B|C|D> are competing for resources (x2, x4, AIL or tri-state) that can't be shared in one PIC.	4x2 PIC Buffer rule: All IOLOGIC(E) types in a 4x2 PIC must be compatible with each other.
ERROR - chipcheck: PIO <*comp_name*> at pad <A|B|C|D> cannot be used as output when one IOLOGIC(E) <*comp_name*> at pad <A|B|C|D> is in SHX4/DDRX4 mode.	4x2 PIC Buffer rule: All IOLOGIC(E) types in a 4x2 PIC must be compatible with each other.
ERROR - chipcheck: PIO <*comp_name*> in bank <*int*> is a differential output driver, there must be a REF_RESISTOR in that bank.	In a bank that has differential PIO output drivers, there must be REF_RESISTOR.
ERROR - chipcheck: PIO <*comp_name1*> and <*comp_name2*> in bank <*int*> are using VCMT but their IO_TYPEs <*io_type1*> and <*io_type2*> are different.	I/O BANK rule: If more than one pad uses VCMT in a bank, the PIOs using them must be the same IO_TYPE.
ERRWARN - chipcheck: PIO <*comp_name*> in bank %d is using VREF1_LOAD, but there is no VREF1_DRIVER in that bank.	I/O BANK rule: When at least one VREF1_LOAD is used in a bank, there must be at least one VREF1_DRIVER in the same bank.
ERRWARN - chipcheck: PIO <*comp_name*> in bank %d is using VREF2_LOAD, but there is no VREF2_DRIVER in that bank.	I/O BANK rule: When at least one VREF2_LOAD is used in a bank, there must be at least one VREF2_DRIVER in the same bank.
ERROR - chipcheck: <*comp_name*> is using an invalid SMI_OFFSET setting: 0x400	0x400 is a base address not allowed in SMIOFFSET setting.
ERROR - chipcheck: When IOLOGIC/IOLOGICE <*comp_name*> is used, its corresponding PIO must also be used.	When a pad's IOLOGIC (E) is used, the corresponding PIO must also be used.
WARNING - chipcheck: In bank <*int*>, PIO <*comp1_name*> and <*comp2_name*> are using the same VREF but their IO_TYPEs <*mode1_name*> and <*mode2_name*> are different.	I/O BANK rule: All VREFs used in a bank should agree with either of the VREF1_LOAD/VREF2_LOAD VREF's IO_TYPEs.
WARNING - chipcheck: When JTAG <*comp_name*> is used, the <*comptype_name*> <*comp_name*> has a signal dedicated to JTAG.	When a JTAG comp is instantiated, the comps TCK, TMS, TDI, and TDO are dedicated to JTAG, so they cannot have any signals on them.

LatticeSC Message	Explanation
WARNING - chipcheck: When RDBK is used in <*comp_name*>, the <*comptype_name*> <*comp_name*> has a signal that is dedicated to RDBK.	When a RDBK comp is enabled, the comp TCK,TMS,TDI, and TDO cannot have any signals on them, otherwise it will potentially unstabilize the chip.
WARNING - chipcheck: <*pin_name*> <*comp_name*> is not connecting to <*pin_name*> of SYSBUS.	A component that sets SMI_OFFSET must use SMI RD to connect to a SYSBUS pin that corresponds to its SMI_OFFSET setting.

LatticeSCM MACO DRC Warnings and Errors

Please note that the LatticeSCM architecture will also include all DRC warning and error messages associated with the LatticeSC architecture.

LatticeSCM MACO Message	Explanation
BLOCK CHECK	
ERROR - blockcheck: Customer defined MACO <*comp_name*> is placed on an invalid site <*site_name*> for package <*package_name*>.	A customer-defined MACO can only be placed on the site assigned by software. Possible cause for this error is no entry for the site has been created in the master table.
ERROR - blockcheck: Customer defined MACO <*comp_name*> is placed on site <*site_name*> that does not support the <*mode_name*> function for package <*package_name*>.	A customer-defined MACO can only be placed on the site assigned by software. Possible cause for this error is no entry for the site has the customer defined mode in the master table.
ERROR - blockcheck: MACO <*comp_name*> is placed on an invalid site <*site_name*> for package <*package_name*>.	All MACO sites cannot be used in all packages. Move the comp to other MACO sites.
ERROR - blockcheck: MACO <*comp_name*> is placed on site <*site_name*> that does not support the <*mode_name*> function for package <*package_name*>.	The MACO comp is using a function that is not supported by the site. Move MACO comp to other sites or change its function.
ERROR - blockcheck: MACO <*comp_name*> on site <*site_name*> is using <*mode_name*> function that is not supported for package <*package_name*> because the MACO comp <*comp_name*> at upper site <*site_name*> is using <*mode_name*> function.	The MACO comp is using a function that is not supported by the site when its upper neighboring MACO site is instantiated for some function. Try to move this MACO comp or its upper neighbor MACO, or change their functions.

Logical DRC Tests

The Logical DRC, also called the NGD DRC, is a series of six tests to verify the logical design in the .ngd file. The checks are device-independent; they do not depend on the FPGA to which you will eventually map the design.

The Logical DRC generates error messages (for conditions where the logic does not operate correctly) or warnings (for conditions where the logic is incomplete).

The Logical DRC runs automatically at these times:

- At the end of the NGDBUILD program, before NGDBUILD writes out the .ngd file. NGDBUILD writes out the .ngd file even if DRC warnings or errors are discovered. The Build Database process in ispLEVER's Project Navigator is the equivalent to NGDBUILD.
- At the end of each netlist writer program (NGD2EDIF, NGD2VHD, or NGD2VER), before writing out the netlist file. The netlist writers do not perform the entire DRC; they only perform a subset of the DRC tests. A netlist writer program writes out a netlist file even if DRC warnings or errors are discovered.

The Block Check

Verifies that each terminal block in the NGD hierarchy (that is, each block that is not resolved to any lower-level components) is an NGD primitive. A block check failure is treated as a warning, since a design containing unexpanded non-primitive blocks may still be mapped if the technology mapper can recognize the block in question.

The Net Check

Determines the number of NGD primitive output pins (drivers), tristate pins (drivers), and input pins (loads) on each signal net in the design. If a net does not have at least one driver (or tristate driver) and at least one load, a warning is generated. A warning is also generated if a net has multiple non-tristate drivers or any combination of tristate and non-tristate drivers.

The Pad Check

Verifies that each net connected to pad primitives obeys the following rules:

- If the PAD is an input pad, the net to which it is connected can only be connected to
  - a BUF primitive.
  - a CKBUF primitive.
  - an OSC primitive.
  - a PULLUP primitive.
  - a PULLDOWN primitive.

  The input net can be attached to multiple primitives, but only one of each of these types. For example, the net can be connected to a BUF primitive, an OSC primitive, and a PULLUP primitive, but it cannot be connected to a BUF primitive and *two* OSC primitives. Also, the net cannot be connected to both a PULLUP primitive and a PULLDOWN primitive. Any violation results in an error, except for nets attached to multiple pullups or pulldowns, which produces a warning. Nets attached to none of these types of primitives also produces a warning.
- If the PAD is an output pad, the net it is attached to can only be connected to:
  - a single BUF primitive output, or

- a single TRI primitive output, or
- a single OSC primitive output,

in addition to:

- a single PULLUP primitive, or
- a single PULLDOWN primitive.

Any other primitive output connections on the net results in an error.

If this condition is met, the output PAD net may also be connected to one CKBUF primitive *input* and/or one BUF primitive *input*.

- If the PAD is a bidirectional or unbonded pad, the net it is attached to must obey the rules for input and output pads, except that OSC connections are prohibited. Any other primitive connections on the net results in an error. The net connected to the pad must be configured as both an input and an output net; if it is not, it produces a warning.
- If the net attached to the pad has a connection to a top-level block of the design, that top-level block pin must have the same type as the pad pin, except that output pads can be associated with tristate top-level pins. A violation of this rule is a warning.
- No net can be connected to multiple pads (an error) or to multiple top-level pins (a warning).

The Clock Buffer Check

Verifies that the output of each clock buffer primitive is connected to only flip-flop or latch primitive clock inputs. Violations are treated as warnings.

The Name Check

Verifies the uniqueness of names on NGD objects as follows:

- Pin names must be unique within a block. (error)
- Block names must be unique within the block's position in the hierarchy. That is, a block cannot have two blocks with the same name under it. (warning)
- Signal names must be unique within the signal's hierarchical level. That is, if you push down into a block, you cannot have two signals with the same name. (warning)
- Global signal names must be unique within the design. (warning)

The Primitive Pin Check

Verifies that certain pins on certain primitives are connected to nets in the design. The check tests these pins on these NGD primitive types:

Table 1: Pins Checked by Primitive Pin Check

NGD Primitive	Pins Checked
neotri	IN, OUT, and CTL
neoff	IN, OUT, and CLK
neolatch	IN, OUT, and CLK
neoipad	PAD
neoopad	PAD
neobpad	PAD

If one of these pins is not connected to a net, you receive a warning.