Microsemi Corporation

October 31, 2018

Customer Notification No: CN19001

Subject: RTG4 LSRAM Data Errors when Accessing SmartDebug

Description

RTG4 LSRAM blocks support real-time memory debug using the SmartDebug software. A debug session using SmartDebug memory debug can inadvertently write incorrect data into the LSRAM being debugged. For example, if the user design has the LSRAM's block select (A_BLK[2:0], B_BLK[2:0]) and byte write enable (A_WEN[1:0], B_WEN[1:0]) inputs enabled when the SmartDebug memory access occurs, the instantaneous transition between user access and SmartDebug access to the LSRAM could result in unintentional writes to the LSRAM.

To prevent this undesirable scenario, employ a manual handshake technique so that the user logic in the fabric can disable the block select and write enable inputs to the LSRAM before performing a SmartDebug memory debug access.

Application Impact

All LSRAM configurations in dual-port and two-port modes are impacted when using SmartDebug memory debug.

Action Required

To avoid LSRAM data errors during SmartDebug access, the write ports A and B must be halted prior to SmartDebug LSRAM access. This requirement applies to both dual-port and two-port LSRAM usage.

A manual handshake signal can be added to the design to control the LSRAM's block select (A_BLK[2:0], B_BLK[2:0]) and byte write enable (A_WEN[1:0], B_WEN[1:0]) inputs prior to accessing the LSRAM using SmartDebug.

Manual Handshake Workaround

The manual handshake signal can be created by using a flip-flop whose value can be manually changed using SmartDebug active probe writes. This flip-flop output can be used to gate the A_BLK, B_BLK , A_WEN, and B_WEN inputs to the LSRAM by modifying the user HDL. When the user is ready to perform a SmartDebug memory access, they must first use the Active Probe interface in SmartDebug, select the flip-flop that gates the control signals to the RAM they plan to debug, and write a logic-1 into the flip-flop. This will disable the fabric block selects and write enables to the LSRAM. Once fabric write access to the LSRAM is disabled the user can switch to the Memory Debug tab in SmartDebug to perform the memory accesses. After the memory debug is complete, another active probe write is required to set the handshake flip-flop to logic-0, thus re-enabling fabric writes to the LSRAM.

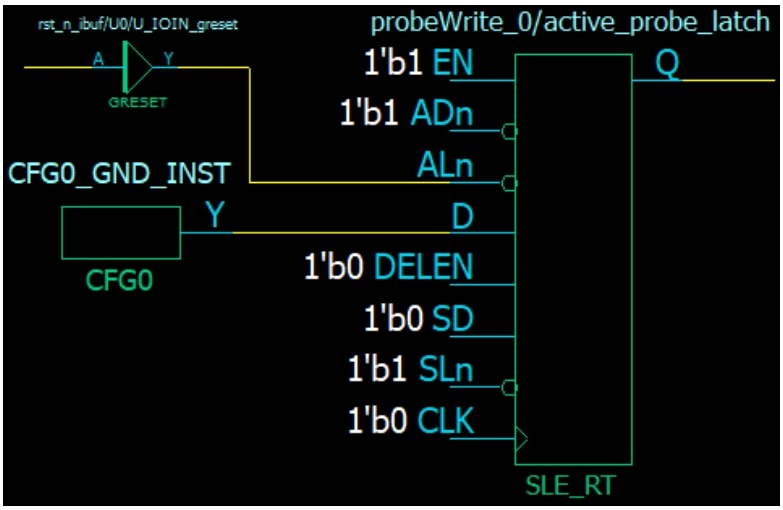
The example HDL in the following code block shows a module that can be added to the user design, and instantiated once per design or once per logical memory instance to create the handshake signal. Note that there are specific synthesis directives applied to prevent the active_probe_latch SLE from being optimized away during synthesis. The HDL in the following code block specifies that the active_probe_latch SLE must be reset to logic-0 during design reset to ensure that the LSRAM write ports are accessible to the user design by default. The RTG4 embedded power-on-reset signal distributed to the SLE ALn input will automatically reset the active_probe_latch flip-flop because the code in the following code block maps the ADn input to a logic-1, as shown in Figure 1. For more information about the SLE_RT truth table, refer to the RTG4 Macro Library Guide.

Code Block 1 • Workaround Module Code

```
module probeWrite (
  RESET_N_IN,
  PRBWR_IN,
  ENABLE_HNDSHAKE,

HNDSHAKE_OUT
  )                    /* synthesis syn_hier = "hard" */;
  input RESET_N_IN;
  input PRBWR_IN;
  input ENABLE_HNDSHAKE;
  output HNDSHAKE_OUT;

  reg active_probe_latch /*synthesis syn_preserve = 1*/;
  wire dummyClk /*synthesis syn_keep=1*/;

// This latch is to be driven asynchronously by Active Probe Write to 1 or 0
  always @(posedge dummyClk or negedge RESET_N_IN)
  begin
      if (!RESET_N_IN)
            active_probe_latch      <= 1'b0;
      else
            active_probe_latch      <= PRBWR_IN;
  end

  assign HNDSHAKE_OUT = active_probe_latch & ENABLE_HNDSHAKE;

endmodule
```

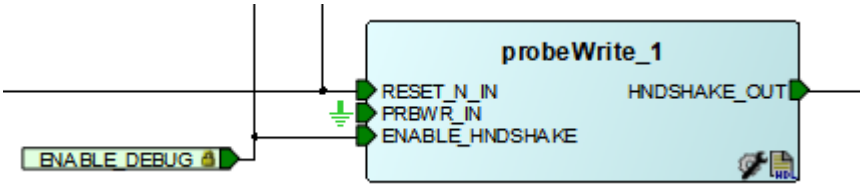
The post-compile netlist view of the active_probe_latch is shown in the following figure.

Figure 1 • Post-Compile Netlist View of active_probe_latch



When instantiating the probeWrite module, the PRBWR_IN input to the active_probe_latch can be statically tied LOW since the writes to this SLE will be driven by SmartDebug Active Probe writes. The HNDSHAKE_OUT signal will only propagate to user logic if the ENABLE_HNDSHAKE input is HIGH. This provides control to enable or disable debug in the design, if required. For example, during design development, it would make sense to tie the ENABLE_HNDSHAKE input HIGH. Once the design is finalized and debug is no longer required, the ENABLE_HNDSHAKE signal can be driven LOW to prevent the active_probe_latch output from gating-off user writes to the LSRAM. The following figure shows an example instantiation of this module.

Figure 2 • Instantiation of probeWrite Module with PRBWR_IN Tied LOW



An instantiated LSRAM can be modified to use gated versions of A_BLK, B_BLK, A_WEN, and B_WEN as shown in the following code.

Code Block 2 • Modified Instantiated LSRAM

```
wire A_BLK_gated, B_BLK_gated, A_WEN_gated, B_WEN_gated;
wire EN_DEBUG;
wire DEBUG_ACTIVE;

assign A_BLK_gated = DEBUG_ACTIVE ? 1'b0 : A_BLK;
assign B_BLK_gated = DEBUG_ACTIVE ? 1'b0 : B_BLK;
assign A_WEN_gated = DEBUG_ACTIVE ? 1'b0 : A_WEN;
assign B_WEN_gated = DEBUG_ACTIVE ? 1'b0 : B_WEN;

probeWrite probeWrite_0(
    // Inputs
    .RESET_N_IN      ( RESET_N ),
    .PRBWR_IN        ( 1'b0 ),
    .ENABLE_HNDSHAKE ( EN_DEBUG ),
    // Outputs
    .HNDSHAKE_OUT    ( DEBUG_ACTIVE )
    );
```

These gated versions of block select and write enable can then be mapped to the RAM macro block select and write enable ports as shown in the following code.

Code Block 3 • Gated Block Select and Write Enable

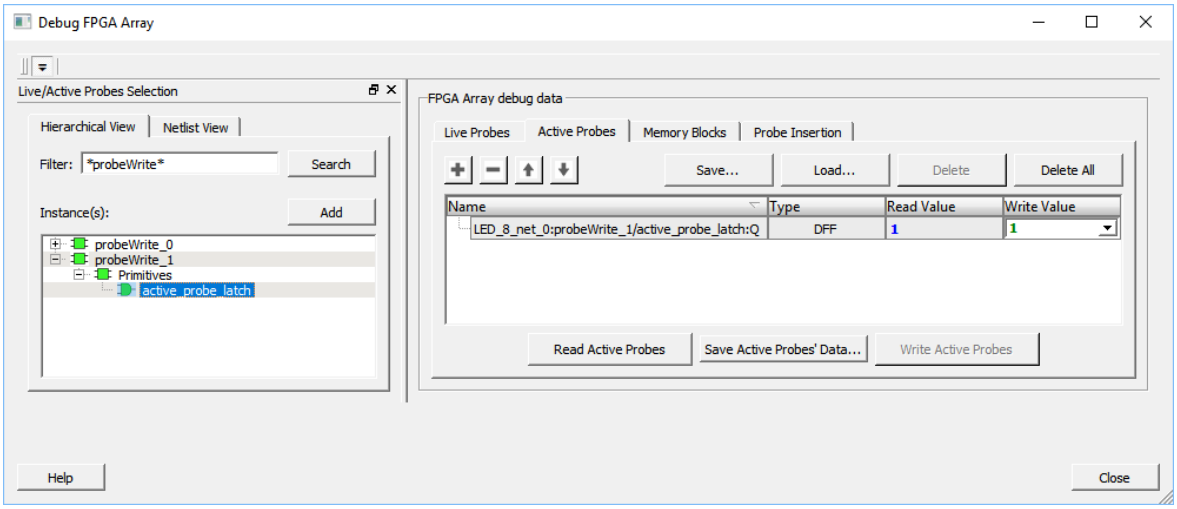
```
RAM1K18_RT myDPRAM (
…
        .A_BLK({A_BLK_gated, 1'b1, 1'b1}),

        .B_BLK({B_BLK_gated, 1'b1, 1'b1}),
…
       .A_WEN({A_WEN_gated, A_WEN_gated}),
.B_WEN({B_WEN_gated, B_WEN_gated}),
…
);
```

Note that connecting any one bit of A_BLK[2:0] and B_BLK[2:0] to a logic-0 is sufficient to turn off that LSRAM port's block select input. The example above shows only one bit of the block select input driven by the gated version of A_BLK and B_BLK while the other two bits are tied to logic-1.

In contrast, the write enable input can be used on a per-byte basis in some RAM configurations, and thus both bits of A_WEN[1:0] and B_WEN[1:0] should be driven by the gated write enable signal.

Once the LSRAM is set up to use the gated versions of block select and write enable, the user can employ the Active Probe write in SmartDebug to enable the handshake signal from the active_probe_latch and then perform Memory Debug on the LSRAM as shown in the following screenshot.

Figure 3 • Writing a Logic-1 to the active_probe_latch Handshake Signal



Products Affected by this Change

See Appendix A.

Contact Information:

If you have any questions, please contact Microsemi's SoC Technical Support at soc_tech@microsemi.com.

Regards,

Microsemi Corporation

Appendix A

Table 1 • Part Numbers Affected by CN19001

Microsemi Part Number	DLA SMD Number
RT4G150-CB1657PROTO	
RT4G150-CG1657B	5962-1620801QXF
RT4G150-CG1657E	5962-1620805QXF
RT4G150-CG1657EV	
RT4G150-CG1657V	5962-1620809VXF
RT4G150-CG1657PROTO	
RT4G150-CQ352B	
RT4G150-CQ352E	
RT4G150-CQ352E	
RT4G150-CQ352PROTO	
RT4G150-LG1657B	5962-1620803QZC
RT4G150-LG1657E	5962-1620807QZC
RT4G150-LG1657EV	
RT4G150-LG1657V	5962-1620811VZC
RT4G150-LG1657PROTO	
RT4G150-1CB1657PROTO	
RT4G150-1CG1657B	5962-1620802QXF
RT4G150-1CG1657E	5962-1620806QXF
RT4G150-1CG1657EV	
RT4G150-1CG1657V	5962-1620810VXF
RT4G150-1CG1657PROTO	
RT4G150-1CQ352B	
RT4G150-1CQ352E	
RT4G150-1CQ352EV	
RT4G150-1CQ352PROTO	
RT4G150-1LG1657B	5962-1620804QZC
RT4G150-1LG1657E	5962-1620808QZC
RT4G150-1LG1657EV	
RT4G150-1LG1657V	5962-1620812VZC
RT4G150-1LG1657PROTO	

Microsemi Headquarters
One Enterprise, Aliso Viejo,
CA 92656 USA
Within the USA: +1 (800) 713-4113
Outside the USA: +1 (949) 380-6100
Sales: +1 (949) 380-6136
Fax: +1 (949) 215-4996
Email: sales.support@microsemi.com
www.microsemi.com

Microsemi, a wholly owned subsidiary of Microchip Technology Inc. (Nasdaq: MCHP), offers a comprehensive portfolio of semiconductor and system solutions for aerospace & defense, communications, data center and industrial markets. Products include high-performance and radiation-hardened analog mixed-signal integrated circuits, FPGAs, SoCs and ASICs; power management products; timing and synchronization devices and precise time solutions, setting the world's standard for time; voice processing devices; RF solutions; discrete components; enterprise storage and communication solutions; security technologies and scalable anti-tamper products; Ethernet solutions; Power-over-Ethernet ICs and midspans; as well as custom design capabilities and services. Microsemi is headquartered in Aliso Viejo, California, and has approximately 4,800 employees globally. Learn more at www.microsemi.com.