Keywords: DDR1, DDR3, jitter, buffer, TDMoP, TDM over packet, DDR, SDRAM, PDV, PSN, double data rate

APPLICATION NOTE 5120

Using a DDR3 Memory Module with the DS34S132

Aug 26, 2011

Abstract: This application note explains how to interface the DS34S132, a 32-point TDM-over-packet IC, with a DDR3 memory chip. The DS34S132 uses an external double data rate (DDR) synchronous DRAM (or DDR1) device to buffer data. The memory supplies sufficient buffer space to support a 256ms Packet-Delay Variation (PDV) for each of the 256 pseudowires (PWs)/bundles and to enable packet reordering.

Introduction

The DS34S132, a 32-port TDM-over-packet IC, uses an external double data rate (DDR) synchronous DRAM (also referred to as "DDR1") device to buffer data. The large memory supplies sufficient buffer space to support a 256ms Packet-Delay Variation (PDV) for each of the 256 individually configurable pseudowires (PWs)/bundles. The memory also enables packet reordering if the packet switch network (PSN) incorrectly arranges the packets. Since DDR3 is readily available, it is more convenient to use a DDR type 3 (DDR3) memory chip with the DS34S132. This application note explains how to interface the DS34S132 with a DDR3 memory chip.

Figure 1 shows the proposed block diagram to replace DDR1 with a FPGA and DDR3.

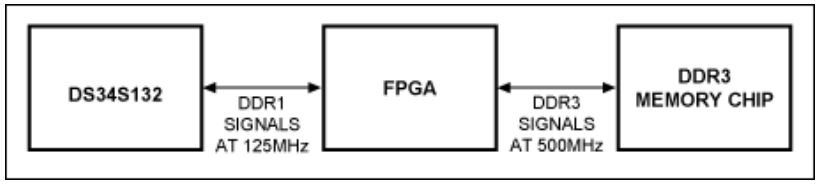


Figure 1. Replacing DDR1 with DDR3 and FPGA.

The DDR SDRAM interface has higher transfer rates than a typical SDRAM, due to its sophisticated timing control of electrical data and clock signals. For example, a DDR SDRAM with a 125MHz clock frequency can achieve almost twice the bandwidth (BW) of a SDRAM running at the same frequency. Thus, Maxim now uses DDR1 instead of SDRAM, which was used in the previous generation of Maxim's TDM over Packet (TDMoP) devices.

DDR3 SDRAM is a DRAM-interface specification. The actual DRAM arrays that store the data are similar to earlier types, with similar performance. DDR3 SDRAM can transfer data at four times the rate of DDR1, and thus enables higher bandwidth.

These days, DDR1 memory modules are not as widely available as DDR2 or DDR3. Unfortunately, DDR2 and DDR3 are neither backward- nor forward-compatible with DDR1. Thus, DDR2 or DDR3 memory modules will not work in DDR-equipped motherboards, and vice versa.

DDR Interface Configuration of the DS34S132

Inside the DS34S132 TDMoP device,

- The DDR1 interface must be programmed to a column-address-strobe (CAS) latency of 3.
- The "refresh rate" must be calculated and programmed to have a fast-enough interval for the DDR3 memory module.
- The clock rate for the DDR is 125MHz.

It is worth noting that DDR3 has 8 banks. The DS34S132 has only 2 bank select bits. Hence one half of the DDR3 memory module (upper banks) will be unused.

Configuration of the DDR3

DDR3 should run at a clock speed of 500MHz. This frequency is chosen since it is four times the DDR1 clock speed of 125MHz.

For the Verilog® RTL simulation, we used 8-bit data, which will have a BW of 500MHz × 2 × 8 bits. This BW is twice the BW of 125MHz × 2 × 16 bits on the DDR side. The extra BW will be used to pipeline and pass back the read data from DDR3 to the DS34S132 without use of FPGA FIFO memory. For the DDR3, we used:

- CAS latency: 8
- CAS write latency: 6
- DLL reset, then enabled.

The Micron DDR3 MT41J128M8 (16 Meg × 8 × 8 banks) specifications were used for simulation. The cycle time for the DDR3 was 1.87ns at CL = 8 (DDR3 - 1066) -187. Support for other DDR3 configurations/speed may require incremental or significant effort. The DDR3 that we used for simulation had the following specifications:

```
'elsif sg187        // sg187  is equivalent to the JEDEC DDR3-1066G (8-8-8) speed bin
eter TCK_MIN         1875; // tCK        ps  Minimum Clock Cycle Time
eter TJIT_PER          90; // tJIT(per)  ps  Period JItter
eter TJIT_CC          180; // tJIT(cc)   ps  Cycle to Cycle jitter
eter TERR_2PER        132; // tERR(2per) ps  Accumulated Error (2-cycle)
eter TERR_3PER        157; // tERR(3per) ps  Accumulated Error (3-cycle)
eter TERR_4PER        175; // tERR(4per) ps  Accumulated Error (4-cycle)
eter TERR_5PER        188; // tERR(5per) ps  Accumulated Error (5-cycle)
eter TERR_6PER        200; // tERR(6per) ps  Accumulated Error (6-cycle)
eter TERR_7PER        209; // tERR(7per) ps  Accumulated Error (7-cycle)
eter TERR_8PER        217; // tERR(8per) ps  Accumulated Error (8-cycle)
eter TERR_9PER        224; // tERR(9per) ps  Accumulated Error (9-cycle)
eter TERR_10PER       231; // tERR(10per)ps  Accumulated Error (10-cycle)
eter TERR_11PER       237; // tERR(11per)ps  Accumulated Error (11-cycle)
eter TERR_12PER       242; // tERR(12per)ps  Accumulated Error (12-cycle)
eter TDS               75; // tDS        ps  DQ and DM input setup time relative to DQS
eter TDH              100; // tDH        ps  DQ and DM input hold time relative to DQS
eter TDQSQ 150; // tDQSQ  ps DQS-DQ skew, DQS to last DQ valid, per group, per access
eter TDQSS  0.25; // tDQSS    tCK   Rising clock edge to DQS/DQS# latching transition
eter TDSS             0.20; // tDSS   tCK DQS falling edge to CLK rising (setup time)

eter TDSH             0.20; // tDSH   tCK DQS falling edge from CLK rising (hold time)
eter TDQSCK            300; // tDQSCK     ps    DQS output access time from CK/CK#
eter TQSH             0.38; // tQSH       tCK   DQS Output High Pulse Width
eter TQSL             0.38; // tQSL       tCK   DQS Output Low Pulse Width
eter TDIPW             490; // tDIPW      ps    DQ and DM input Pulse Width
eter TIPW              780; // tIPW       ps    Control and Address input Pulse Width
eter TIS               275; // tIS        ps    Input Setup Time
eter TIH               200; // tIH        ps    Input Hold Time
eter TRAS_MIN        37500; // tRAS    ps   Minimum Active to Precharge command time
eter TRC             52500; // tRC     ps   Active to Active/Auto Refresh command time
eter TRCD            15000; // tRCD       ps    Active to Read/Write command time
eter TRP             15000; // tRP        ps    Precharge command period
eter TXP              7500; // tXP        ps    Exit power down to a valid command
eter TCKE             5625; // tCKE       ps    CKE minimum high or low pulse width
eter TAON              300; // tAON       ps    RTT turn-on from ODTLon reference
eter TWLS              245; // tWLS       ps    Setup time for tDQS flop
eter TWLH              245; // tWLH       ps    Hold time of tDQS flop
eter TWLO             9000; // tWLO       ps    Write levelization output delay
eter TAA_MIN         15000; // TAA        ps    Internal READ command to first data
eter CL_TIME         15000; // CL         ps    Minimum CAS Latency
```

FPGA Configuration

The FPGA has critical functions:

- At power-up, it initializes the DDR3 memory chip.
- Once initialization is complete, it services the following incoming DDR1 commands from the DS34S132 by converting to DDR3 commands/data.
    i. Read
    ii. Write

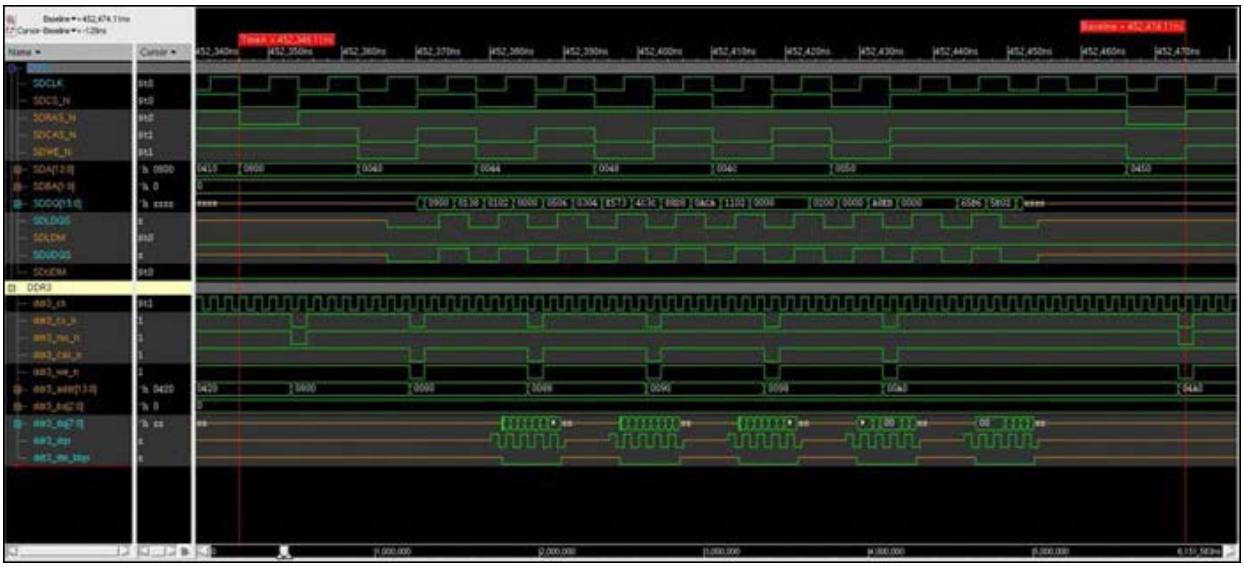
iii. Precharge
iv. Use a digital clock module (DCM) to multiply the incoming DDR1 clock (125MHz) from the DS34S132 by 4 to generate the DDR3 clock (500MHz).
v. Use a second DCM to generate four phases of the DDR3 clock for edge timing to the DDR3. Additionally a feedback clock (out and back of an IO pad) will optimally adjust the phase relationship of these clocks to the DDR1 clock edge. This would help read data timing back to the DS34S132.

Next, the FPGA and the DDR3 memory combined will appear like a DDR1 memory for the DS34S132. The FPGA code includes DCMs and IO DDR primitives. Please download these codes and the specifications from this link here.

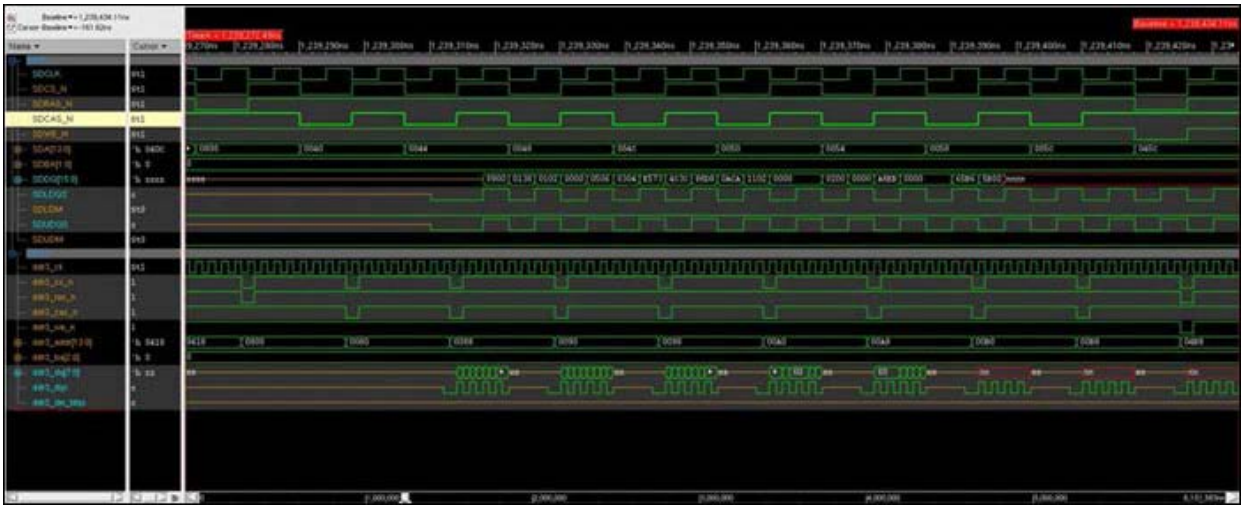
The FPGA used is Spartan for Verilog RTL simulation has a speed grade of -4, which should be capable of 500MHz. FPGA resources use approximately 300 flip flops with no FPGA RAM being used. We also used two DCMs to convert clock rates. To verify the operation of the DDR1-to-DDR3 conversion scheme, we ran the following tests in Verilog RTL simulations:

- Run test in DDR1 mode and log results.
- Run test in DDR3 mode via FPGA and log results.
- Monitor some DDR3 writes and read individually verified using monitor against DDR1 writes and read. **Figure 2** and **Figure 3** show the write and read simulation results of DDR1 and DDR3.

In this Verilog RTL simulation, we could successfully send the signal from DDR3 through FPGA to the device effectively. We also successfully read the signal from the device to DDR3. Thus, the simulation result proved that this conversion scheme from DDR1 to DDR3 test operates correctly.



More detailed image (PDF, 2.3MB)
Figure 2. The write simulation results of DDR1 (top) and DDR3 (bottom).



More detailed image (PDF, 2.1MB)

Figure 3. The read simulation results of DDR1 (top) and DDR3 (bottom).

Conclusion

We implemented a Verilog RTL simulation of FPGA with the Micron DDR3 MT41J128M8 parameter model and are confident that it will work with the DDR3 and the Spartan FPGA. However, this solution will not be successful for all DDR3s, and knowing which DDR3 to implement is the designer's responsibility. FPGA mapping was not implemented and bit file was not used.

For further information on TDMoP devices or other Maxim telecom products, please visit our contact the Communication and Timing Products applications support team.

Verilog is a registered trademark of Gateway Design Automation Corporation.

Related Parts		
DS34S132	32-Port TDM-over-Packet IC	Free Samples

More Information
For Technical Support: http://www.maximintegrated.com/support
For Samples: http://www.maximintegrated.com/samples
Other Questions and Comments: http://www.maximintegrated.com/contact

Application Note 5120: http://www.maximintegrated.com/an5120
APPLICATION NOTE 5120, AN5120, AN 5120, APP5120, Appnote5120, Appnote 5120