Booting from the Serial EEPROM for the First Time in Satellite Mode

Application Note AN-361

Notes

By Fred Santilo

Revision History

February 11, 2002: Initial publication.

October 18, 2002: Changed device designation to RC3233x which includes 3 devices: RC32334, RC32333, and RC32332.

Background

The RC3233x series are integrated processors based on the RC32300 CPU core. They incorporate a high-performance, low cost 32-bit CPU core with functionality common to a large number of embedded applications.

This application note describes the initial programming of the Serial EEPROM when the RC3233x devices are configured to boot from the Serial EEPROM in Satellite mode. Additional information and example code pertaining to the PCI Satellite Mode can be found in IDT's Application Note AN-08 and the RC3233x User Reference Manual.

RC3233x

The RC3233x PCI bus interface can operate in either a Host mode or a Satellite mode, with the Satellite mode being able to boot from the Memory Controller or the PCI Serial EEPROM. When booting from the Serial EEPROM, the RC3233x loads its PCI Configuration registers with values stored in the Serial EEPROM. After its Bus Master Enable bit is set to a one, the RC3233x will access the external device at PCI address 0x0FC0 0000 to obtain its bootstrap code. The Serial EEPROM memory mapping is described in Table 12.3 of the 79RC3233x User Reference Manual.

The first time the RC3233x boots from the Serial EEPROM (in Satellite mode), it will read random information from the Serial EEPROM, since the Serial EEPROM has not yet been loaded with the desired register values. Thus, the RC3233x Configuration registers will not contain valid register values and this may cause unpredictable behavior.

There are two methods that can be used to load the Serial EEPROM with valid register values. Upon loading the Serial EEPROM, the RC3233x can be reset, which will enable the device to properly load its PCI Configuration registers from the Serial EEPROM with valid values.

Normal Satellite Mode Boot Initialization Sequence

The normal Satellite boot initialization sequence from the PCI Serial EEPROM assumes the Serial EEPROM has been preloaded with the desired PCI Configuration register values, which excludes the first time boot initialization. Refer to Figure 1 which illustrates a typical RC3233x system application block diagram.

The initialization steps are as follows:

1. Power is applied to the system.

2. The RC3233x PCI Bridge loads its PCI Configuration registers with the pre-loaded register contents contained in the Serial EEPROM.

3. The PCI Host device detects and appropriately configures all the devices connected to the PCI interface. Configuration is based on the value of the Device ID field of each devices PCI Configuration register.

Notes

4. The PCI Host writes a one to the Bus Master Enable bit of the RC3233x.

5. The RC3233x CPU issues an instruction fetch cycle across the PCI interface to the external device with PCI address 0x0FC0 0000 to obtain its bootstrap code.

6. System initialization is complete.

Method 1

Method 1 uses a PCI Host device to directly write the register values to the Serial EEPROM the first time the system is initialized. Refer to Figure 1 which illustrates a typical RC3233x system application block diagram.

Method 1 initialization sequence for the first time boot from the Serial EEPROM in Satellite mode is as follows:

1. Power is applied to the system.

2. The RC3233x PCI Bridge loads its PCI Configuration registers with the contents of the Serial EEPROM, which at this time are not valid register values. Once the PCI Bridge completes the loading of its PCI Configuration registers, the RC3233x CPU will try and access its bootstrap code residing in the external device at PCI address 0x0FC0 0000. However, it will be denied access to the PCI interface until the Bus Master Enable bit of the PCI Command register is set to a one by the PCI Host device.

3. The PCI Host device detects and appropriately configures all the devices connected to the PCI interface. Configuration of each device is normally based on the value of the Device ID field of its PCI Configuration register. However, the RC3233x must be detected by the PCI slot it resides in since its Device ID field contains a random value.

4. The PCI Host writes a dummy instruction routine, for the CPU, in the external device at PCI address 0x0FC0 0000. The dummy instruction routine instructs the CPU to go into a dead lock state. The dead lock state can be accomplished using the Jump command, which would put the CPU in an endless loop routine.

5. The PCI Host device writes to the Bus Master Enable bit of the RC3233x allowing the CPU to fetch the dummy instruction routine that it has loaded in the external device at PCI address 0x0FC0 0000.

6. Once the CPU is in the dead lock state the PCI Host writes the desired valid register values to the EEPROM via the appropriate PIO registers associated with the Serial Peripheral Interface. A detailed description of this code can be found in IDT Application Note AN-08.

7. The PCI Host clears the dummy instruction routine from the external device at PCI address 0x0FC0 0000 and replaces it with the actual RC3233x bootstrap code.

8. The RC3233x is issued a hard reset so the PCI Bridge will read the desired register values contained in the Serial EEPROM and load the values into its PCI Configuration registers. This can be accomplished either by resetting the entire system, or by issuing a hard reset to the RC3233x.

9. The RC3233x PCI Bridge loads its PCI Configuration registers with the contents contained in the Serial EEPROM.

10. The PCI Host device detects and configures all the devices connected to the PCI interface. Configuration is based on the Device ID field of each devices PCI Configuration register.

11. The PCI Host writes a one to the Bus Master Enable bit of the RC3233x.

12. The RC3233x CPU issues an instruction fetch cycle across the PCI interface to the external device at PCI address 0x0FC0 0000 to obtain its bootstrap code.

13. System initialization is complete.

Method 2

Method 2 uses a PCI Host device to instruct the CPU to load the Serial EEPROM with the desired valid register values the first time the system is initialized. Refer to Figure 1 which illustrates a typical RC3233x system application block diagram.

Notes

Method 2 initialization sequence for the first time boot from the Serial EEPROM in Satellite mode is as follows:

1. Power is applied to the system.

2. The RC3233x PCI Bridge loads its PCI Configuration registers with the contents of the Serial EEPROM which at this time are not valid register values.

3. The PCI Host device detects and appropriately configures all the devices connected to the PCI interface. Configuration of each device is normally based on the value of the Device ID field of its PCI Configuration register. However, the RC3233x must be detected by the PCI slot it resides in since its Device ID field contains a random value.

4. The PCI Host loads a dummy instruction routine for the CPU into the external device at PCI address 0x0FC0 0000. The dummy instruction routine instructs the CPU to load the desired register values in the Serial EEPROM and then go into a dead lock state.

5. The PCI Host device writes to the Bus Master Enable bit of the RC3233x allowing the CPU to fetch the dummy instruction routine residing at PCI address 0x0FC0 0000.

6. The CPU fetches the dummy instruction routine and loads the Serial EEPROM with the desired register values. Once the Serial EEPROM is loaded the CPU goes into a dead lock state.

8. Once the CPU goes into the dead lock state, the PCI Host clears the dummy instruction routine from the external device at PCI address 0x0FC0 0000 and replaces it with the actual RC3233x bootstrap code.

9. The RC3233x is issued a hard reset so the PCI Bridge will read the desired register values contained in the Serial EEPROM and load the values into its PCI Configuration registers. This can be accomplished either by resetting the entire system or by issuing a hard reset to the RC3233x.

10. The RC3233x PCI Bridge loads its PCI Configuration registers with the contents contained in the Serial EEPROM.

11. The PCI Host device detects and configures all the devices connected to the PCI interface. Configuration is based on the Device ID field of each devices PCI Configuration register.

12. The PCI Host writes a one to the Bus Master Enable bit of the RC3233x.

13. The RC3233x CPU issues an instruction fetch cycle across the PCI interface to the external device at PCI address 0x0FC0 0000 to obtain its bootstrap code.

14. System initialization is complete.

Notes

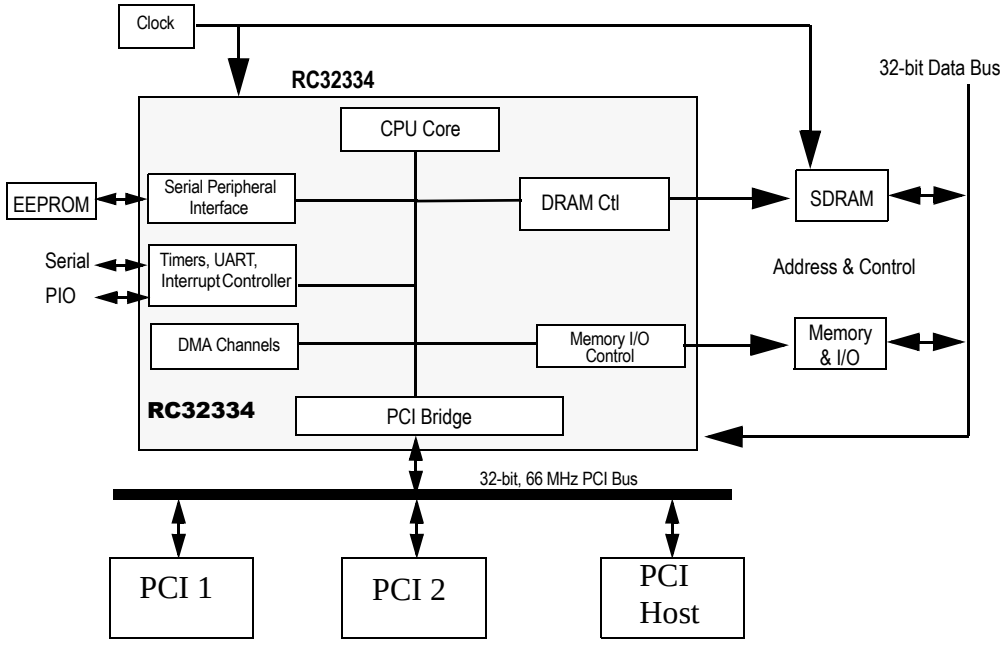

Figure 1 Typical RC32334 System Application

Conclusion

Booting the RC3233x in satellite mode from the Serial EEPROM for the first time is easily accomplished using one of two methods:

Method 1 — the PCI Host device directly writes the RC3233x PCI Configuration register values to the Serial EEPROM

Method 2 — the PCI Host instructs the RC3233x CPU to load the correct PCI Configuration register values.

References

RC3233x User Reference Manual

IDT MIPS Microprocessor Family Software Reference Manual

RC32334 Data Sheet, RC32333 Data Sheet, and RC32332 Data Sheet

IDT Application Note AN-08

IMPORTANT NOTICE AND DISCLAIMER

RENESAS ELECTRONICS CORPORATION AND ITS SUBSIDIARIES (“RENESAS”) PROVIDES TECHNICAL SPECIFICATIONS AND RELIABILITY DATA (INCLUDING DATASHEETS), DESIGN RESOURCES (INCLUDING REFERENCE DESIGNS), APPLICATION OR OTHER DESIGN ADVICE, WEB TOOLS, SAFETY INFORMATION, AND OTHER RESOURCES “AS IS” AND WITH ALL FAULTS, AND DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING, WITHOUT LIMITATION, ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT OF THIRD-PARTY INTELLECTUAL PROPERTY RIGHTS.

These resources are intended for developers who are designing with Renesas products. You are solely responsible for (1) selecting the appropriate products for your application, (2) designing, validating, and testing your application, and (3) ensuring your application meets applicable standards, and any other safety, security, or other requirements. These resources are subject to change without notice. Renesas grants you permission to use these resources only to develop an application that uses Renesas products. Other reproduction or use of these resources is strictly prohibited. No license is granted to any other Renesas intellectual property or to any third-party intellectual property. Renesas disclaims responsibility for, and you will fully indemnify Renesas and its representatives against, any claims, damages, costs, losses, or liabilities arising from your use of these resources. Renesas' products are provided only subject to Renesas' Terms and Conditions of Sale or other applicable terms agreed to in writing. No use of any Renesas resources expands or otherwise alters any applicable warranties or warranty disclaimers for these products.

(Disclaimer Rev.1.01)

Corporate Headquarters

TOYOSU FORESIA, 3-2-24 Toyosu,
Koto-ku, Tokyo 135-0061, Japan
www.renesas.com

Contact Information

For further information on a product, technology, the most up-to-date version of a document, or your nearest sales office, please visit www.renesas.com/contact-us/.

Trademarks

Renesas and the Renesas logo are trademarks of Renesas Electronics Corporation. All trademarks and registered trademarks are the property of their respective owners.