

Multiboot for SOC Modules

API Specification

Revision: 1.0

2016.08.08

© 2016 SOC Technologies Inc.

SOC is disclosing this user manual (the "Documentation") to you solely for use in the development of designs to operate with SOC hardware devices. You may not reproduce, distribute, republish, download, display, post, or transmit the Documentation in any form or by any means including, but not limited to, electronic, mechanical, photocopying, recording, or otherwise, without the prior written consent of SOC. SOC expressly disclaims any liability arising out of your use of the Documentation. SOC reserves the right, at its sole discretion, to change the Documentation without notice at any time. SOC assumes no obligation to correct any errors contained in the Documentation, or to advise you of any corrections or updates. SOC expressly disclaims any liability in connection with technical support or assistance that may be provided to you in connection with the Information.

THE DOCUMENTATION IS DISCLOSED TO YOU "AS-IS" WITH NO WARRANTY OF ANY KIND. SOC MAKES NO OTHER WARRANTIES, WHETHER EXPRESSED, IMPLIED, OR STATUTORY, REGARDING THE DOCUMENTATION, INCLUDING ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NONINFRINGEMENT OF THIRD-PARTY RIGHTS. IN NO EVENT WILL SOC BE LIABLE FOR ANY CONSEQUENTIAL, INDIRECT, EXEMPLARY, SPECIAL, OR INCIDENTAL DAMAGES, INCLUDING ANY LOSS OF DATA OR LOST PROFITS, ARISING FROM YOUR USE OF THE DOCUMENTATION.

© 2008-2016 SOC, Inc. All rights reserved.

SOC, the SOC logo, the Brand Window, and other designated brands included herein are trademarks of SOC, Inc.

Revision History

The following table shows the revision history for this document.

Date	Comment	Revision	Author
08/08/2016	First Revision	1.0	Chad
09/01/2016	Auto Erase Feature + Error Registers, Update Process	1.1	Chad

Table-1: Multiboot Registers

The multiboot process allows for updating the FPGA image via the networks stack or other transmission interface.

Addr (HEX)	Information	Access
A2	*Start Update Command = h1234 Bit[31]: Update Complete Bit[30]: Erase Done Bit[29]: Programming Mode Active Bit[28]: Erase Mode Active Bit[27]: Image Check Failed (errors detected) Bit[26]: Image Check Complete Bit[25]: Receiving Update Bit[23:16]: Erase Location (FF=last address) Bit[23:12]: Programming Location (FFF=last address) Bit[10:8]: Error Code <ul style="list-style-type: none"> - 5 = Update Stalled (No end of flash Packet) - 4 = Flash Start Error (Corrupt/missing Start of Flash) - 3 = Buffer Overflow (Tx too fast) - 2 = CRC Error (Corrupt packets received) - 1 = Flash not ready (Rx when erasing or update process not started/running) - 0 = No Error detected Bit[7:0]: Multiboot CRC error count	R/W*
FD	Bit[30:28]: Error Code <ul style="list-style-type: none"> - 5 = Update Stalled (No end of flash Packet) - 4 = Flash Start Error (Corrupt/missing Start of Flash) - 3 = Buffer Overflow (Tx too fast) - 2 = CRC Error (Corrupt packets received) - 1 = Flash not ready (Rx when erasing or update process not started/running) - 0 = No Error detected Bit[27:0]: Rx Update Byte Counter (Reset when update processes invoked)	R

Multiboot Registers

Update Sequence:

1. **Invoke the Erase+Update command**
 - a. **Write h1234 to 0x12**
 - b. **Module will indicated it is in receiving mode**

2. Send Update image

- a. Use recommended packet size and packet interval suggested by SOC
 - i. Failure to do so will introduce errors and corrupt the update process
- b. Use 0xFD to ensure bytes are being processed and received.
 - i. 0xFD[27:0] will show the exact byte count of the Update image

3. Image Receive Complete

- a. The flash will only be erased when a complete error free image is received.
 - i. Erasing: 0x12-Bit[28] = '1'
 - ii. Image Check complete: 0x12[26]='1'
- b. Erase Complete 0x12-Bit[30] = '1'

4. Programming

- a. After erasing is complete the flash will now be programmed with the updated image

5. Check for successful Update

- a. Update Done: 0x12-Bit[31]='1'
- b. Ensure that no errors have been detected
 - i. Error Type: 0x12-Bit[10:8]=0
 - ii. Error Count: 0x12-Bit[7:0]=0

6. No Errors: Power Cycle the device and the update image will be booted

7. Errors:

- a. The FPGA will not modify the flash if errors are encountered
- b. Restart the Update process (Step 1)