ALPHA DATA

ADB3 Driver 1.1.2 for Linux Release Note

Introduction

This release note accompanies the ADB3 Driver for Linux. The latest version of this driver can be found at:
ftp://ftp.alpha-data.com/pub/admxrcg3/linux
For support, send e-mail to **support@alpha-data.com**

Operating systems supported

This release of the ADB3 Driver supports the following operating systems:

- GNU/Linux distribution with 2.6.x kernel

Due to the ever-changing nature of GNU/Linux, Alpha Data cannot guarantee that this driver can be successfully configured, built, installed and run on all Linux distributions past, present and future. Alpha Data makes best-efforts to ensure compatibility with all Linux distributions, but should a problem be encountered, please contact **support@alpha-data.com**.

Hardware supported

This release of the ADB3 Driver supports the following Alpha Data hardware:

- ADM-XRC-6TL
- ADM-XRC-6T1

License Agreement

This release of software is licensed according to the terms of GNU Public License Version 2 (GPL V2). A copy of this license can be found in the file **gpl-2.0.txt** within this software package. Please contact Alpha Data if alternative licensing conditions are required.

Alpha Data reserves the right to use a different license agreement for future releases of this software.

Installation instructions

This release of the driver is distributed in source code form as a tarball (.tar.gz file extention). Please refer to the README file inside the tarball for instructions on how to configure, build and install the driver.

VPD write-protection mechanism

The VPD write-protection mechanism described in the ADM-XRC Gen 3 SDK User Guide is implemented as of release 1.1.0. To enable write-to-VPD, the kernel module parameter **EnableVpdWrite** must be nonzero. This value is set when the driver starts, so it can be changed only by unloading the driver and restarting the driver with a different value for **EnableVpdWrite**. If this parameter is not specified, the driver considers it to be zero (write-to-VPD disabled).

Known issues

Security considerations

As of release 1.1.0, the script adb3.rc that creates device nodes in /dev applies permissions 666 (previous version of the driver applied permissions 600). When a user-mode process attempts to open a device, the driver verifies that either (a) the user-mode process is opening the device in "read only" mode, or (b) the user owning the process is a system administrator. If neither are true, the driver rejects the attempt to open the device. In the case of (a), the driver restricts the user-mode process to using a subset of the device's functionality so that the device does not compromise system security.

Partial udev support

In this release of the driver, a script adb3.rc is provided in order to start the driver. This creates device nodes in /dev rather than relying on udev rules. This will be corrected in a future release.

Downgrading to an earlier version

When downgrading to an earlier version of the driver, remove all files named /usr/lib/libadmxrc3.*, (and /usr/lib64/libadmxrc3.* if on a 64-bit architecture machine), before executing the **make install** command as root. Otherwise, the shared libraries remaining from the later version of the driver will be preferred by the system as they have a higher version number.

When downgrading to a driver earlier than 1.1.0, delete any device nodes /dev/admxrc3* before installing the earlier driver, as a safety precaution. This is recommended because a pre-1.1.0 ADB3 driver relies upon the device nodes being owned by root with permissions 600 in order to restrict access to privileged users.

Fixed-local addressing DMA transfers

The flag ADMXRC3_DMA_FIXEDLOCAL currently has no effect for the ADM-XRC-6TL and ADM-XRC-6T1 when using the DMA functions in the ADMXRC3 API.

Release history

Release 1.1.2

Corrections:

1. Fixed a user-mode memory leak that can occur when the API functions **ADMXRC3_LoadBitstreamA** and **ADMXRC3_LoadBitstreamW** return a failure status code.

2. Fixed a bug in **ADMXRC3_SetClockFrequency** where on failure, the wrong status code was returned.

Release 1.1.0

Corrections:

1. Fixed a potential memory corruption issue on some architectures due to incorrect byte size calculation in device context structure.

2. Fixed a memory leak related to CFI Flash functionality when stopping and restarting the driver.

3. Fixed user VPD area (VPD space address range 0x100000-0x1FFFFF) not being accessible on ADM-XRC-6TL and ADM-XRC-6T1.

4. Corrected error codes returned by several ADMXRC3 API functions when invalid parameters are passed:

    1. Non-Locked DMA functions now return 'ADMXRC3_INVALID_BUFFER' if the 'pBuffer' and/or 'length' parameters are invalid.

    2. Locked DMA functions now return 'ADMXRC3_INVALID_BUFFER_HANDLE' if the 'hBuffer' parameter is invalid.

    3. ADMXRC3_Unlock now returns 'ADMXRC3_INVALID_BUFFER_HANDLE' if the 'hBuffer' parameter is invalid.

5. Fixed 'ADMXRC3_Unconfigure' failing with 'ADMXRC3_NOT_OWNER' even when the target FPGA has no owner.

6. Fixed the ADMXRC3 DMA functions not properly co-validating the 'local' and 'length' parameters.

7. Fixed the ADMXRC3 Locked DMA functions not properly co-validating the 'offset' and 'length' parameters.

8. Fixed the ADMXRC3 nonblocking DMA functions incorrectly returning 'ADMXRC3_SUCCESS' instead of 'ADMXRC3_PENDING' when 'length' is zero and all other parameters are OK.

9. Fixed the ADMXRC3 non-Locked DMA functions not properly co-validating the 'pBuffer' and 'length' parameters.

10. Fixed certain ADMXRC3 API functions not trapping NULL pointers being passed, typically resulting in an application crash.

11. Fixed a bug in driver parameter handling so that driver parameters are now correctly applied and have correct default values.

12. Implemented VPD write-protection mechanism (now protected by default).

13. Added workaround for 4k crossing issue in ADB3 Bridge rev 0x01 and earlier. Workaround is not applied for rev 0x02 or later.

Enhancements:

14. Added support for ADM-XRC-6T1.

15. Added API function ADMXRC3_OpenEx, which allows an unprivileged process to open a device in "read only" mode and use a subset of the ADMXRC3 API functions.

16. Added API functions ADMXRC3_StartNotificationWait and ADMXRC3_FinishNotificationWait, which allow Linux applications to wait for events (since there is no Linux equivalent of ADMXRC3_RegisterWin32Event).

17. Added API function ADMXRC3_GetCardInfoEx (with structure ADMXRC3_CARD_INFOEX), which returns a superset of data supplied by ADMXRC3_GetCardInfoEx.

18. Added diagnostic API functions ADMXRC3_GetSensorInfo and ADMXRC3_ReadSensor, with associated types and structures. These functions allow applications to monitor the health of a Gen 3 reconfigurable computing card.

Release 1.0.0

This is the first release of the ADB3 Driver for Linux.

ALPHA DATA

Page Intentionally left blank.