



ADB3 Driver 1.2.0 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 extension). 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

This release features a new permissions mechanism for controlling access to devices. By default, the script **adb3.rc** creates device nodes in **/dev** as follows:

- Mode: 660 (octal) => owner read, owner write, group read, group write
- UID: same as whoever runs the script (**root**)
- GID: same as whoever runs the script (**root**)

This means that the default permissions are that only **root** and members of the group **root** can open devices, in read-only or read-write mode. However, the **adb3.rc** script can now be customized to relax permissions. See the comments in that file for details.

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/libadmxcrc3.*** and **/usr/lib/libadb3.***, (and **/usr/lib64/libadmxcrc3.*** and **/usr/lib64/libadb3.*** if on a 64-bit bi-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.

Additionally, delete any device nodes **/dev/admxcrc3*** before installing the earlier driver version, as a safety precaution. This is recommended because earlier versions of the driver have different permissions mechanisms when opening devices.

Fixed-local addressing DMA transfers

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

Release history

Release 1.2.0

This release implements ADMXRC3 API Specification version 1.2.0.

New behavior:

1. Bi-architecture build is now not performed by default, as most 64-bit Linux distributions do not install (by default) the necessary compatibility packages for building 32-bit binaries. To build both 32-bit and 64-bit binaries for the API libraries, specify '-biarch yes' when running the 'configure' script.

Corrections:

2. Fixed some build problems relating to kernel features in RHEL / CentOS 5.5 kernels (derived from Linux 2.6.18). The 'configure' script now detects those kernel features correctly.
3. Fixed a thinking issue that may occur if the 32-bit and 64-bit C compilers on the same bi-architecture-capable system have different struct packing rules, where certain ADMXRC3 API calls made by a 32-bit executable running under a 64-bit Linux kernel fail with **ADMXRC3_UNKNOWN_ERROR**.
4. Fixed a problem where, when a process closes a device handle that has ongoing non-blocking operations, the process may crash due to incorrect cleanup being performed by the driver.

Enhancements:

- Added new API functions for performing DMA transfers with 64-bit local addresses:
 - [ADMXRC3_ReadDMAEx](#)
 - [ADMXRC3_ReadDMALockedEx](#)
 - [ADMXRC3_StartReadDMAEx](#)
 - [ADMXRC3_StartReadDMALockedEx](#)
 - [ADMXRC3_StartWriteDMAEx](#)
 - [ADMXRC3_StartWriteDMALockedEx](#)
 - [ADMXRC3_WriteDMAEx](#)
 - [ADMXRC3_WriteDMALockedEx](#)
- Added support for new sensors in ADM-XRC-6TL and ADM-XRC-6T1 with firmware 1.4 or later. This provides additional sensors that show internal temperature and voltages in the PCI Express® to OCP Bridge.
- Changed the way that permissions are checked on opening a device to allow for a more flexible system that can be customized via the `adb3.rc` script. See the section "[Security considerations](#)" and the comments in the file `driver/adb3.rc` for details.
- Made the driver's build system more compatible with cross-compilation, now using the same ARCH and CROSS_COMPILE environment variables as the Linux kernel. See the README file for details.

Release 1.1.2

Corrections:

- Fixed a user-mode memory leak that can occur when the API functions [ADMXRC3_LoadBitstream](#) and [ADMXRC3_LoadBitstreamW](#) return a failure status code.
- Fixed a kernel-mode memory leak that can occur when a device handle is closed when at least one non-blocking operation has not been finished.
- Fixed a bug in [ADMXRC3_SetClockFrequency](#) where on failure, the wrong status code was returned.

Release 1.1.0

Corrections:

- Fixed a potential memory corruption issue on some architectures due to incorrect byte size calculation in device context structure.
- Fixed a memory leak related to CFI Flash functionality when stopping and restarting the driver.
- Fixed user VPD area (VPD space address range 0x100000-0x1FFFFFF) not being accessible on ADM-XRC-6TL and ADM-XRC-6T1.
- Corrected error codes returned by several ADMXRC3 API functions when invalid parameters are passed:
 - Non-Locked DMA functions now return 'ADMXRC3_INVALID_BUFFER' if the 'pBuffer' and/or 'length' parameters are invalid.
 - Locked DMA functions now return 'ADMXRC3_INVALID_BUFFER_HANDLE' if the 'hBuffer' parameter is invalid.
 - ADMXRC3_Unlock now returns 'ADMXRC3_INVALID_BUFFER_HANDLE' if the 'hBuffer' parameter is invalid.
- Fixed 'ADMXRC3_Unconfigure' failing with 'ADMXRC3_NOT_OWNER' even when the target FPGA has no owner.

- Fixed the ADMXRC3 DMA functions not properly co-validating the 'local' and 'length' parameters.
- Fixed the ADMXRC3 Locked DMA functions not properly co-validating the 'offset' and 'length' parameters.
- Fixed the ADMXRC3 nonblocking DMA functions incorrectly returning 'ADMXRC3_SUCCESS' instead of 'ADMXRC3_PENDING' when 'length' is zero and all other parameters are OK.
- Fixed the ADMXRC3 non-Locked DMA functions not properly co-validating the 'pBuffer' and 'length' parameters.
- Fixed certain ADMXRC3 API functions not trapping NULL pointers being passed, typically resulting in an application crash.
- Fixed a bug in driver parameter handling so that driver parameters are now correctly applied and have correct default values.
- Implemented VPD write-protection mechanism (now protected by default).
- Added workaround for 4k crossing issue in ADB3 Bridge rev 0x01 and earlier. Workaround is not applied for rev 0x02 or later.

Enhancements:

- Added support for ADM-XRC-6T1.
- 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.
- 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).
- Added API function ADMXRC3_GetCardInfoEx (with structure ADMXRC3_CARD_INFOEX), which returns a superset of data supplied by ADMXRC3_GetCardInfoEx.
- 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.