

Aardvark Embedded Solutions

PayLink Linux User Guide

Issue	Date	Author	Comments
D1	19/04/2006	Len Meakin	Initial release for review.
D2	30/04/2006	Len Meakin	Improved installation script.
1.1	09/10/2008	Dave Bush	Addition of Firmware Loader

Table of Contents

1	Installation	3
2	Required System Libraries	4
3	Automatic Installation	5
4	Manual Installation	6
4.1	USB Driver Executable	6
4.2	AES Shared Library (DLL)	6
4.3	AES FTDI Static Library	6
4.4	FTDI Shared Library	7
5	AESWDriver	8
6	Coin Read.....	11
7	Pay Out	11
8	Lumina Serial Number	11
9	AESDemo	12
10	Distribution Recommendations.....	14
10.1	General Use.....	14
10.2	Advanced Use	14

1 Installation

The installation is to be performed by the `root` user (super-user). All instructions that follow will not execute correctly if issued as a non `root` user.

If you have a previous installation then please rename the existing `LinuxPayLink` directory:

```
# mv LinuxPayLink/ LinuxPayLinkOLD/
```

Copy the provided `LinuxPayLink.tar.gz` archive to `root`'s home directory:

```
# cp -v LinuxPayLink.tar.gz /root/  
# cd /root
```

Decompress the archive to the current directory (`/root`):

```
# tar -zxvf LinuxPayLink.tar.gz
```

Change directory to the newly create `LinuxPayLink` directory:

```
# cd LinuxPayLink
```

2 Required System Libraries

The USB PayLink drivers and libraries have been compiled using gcc version 3.3.5, therefore use version 5 of the standard c++ libraries. This version of the library must also be located on the host system. If you are missing this library a generic version has been compiled and included within the installation archive. In order to install this version follow the following commands:

The following command will search your system for the required library. If this command returns NO results you should install the supplied library:

```
# find / -name libstdc++.so.5*
```

The following commands will copy the required library to the correct location on your system and then create required system symbolic links:

```
# cp ./lib/AESlibstdc++.so.5.0.6 /usr/lib/libstdc++.so.5.0.6  
# ldconfig
```

NOTE: The automatic installation script should take care of this for you.

3 Automatic Installation

For convenience sake an automatic installation script has been provided. This script copies the provided executables / libraries to the correct locations on the system. If the user wishes to perform a development installation, required header files will also be copied. The script also performs various checks on the system to ensure it is configured correctly. Enter the following command to start the installation process:

```
# ./Install.sh
```

Firstly the driver executables are copied and made executable on the host system. Required libraries are then copied to the host system. A check is then performed to ensure the correct version of the standard c++ library is present (libstdc++.so.5). If it is determined the system is missing this library the installation script will copy a generic version of the library to the system (/usr/lib).

A check is then performed to ensure the system has been configured with support for the USB file system. If this support is missing installation will not continue, as this is an essential component. Linux kernel configuration and compilation is required to add this feature if it is missing.

A check is then performed on Fedora Core 5 systems to ensure the USB access library is SELinux approved.

After the required libraries have been installed the user will be prompted as to whether they wish to perform a development installation. Upon the user answering yes [Y/y] the script will perform a basic system check to ensure development tools are present on the host. If development tools are detected the script will start the installation of header files to the system.

This script performs the majority of installation requirements but it is still highly recommended that you continue to read the following "*Manual Installation*" section as this will provide you with further information should you encounter any issues.

4 Manual Installation

The following files should be copied to the given locations on a typical Linux Installation.

4.1 USB Driver Executable

```
/root/LinuxPayLink/AESWDriver      =>    /usr/bin/  
/root/LinuxPayLink/showtraf.sh     =>    /usr/bin/
```

Also ensure the two files are made executable by issuing the following commands:

```
# chmod 6775 /usr/bin/AESWDriver  
# chmod 6775 /usr/bin/showtraf.sh
```

4.2 AES Shared Library (DLL)

```
/root/LinuxPayLink/lib/libaes_access.so.1.0.1  =>    /usr/lib/
```

A symbolic link also requires creating for the shared library using a command sequence similar to that below:

```
# cd /usr/lib/  
# rm libaes_access.so  
# ln -sf libaes_access.so.1.0.1 libaes_access.so  
# ldconfig
```

4.3 AES FTDI Static Library

```
/root/LinuxPayLink/lib/libaes_ftdi.a          =>    /usr/lib/
```

4.4 FTDI Shared Library

This is the shared library supplied by FTDICHIP that provides the same API as the supplied Windows DLL.

```
/root/LinuxPayLink/lib/libftd2xx.so.0.4.10      =>      /usr/lib/
```

A symbolic link also requires creating for the shared library using a command sequence similar to that below:

```
# cd /usr/lib/
# rm libftd2xx.so
# ln -sf libftd2xx.so.0.4.10 libftd2xx.so
# ldconfig
```

This shared library also requires the following lines to be added to the systems `/etc/fstab` file:

```
none /proc/bus/usb      usbfs      defaults,devmode=0666 0      0
```

There have been reports that you may need to use the following command for some distributions (use `usbfs` in 2.6 kernels):

```
none /proc/bus/usb      usbfs      defaults,mode=0666      0      0
```

If you have just added the command then issue the `"mount -a"` command before proceeding any further. This command is not required on subsequent reboots of the system.

If you have problems with this check with `usbview` (<http://www.kroah.com/linux-usb/>) to check the USB file system is mounted properly.

Other problems will be related to the `ftdi_sio` kernel module loading. If this is the case you must unload this kernel module and `usbserial` if they are attached to your device:

```
# rmmod ftdi_sio
# rmmod usbserial
```

Please Note: You must be logged into the system as root to issue the above commands.

If you have an issue with a 2.6 kernel, FTDICHIP have noted some issues with certain kernels of the 2.6 branch. Currently these stand at 2.6.4, 2.6.7, 2.6.8 and 2.6.11. FTDICHIP try to test with as many kernels as practically possible prior to release but if you see an issue that is not one of the above kernels then contact FTDICHIP support and they will test that kernel on site.

Please Note: If you receive an error similar to that below when loading the AESWDriver process (when using SELinux):

```
... cannot restore segment prot after reloc: Permission denied
```

Issue the following command to resolve this issue:

```
# chcon -t texrel_shlib_t /usr/lib/libftd2xx.*
```

4.5 Firmware Updating

The release archive has a folder called Firmware. This folder contains a number of programs each of which will load a new copy of the firmware into Paylink.

The naming convention used in this folder are that:

- the first part is "Genoa", the internal name of the firmware
- the 2nd Part specifies the note reader protocol supported by the RJ45 serial connector, either:
 - The ID003 protocol, originating with JCM
 - The GPT V2.2 Serial protocol
 - The Ardac 2 protocol for the MCL Ardac 2 acceptor.
- a 4th option does not use the RJ45 for a protocol and outputs program diagnostics on the RJ45.
- the final part is v_{n-n-n-n} - the release number of the firmware.

5 AESWDriver

The driver runs in command line mode, no graphical user interface has been developed. However the driver is able to run in various modes:

Verbose mode (-v): certain information is printed to stdout (console window). This is analogical to running the Windows driver with the hide traffic option selected. Without the verbose mode flag set the driver will print very minimal information (if any).

ShowTraffic mode (-t): will display all data that is being sent / received to / from the PayLink interface (all messages are time stamped to millisecond resolution). This is analogical to running the Windows driver with the show traffic option selected. This option is only usually useful when no traffic at all appears to be taking place.

High Priority mode (-p): will cause the driver program to increase the priority of the communication process.

Serial Number Specific mode (-s <Serial>): will cause the driver program to only communicate with a Paylink set with the specific USB serial number (for us on a multiple Paylink installation.)

The ShowTraffic option may be set at driver start-up (-s) or by sending a SIGUSR1 signal to the driver. The driver will toggle the value of the ShowTraffic option upon receiving a SIGUSR1 signal. The supplied `showtraf.sh` shell script will perform this action for you.

The shared memory segment has been named USBDriver (or USBDriver<Serial> if the serial number option is being used), and can be viewed as part of the file system by issuing the following command (this is not recommended and should only be used for diagnosing driver issues).

```
# ls /dev/shm/USBDriver
```

To Start the AESWDriver at the root prompt issue the following command:

```
# AESWDriver -v
```

To force the driver to exit either send the driver a signal (other than SIGUSR1), or press CTRL+C on the console where the driver process was started.

Upon the driver starting it will attempt to open a link to the attached interface, if unsuccessful the driver will continue connection attempts every second until the device has been opened successfully.

6 Firmware updating

The firmware updating program has the new firmware embedded in the program file. When run it will automatically disable the Driver program (if running) and then download the new firmware to the Paylink. If there is any error in the programming, the program exits with a code of to exit with 1 for an error and 0 is everything is OK.

Options available are as follows:

Serial Number Specific mode (-s <Serial>): will cause the program to only update a Paylink set with the specific USB serial number (for us on a multiple Paylink installation.)

Force mode (-f): regardless of the version already programmed, this mode will reprogram the Paylink.

Check mode (-c): If this is set, then the programming will not happen if the Paylink already contains code with the same version number and program checksum.

6.1 Reprogramming Kernel

Firmware versions x-1-10-9 and later contain a reprogramming kernel. This greatly reduces the chances of a failed reprogramming exercise resulting a non-functional Paylink. If the Paylink already contains a reprogramming kernel then this is not re-written during an upgrade, meaning that following failures, the kernel will allow for further attempts.

Even if the Paylink does not already contain a reprogramming kernel, the first 5% of the full download contains the reprogramming kernel and, provided this is successful, subsequent failures during the same download are tolerated as above.

7 Examples

A number of example programs are included in the distribution that can be used / compiled "straight out of the box" to show that the installation is working.

7.1 *Coin Read*

An example CoinRead program has been provided, to compile this program issue the following commands:

```
# cd /root/LinuxPayLink/CoinRead
# make
# ./CoinRead
```

7.2 *Pay Out*

An example PayOut program has been provided, to compile this program issue the following commands:

```
# cd /root/LinuxPayLink/PayOut
# make
# ./PayOut
```

7.3 *Lumina Serial Number*

The Lumina Serial Number program has been provided, to compile this program issue the following commands:

```
# cd /root/LinuxPayLink/LuminaSerialNo
# make
# ./LuminaSerial
```

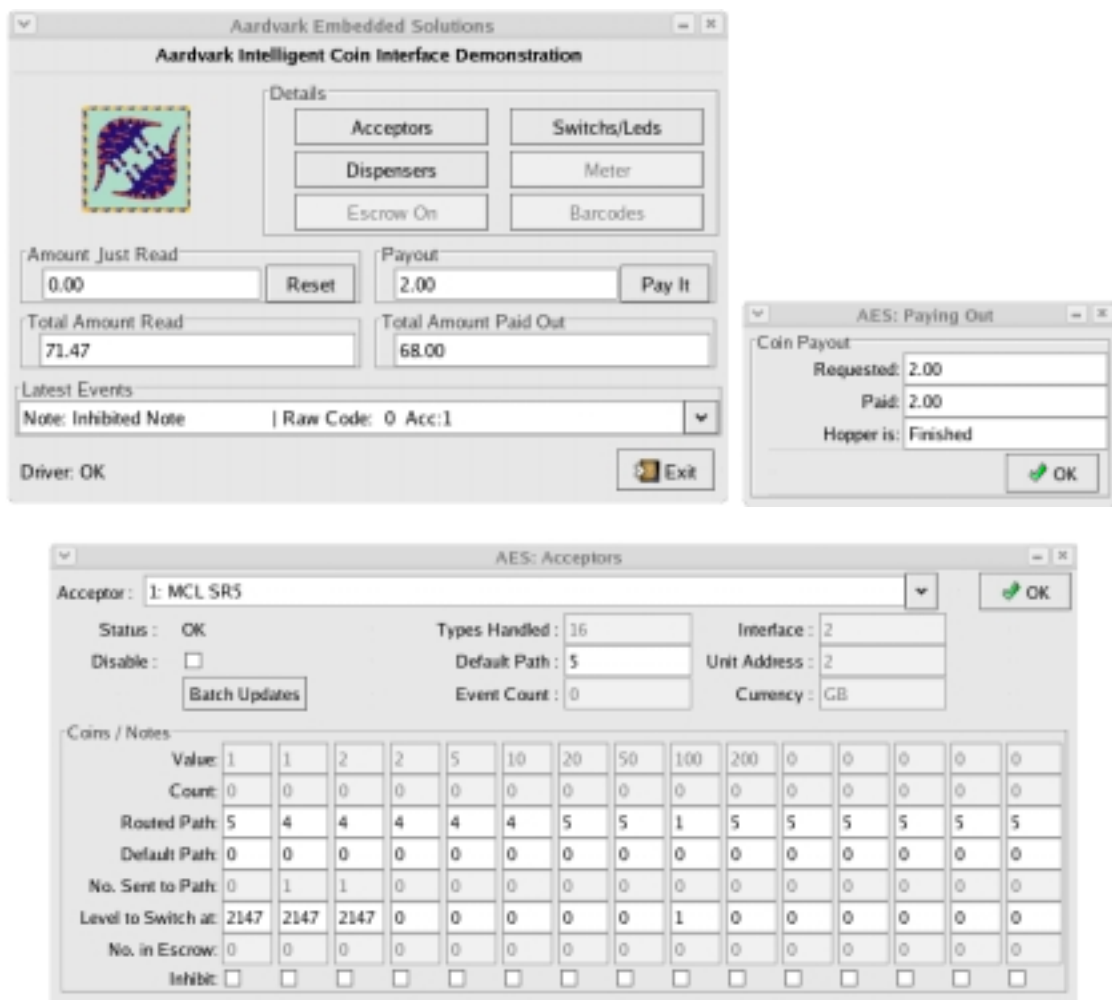
7.4 AESDemo

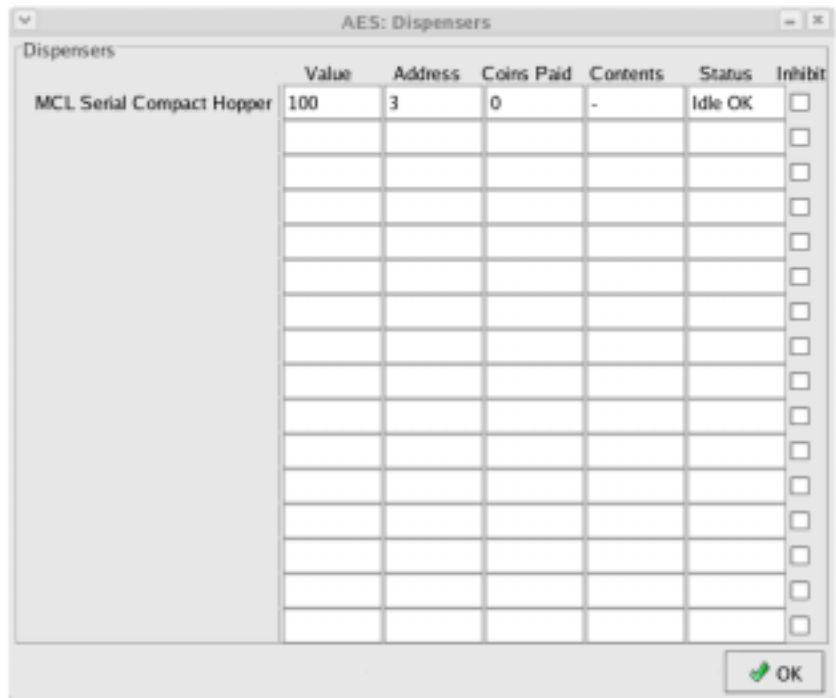
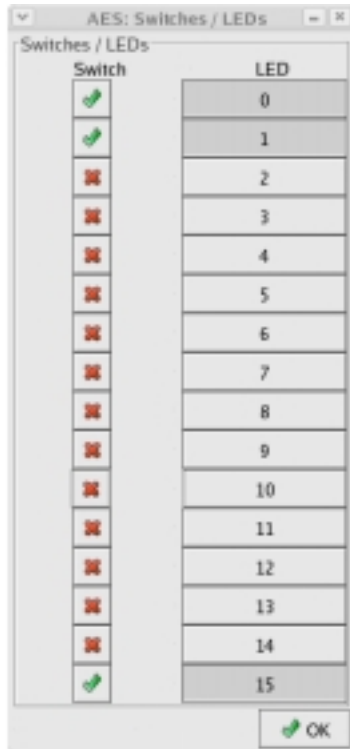
The AES Demo application has been provided, to compile this program issue the following commands:

```
# cd /root/LinuxPayLink/
# tar -zxvf AESDemo.tar.gz
# cd AESDemo
# ./configure
# make
# cd ./src
# ./AESDemo
```

This application uses the gtk+ graphical tool kit library. Please ensure that X Windows is running before issuing the above commands.

The application has been tested with GNOME 2.14 (Gentoo / FC5) and GNOME 2.8 (FC3).





8 Distribution Recommendations

The following Linux distributions are recommended:

8.1 General Use

For ease of installation and use the following Linux distributions are recommended and have been tested:

- [Fedora Core 5](http://fedora.redhat.com/)
<http://fedora.redhat.com/>
- [Fedora Core 3](http://fedora.redhat.com/)
<http://fedora.redhat.com/>

8.2 Advanced Use

For users who are competent installing and using Linux systems the following Linux distributions are recommended:

- [Gentoo Linux \(2006.0\)](http://www.gentoo.org/) (Tested)
<http://www.gentoo.org/>
- [Slackware Linux 10.2 \(2.6 / 2.4 kernel\)](http://www.slackware.org/) (Untested)
<http://www.slackware.org/>