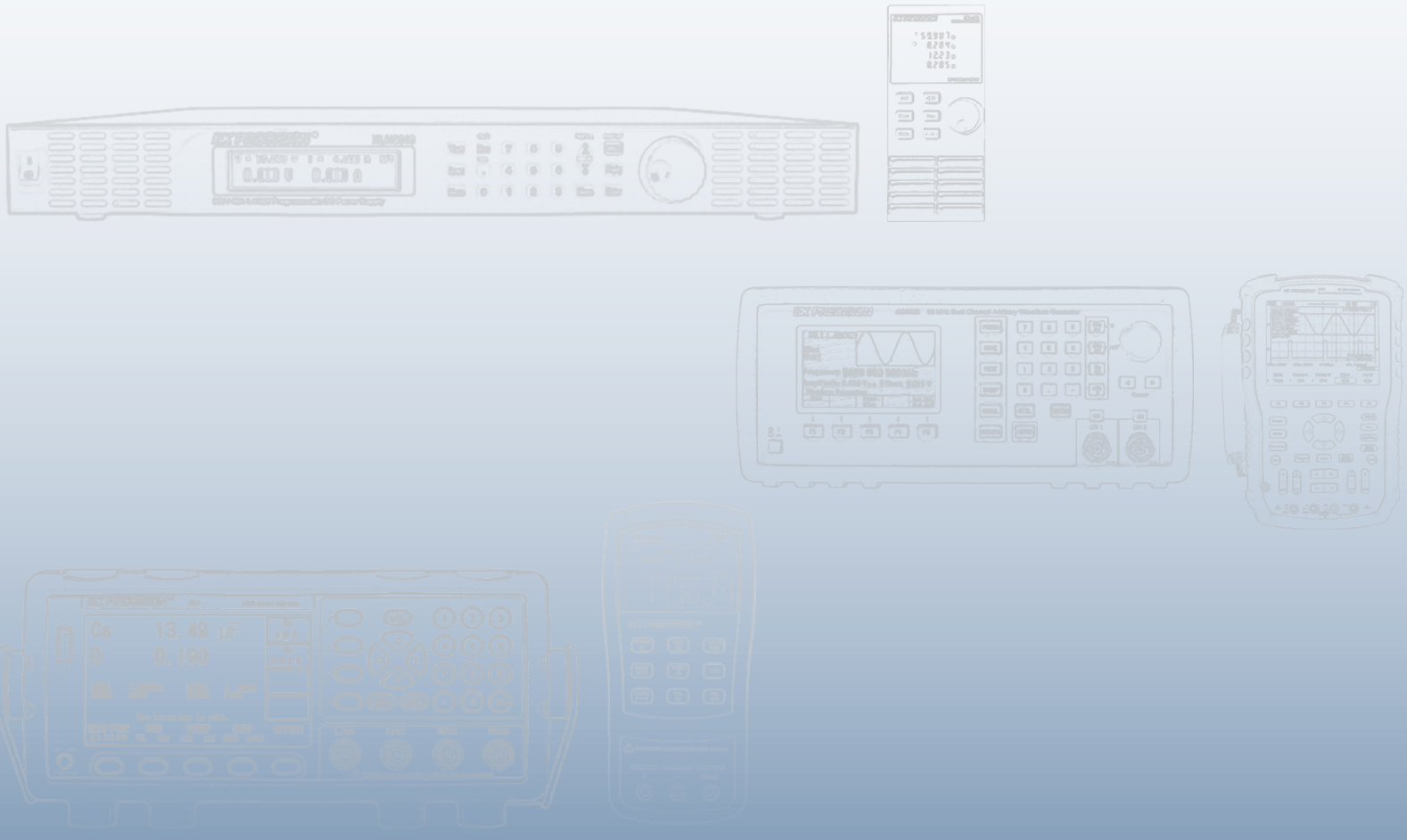
DC Power Supply Operating Software Guide

Version 1.0.0

Copyright (c) 2020-2021

B&K Precision Corporation. All rights reserved.



Contents

1 NI-VISA Drivers 3
1.1 Install NI-VISA for Linux . . . 3
1.2 Configure Permissions . . . 4
1.3 Verify the Installation . . . 5
1.4 Troubleshoot Common Issues . . . 5

2 Installing Prolific Drivers 6
2.1 Checking for Existing Driver Support . . . 6
2.2 Install Required Packages . . . 6
2.3 Installing the Driver . . . 7
2.4 Test the Connection and Troubleshoot . . . 8

3 Ethernet Communication Without VISA 9
3.1 Install Required Packages . . . 9
3.2 Network Configuration . . . 9
3.3 Include Headers . . . 9

NI-VISA Drivers

Establishing communication with a National Instruments (NI) Virtual Instrument Software Architecture (VISA) device on Linux involves several steps, including installing necessary software, configuring the device, and using programming languages or tools that support VISA. Here's a comprehensive guide to help you get started.

1.1 Install NI-VISA for Linux

Step 1. Download NI-VISA:

- Go to the **NI VISA**
- Choose the version compatible with your Linux distribution (e.g., Ubuntu, Fedora).

Step 2. Install Dependencies

- Ensure you have essential build tools and libraries. Open a terminal and run:

```
sudo apt-get install build-essential libtool autoconf pkg-config
```

Step 3. Install NI-VISA

- Extract the downloaded package:

```
tar -xvf ni-visa-<version>.tar.gz
```

```
cd ni-visa-<version>
```

- Follow the included README or INSTALL file for specific installation instructions.
- You may need to run:

```
sudo ./install
```

1.2 Configure Permissions

Step 1. Add User to Groups

- Add your user to the `plugdev` group:

```
sudo usermod -aG plugdev $USER
```

Step 2. Log out and back in for the changes to take effect.

Step 3. Set Up Udev Rules

- Create a udev rule file for your VISA device (replace `<your_device>` with the actual identifier):

```
sudo nano /etc/udev/rules.d/99-ni-visa.rules
```

- Add the following line to grant permission:

```
SUBSYSTEM=="usb", ATTRidVendor=="<vendor_id>", ATTRidProduct=="<product_id>",
MODE="0666"
```

- Reload udev rules:

```
sudo udevadm control –reload-rules
```

```
sudo udevadm trigger
```

1.3 Verify the Installation

Step 1. Check NI-VISA Installation

- Use the `niVisa` command to verify the installation:

  visaFind

- This will list the connected VISA devices.

Step 2. Test with a Simple Script

- Create a simple script to communicate with your device using a programing language or tool that supports VISA.
- Intall Hardware Configuration Utility to verify without having to write a script

  ni-hwcfg-utility

For a detail example on creating scripts in various programming languages rever to the **Interface Connection Guide**

1.4 Troubleshoot Common Issues

Step 1. Device Not Found

- Ensure the device is powered and connected.
- Check USB or other connections.
- Verify that you have the correct permissions.

Step 2. VISA Errors

– Check the syntax of your commands.

– Refer to the device's error codes in its documentation.

Step 3. Library Conflicts

- If you have other VISA libraries (like those from Keysight or Tektronix), ensure there are no conflicts.

Installing Prolific Drivers

Some BK Products require a TTL to serial converter in order to remote control the instrument. This will require installing the PL2303 Prolific driver on a Linux Many distributions include support for the device out of the box, however, if you encounter issues, you might need to install or update the driver manually.

2.1 Checking for Existing Driver Support

Before installing any drivers, check if your Linux distribution already supports the PL2303 chipset.

Step 1. Connect the Device

- Plug in your PL2303 USB-to-serial adapter.

Step 2. Check dmesg

- Open a terminal and run:

```
dmesg | grep ttyUSB
```

- This command should show if the device was recognized and assigned a `/dev/ttyUSBx` device file.

Step 3. List Serial Device

- You can also list connected serial devices:

```
ls /dev/ttyUSB*
```

- If you see entries like `/dev/ttyUSB0`, it means the driver is already loaded.

2.2 Install Required Packages

If your distribution does not recognize the device, you may need to install some packages.

Step 1. Update the System

- Make sure your package list is up to date:

```
sudo apt update
```

Step 2. Install Necessary Packages

- You may need the following packages (install according to your distribution):

```
sudo apt install build-essential linux-headers-$(uname -r)
```

2.3 Installing the Driver

If you need a newer driver or the default one doesn't work:

Step 1. Download the Driver

- Visit the **Prolific website** or look for a reputable source for the driver.
- You can also find the driver in the Linux kernel source.

Step 2. Install the Driver

- If you downloaded a driver package extract the package

```
tar -xvf pl2303_driver.tar.gz
```

```
cd pl2303_driver
```

Step 3. Compile and Install

- Run

```
make
```

```
sudo make install
```

Step 4. Load the Driver

- If the driver is not automatically loaded, you can load it manually:

```
sudo modprobe pl2303
```

Step 5. Check Module Status

- Ensure the module is loaded:

```
lsmod | grep pl2303
```

2.4 Test the Connection and Troubleshoot

Step 1. Connect to the Serial Port

- Use a terminal program like `screen`, `minicom`, or `picocom` to connect to the device:

```
screen /dev/ttyUSB0 9600
```

NOTICE By default majority of the BK Products set baudrate to 9600 by default. Replace **9600** with your desired baud rate in the instrument's settings as well as in the command.

If the device is still not recognized:

Step 2. Check Permissions

- Ensure your user is in the `dialout` group:

```
sudo usermod -a -G dialout $USER
```

- Log out and back in for the changes to take effect.

Step 3. Recheck dmesg

- Look for any errors or issues related to USB or the PL2303:

```
dmesg | grep -i usb
```

Step 4. Consult Logs

- Check system logs for additional errors:

```
journalctl -xe
```

Ethernet Communication Without VISA

If NI-VISA is not an option available for the project, implenting pure VXI-11 or raw socket drivers is an option. This will require installing the packages for the programming language being used.

3.1 Install Required Packages

Step 1. Update your package list

- Open a terminal and run:

  sudo apt update

- Install the required packages
- You need to install `libvxi11-dev` for VXI-11 communication, or socket for socket communication

  sudo apt install libvxi11-dev

- If you are using socket install the module correspoding to the programming language being used. For example, for python:

  pip install socket.py

3.2 Network Configuration

Step 1. Network Configuration

- Ensure that your Linux machine and the instrument are on the same network.
- You may need to set a static IP for the instrument if necessary.
- Use the terminal to ping the instrument's IP address. Replace `<instrument_ip_address>` with the actual IP address of the instrument.

  ping <instrument_ip_address>

3.3 Include Headers

Step 1. Write a simple program to communicate

- Use your preferred text editor to create a file, e.g., `Ethernt Comm Example.c`.

  nano Ethernet_Comm_Example.c

- Include necessary headers and write code
  - #include <vxi11.h> for VXI-11 instruments.
  - #include <sys/socket.h> for socket instruments.
- Sample programs can be found here: **Interface Connection Guide**