



M-MODULE INSTRUMENT DRIVER ARCHITECTURE Maximizing Platform Independence and Portability

INTRODUCTION

C&H’s M-module instrument drivers are designed to maximize platform independence and ease of portability. This document describes the architecture used to achieve this objective and introduces the reader to the terminology used by C&H to describe the various components of M-module software.

It is generally not required for the typical user to fully understand this architecture. C&H will continue to deliver fully functional “plug and play” type M-module drivers that install easily and run in the Microsoft Windows environment with very little effort. The primary target audience of this document is advanced users who are using M-modules across multiple platform interfaces or who are attempting to port a driver to a previously unsupported platform.

ARCHITECTURE

The architecture is simple in that it modularizes the driver, breaking it up into components that provide a separation between platform specific and platform independent elements. The architecture consists of three basic building blocks, a Framework Wrapper, an ANSI-C Source Driver, and a Platform Support Module. The illustration in Figure 1 provides a general overview of the architecture with the three components making up the driver, outlined in the yellow box.

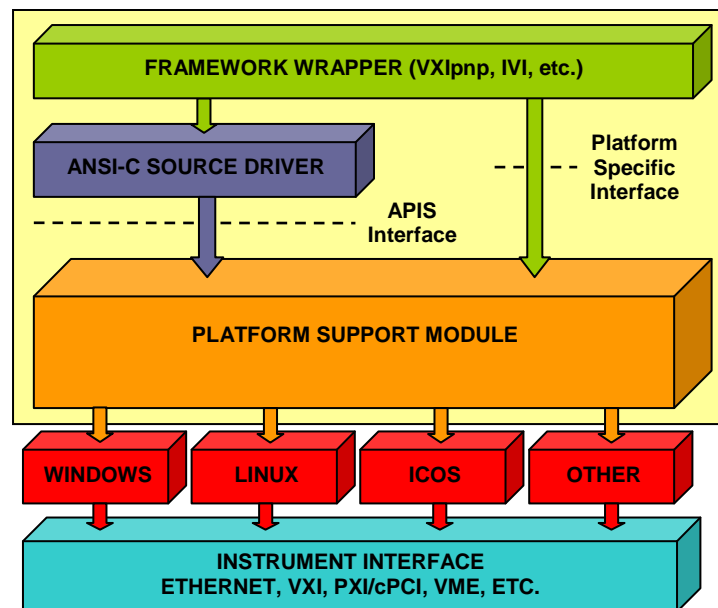


Figure 1: M-module Driver Architecture Overview

The framework wrapper provides an interface to the instrument that meets the definitions of a certain framework or specification. For example, in a VXI Plug and Play (VXIpnp) framework wrapper, all VXIpnp required functions will be available, every function will be in the appropriate format, and only VXIpnp defined data types will be used. In addition, a VXIpnp framework wrapper provides a standard Windows dll interface. The framework wrapper provides very little, if any, true instrument functionality. It instead, calls the ANSI-C driver to perform instrument related tasks. By definition, the framework wrapper is not platform independent.

The platform support module is a library of functions that can be used to access the M-module, the M-module carrier, and operating system utilities. The programming interface to the platform support module is separated into two pieces: 1) a platform independent interface that follows the AcQuisition Technology AcQ Platform Interface Software (APIS) definition and 2) an undefined platform specific interface that will vary between platform support modules. The platform support module is inherently not platform independent; however, a calling software module can remain platform independent as long as it only makes function calls to the APIS programming interface.

The ANSI-C Source Driver provides a fully platform independent interface to the instrument functionality of the M-module. The driver functions are determined by the module's hardware capabilities and are intended to provide the instrument user with a high level software interface with which to perform the necessary measurement or configuration tasks. In essence, the ANSI-C module is the meat of the driver, providing over 90% of the instrument functionality. It is written entirely in ANSI-C and it must only call ANSI-C libraries so that it can be compiled for any platform that is supported by an ANSI-C development environment. All instrument I/O must be performed through the platform support module using the APIS programming interface. This ensures portability across all platforms.

The combination of the Framework Wrapper, ANSI-C Source Driver and Platform Support Module creates a fully functional driver that can be easily ported from one platform to another. C&H has developed this architecture with the goal of providing comprehensive software support for its M-modules regardless of the platform.

PLATFORM INDEPENDENCE

Since the goal of this architecture is to maximize platform independence, it is important to define what is meant by "platform." In our case, a "platform" is a combination of an M-module carrier(s) and an operating system. For example, the following list describes a few of the many possible platforms: 1) VX405C/Windows, 2) VX405C/Linux, 3)AMi3002/Windows, 4) VX406C/embedded Linux. As you can imagine, there is a very large number of potential platforms.

The ideal driver architecture would allow a single driver for a given M-module to be used on any platform without modification. In reality, if a driver is to provide full instrument functionality or is to meet a certain software specification, it must contain some platform dependant features, for example, triggering features that are specific to a given M-module carrier. This is also true if the driver is to provide features that are specific to a certain operating system such as a Windows dll interface. It is therefore understood that 100% platform independence is an ideal objective that is not entirely achievable without limiting the functionality of the driver. Therefore, it is not the objective of the

architecture to provide 100% platform independence but to *maximize* platform independence and by doing so, greatly reduce the level of effort required to cross platforms.

With that said, it is also important to understand that **it is possible for a platform to be a combination of multiple M-module carriers and a single operating system.** This is exactly the case for most of the drivers provided by C&H where we support a number of M-module carriers on Microsoft Windows using a single platform we call CHWIN.

VXIpnp DRIVER

The vast majority of C&H M-module users will only require the module's VXIpnp driver. It is a common misconception that VXIpnp drivers are for VXI instruments only. True, the VXIpnp specifications were developed by the VXIbus Consortium and were originally intended for VXI instruments; however, the specification is suitable for instruments hosted on a number of physical interfaces.

A VXIpnp driver is a combination of a VXIpnp Framework Wrapper, an ANSI-C driver and CHWIN. In addition, it includes various other utilities such as a soft front panel executable and Windows help. It is delivered as a self contained installation allowing for easy "plug and play" functionality. The only external requirement is the VISA library for use in instrument communication.

The key features of C&H's VXIpnp drivers are:

- Microsoft Windows Soft Front Panel executable that can be used to interactively control the instrument
- A Microsoft Windows dll library with static import libraries for both Microsoft Visual C/C++ and Borland development environments.
- LabWindows/CVI function panels
- Windows help file
- Automatic installation
- Source Code
- Support for other development environments including Visual Basic and LabView
- Use on all M-module carriers supported by CHWIN.

CHWIN – PLATFORM SUPPORT MODULE

CHWIN is C&H's platform support module for windows that supports most of C&H's M-module carriers. As of the publication of this document, CHWIN supports the following carriers: VX405C, AMi2001, AMi3000, AMi3002, EM405D, EM405-8. Support for other carriers will be added as required.

CHWIN is delivered in a Microsoft Windows dll with supporting files including header files, import libraries, and configuration files. A self contained installation is available; however, in most cases, CHWIN will be installed as part of a VXIpnp driver installation.

User applications can link directly to CHWIN to access device and platform functionality. Details on CHWIN are provided in the CHWIN User's Manual.

OTHER PLATFORM SUPPORT MODULES

Other platform support modules may be available for various M-module carriers and/or operating systems. C&H will develop platform support modules as needed. Also, many are currently available from AcQuisition Technology. Further, the APIS definition is openly available from AcQuisition Technology; therefore, users can readily develop their own platform support modules.

For information on APIS or AcQuisition Technology's Platform Support Modules, visit the AcQuisition Technology website at www.acq.nl

ANSI-C DRIVER

Advanced users such as those using a platform that is not yet supported by a platform support module or those using an embedded system that requires compilation of the driver directly into the application, will typically require the ANSI-C driver.

The ANSI-C driver consists of a single ".c" source file and a header file. It is written entirely in ANSI-C and only calls the ANSI-C Standard Libraries, with the exception of APIS functions. Porting this driver to a different platform simply requires replacing the platform support module and re-compiling.

The ANSI-C Source Driver is installed as part of the VXIpnP Driver; however it is also available for download separately.

LEGACY VXIpnP DRIVER

Prior to developing this architecture, C&H delivered VXIpnP drivers that were written in a non-modular fashion and only supported a limited number of carriers. Previous users who do not intend to cross platforms may wish to continue using these old drivers. C&H will continue to make these drivers available for download under the label "Legacy VXIpnP Driver."

C&H will no longer add functionality nor maintain Legacy VXIpnP Drivers; therefore, you are encouraged to, when possible, migrate away from the Legacy VXIpnP Drivers.

LABVIEW

VXIpnP driver functions can be called in LabView by using the *Call Library Function Node* to link to the Windows dll. Various National Instruments' documentation describes, in detail, how to perform this action.

Further LabView support can be provided by using the "Import CVI Instrument Driver" (or "Create VI Interface to CVI Instrument Driver" in LabView 8.0 and later) feature of LabView. This feature will use the VXIpnP driver's ".fp" file to create a LabView wrapper around the driver dll. This will create a fully functional VI for each driver function defined in the ".fp" file.

IMC

Interactive Mezzanine Control (IMC) is a Windows application that allows the user to interface to any M-Module on any of the M-module carriers that are supported by CHWIN. It provides low level register and memory access to the M-module through an easy to use graphical interface. It does not employ M-module drivers but instead, communicates with CHWIN directly. It is generic in nature, meaning it contains no knowledge of the M-module functionality and therefore can be used with any M-module regardless of the manufacturer and regardless of whether the M-module has a driver.

DRIVERS ON THE C&H WEBSITE

On the C&H website, each M-module will have a unique “software support” page that contains links to all the relevant software for that particular module. You can navigate to the software support page from a link found on the module’s product page or from the menu under Support → Drivers. A sample screen shot of a software support page is shown in Figure 2. Please note that at the time of publication of this document, C&H is still transitioning from Legacy VXIppn Drivers to this new architecture. Software support pages will be added to the website as drivers are released during the transition period.



Figure 2: Screen shot of the MA203 Software Support web page

For questions, please contact C&H Technologies, Inc. at 512-733-2621 or support@chtech.com.