Nominal Data Acquisition Device Support for EPICS

Vyacheslav A. Isaev, Niklas Claesson (Cosylab, Ljubljana, Slovenia),
Mark Plesko, Klemen Zagar (COBIK, Solkan, Slovenia)

Abstract

A large number of devices offer a similar kind of capabilities. For example, data acquisition all offer sampling at some rate. If each such device were to have a different interface, engineers using them would need to be familiar with each device specifically, inhibiting transfer of know-how from working with one device to another and increasing the chance of engineering errors due to a miscomprehension or incorrect assumptions. In the Nominal Device Model (NDM) model, we propose to standardize the EPICS interface of the analog and digital input and output devices, and image acquisition devices. The model describes an input/output device which can have digital or analog channels, where channels can be configured for output or input. Channels can be organized in groups that have common parameters. NDM is implemented as EPICS Nominal Device Support library (NDS). It provides a C++ interface to developers of device-specific drivers. NDS itself inherits well-known asynPortDriver. NDS hides from the developer all the complexity of the communication with asynDriver and allows to focus on the business logic of the device itself.

Nominal Device Model

Generalized interfaces for:

- Data acquisition (DAQ).
- Signal generators (analog output).
- Digital input and output devices.
- Cameras and other image acquisition devices cameras / image acquisition devices.
- Timing receivers.

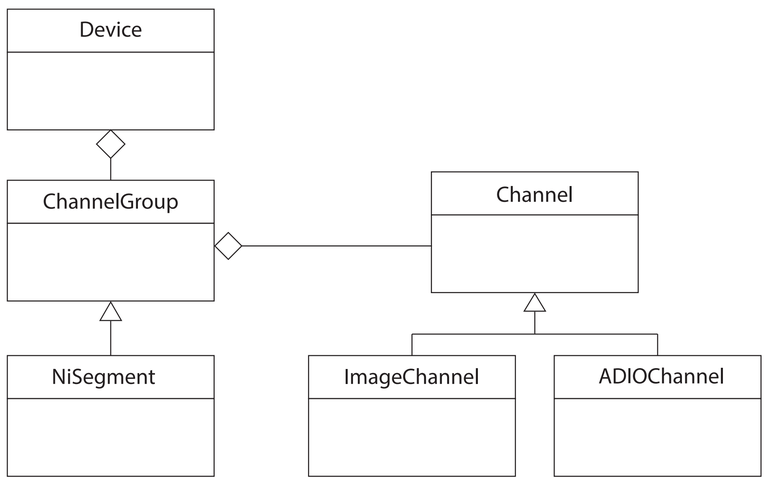


Figure 1: Class diagram of NDS class hierarchy (excerpt)

The base class **Device** provides functions common to all devices (e.g., the state machine, firmware upgrade management, etc).

Devices consist of **channel groups**. Channel groups are used to model channels that are coupled in a way (e.g., sample rate that can only be set for multiple channels simultaneously).

Channels are of multiple types. E.g., cameras would have **Image Channels**, whereas analog/digital I/O devices would have **ADIO Channels**.

Benefits

Users:

- Common EPICS interface: same structure of record names for same functions.
- Understand to work with one device -- understand them all.
- Same user interfaces.
- Scripts can be more easily adapted to new device types.

Developers:

- Straightforward C++ interface for base functions.
- Reduces implementation time.
- Increases maintainability of the device driver.
- Hide EPICS (asynDriver) complexity from the device driver developer where it is not necessary.
- Allows developers to focus on the device specific parts of the driver.

Acknowledgement

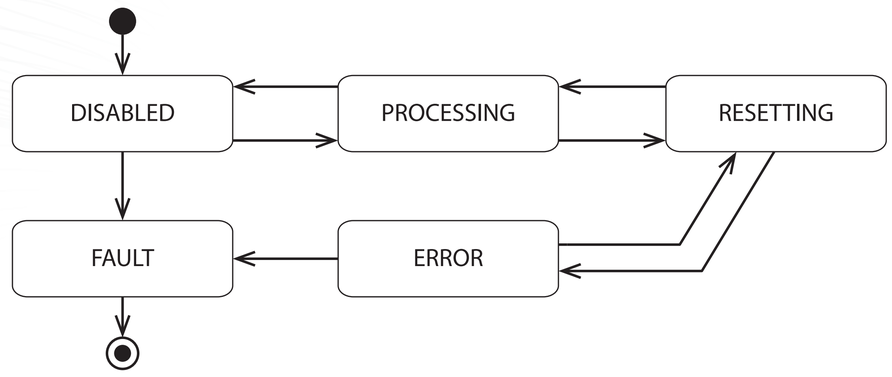
COBIK "The Centre of Excellence for Biosensors, Instrumentation and Process Control" - COBIK, Velika pot 22, SI-5250 Solkan, Slovenia

The Centre of Excellence for Biosensors, Instrumentation and Process Control - COBIK is an operation financed by the European Union, European Regional Development Fund and the Republic of Slovenia, Ministry of Higher Education, Science and Technology.

Behavior

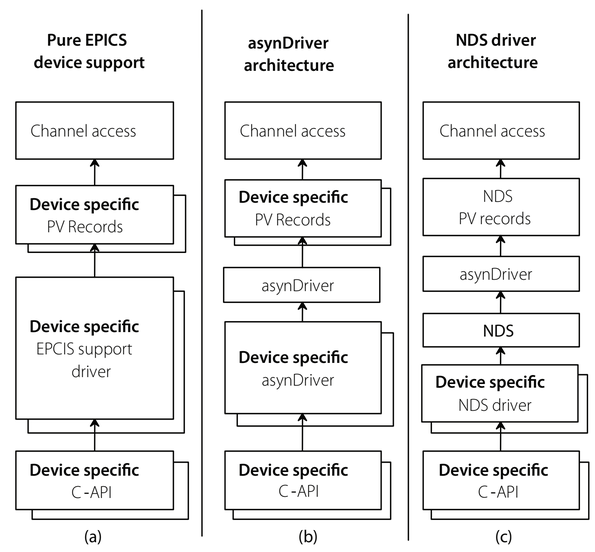
Behavior of devices and their constituents (channels, channel groups) is standardized.

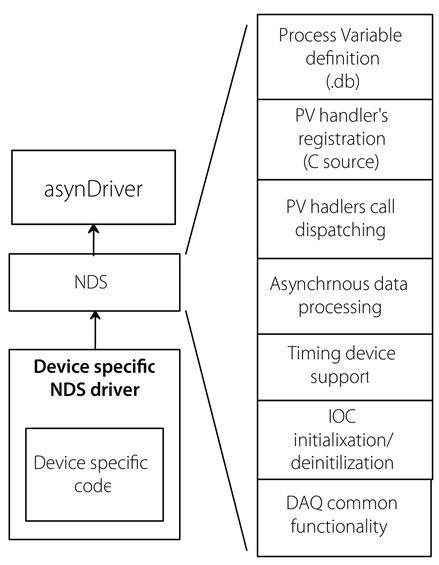
State machines (states and transitions among them) are pre-defined.



Architecture

Nominal Device Support is EPICS device support, implemented with the asynDriver library.





Functionality

- Signal acquisition as single value or waveform.
- Continuous and triggered data acquisition.
- Signal filtering.
- Fourier transforms of signals.
- Conversion of units using piecewise cubic spline curves, configurable at run time.
- Firmware upload.
- State management at the device and channel level.
- Messaging and remote procedure calls.
- Advanced threading support (polling, timers, periodic tasks, etc.)

Extensibility

NDS provides a way to easily add functions that are not standardized by NDS, but supported by a particular hardware device.

Summary

- Development done by Cosylab.
- Primarily driven by requirements of ITER CODAC diagnostics.
- Source code available under GPL license -- please contact authors of this poster for more information (klemen.zagar@cosylab.com).

