



Labview Programming Guideline for Idea Fast Proto

Mike Qin

*Lead Electrical Engineer
China Innovation Center*

2017-03-20 Initial created.

Table Of Contents

- **Chapter 1 – Using TortoiseSVN (Subversion) for Source Code Control**
- **Chapter 2 – Modifying the Acquisition Code for New DAQ Hardware**
- **Chapter 3 – Code Review & Commit Code**
- **Chapter 4 – Adding the Processing Method for New Algorithm**
- **Chapter 5 – Real-time Version of Framework**

Chapter 1

Using TortoiseSVN (Subversion) for Source Code Control

Using TortoiseSVN (Subversion) for Source Code Control

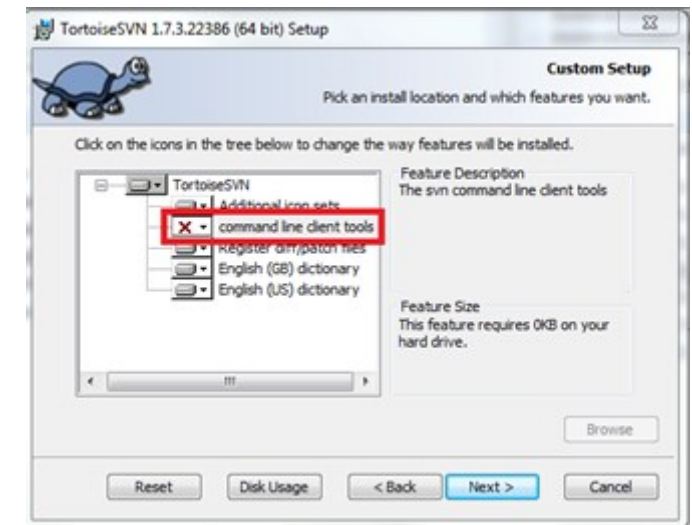
TortoiseSVN is a Subversion client, implemented as a windows shell extension, a plug-in to Windows Explorer. (<https://tortoisesvn.net/>)

1. Download TortoiseSVN (stable version 1.9.x is recommended)

The latest version can be downloaded at <https://tortoisesvn.net/downloads.html>. Select the right installer according to the 32-bit/64-bit processor of your PC, otherwise the setup will fail.

2. Install TortoiseSVN

The installation is straightforward, just accept the defaults. The only point of interest is to select the “command line client tools”. This tool is not included in the default installation choice.



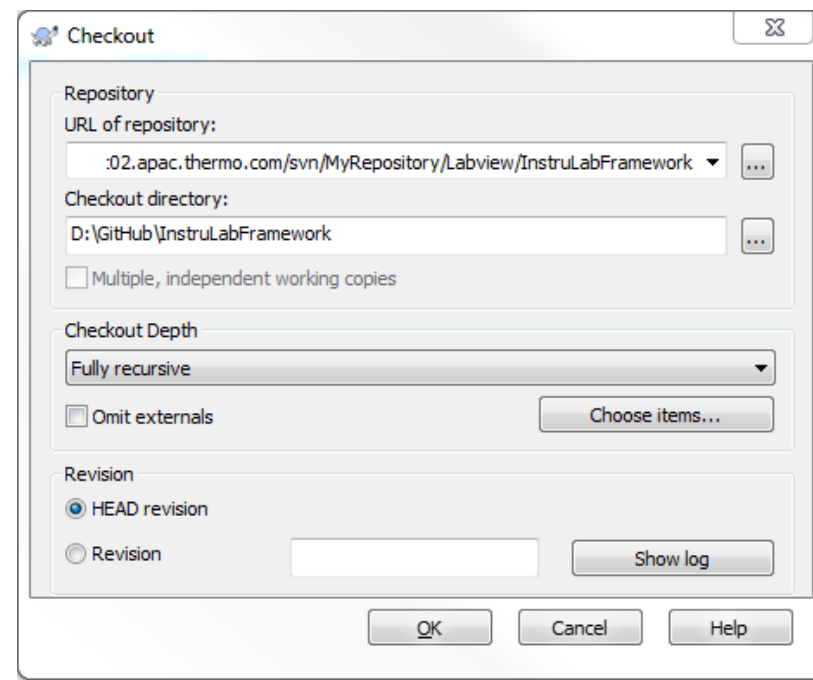
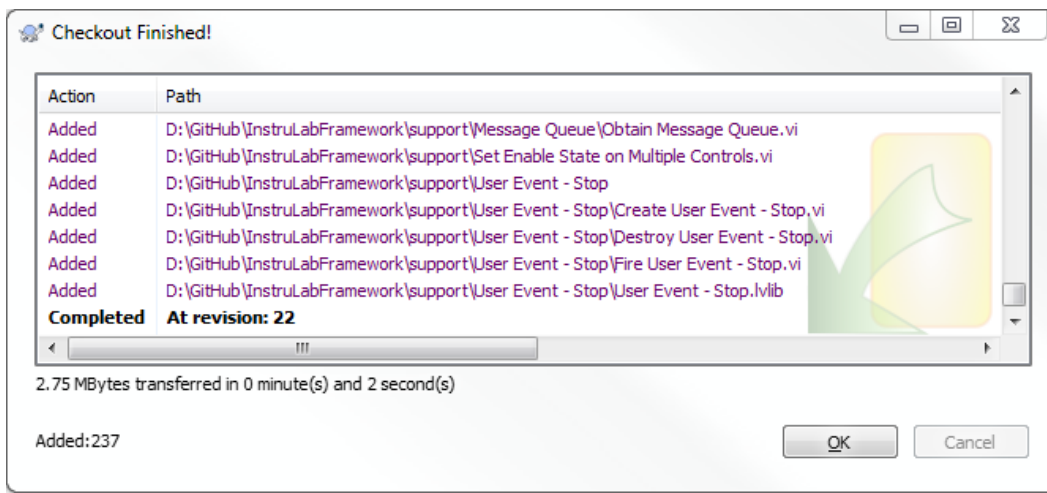
Check out the Latest Framework Code

"To get your hands on blessed, completely approved, and fully loaded Subversion directory, you need to check it out from your repository".

1. Create a folder D:\Projects\InstruLabFramework (or equivalent)

2. Right-click on D:\Projects\InstruLabFramework and choose "SVN Checkout"

- Set URL to <https://cnsho-cic02.apac.thermo.com/svn/MyRepository/Labview/InstruLabFramework>
- Set Checkout directory to D:\XXX\InstruLabFramework
- Press OK. Checkout has finished:



Check out the Latest Framework Code (2)

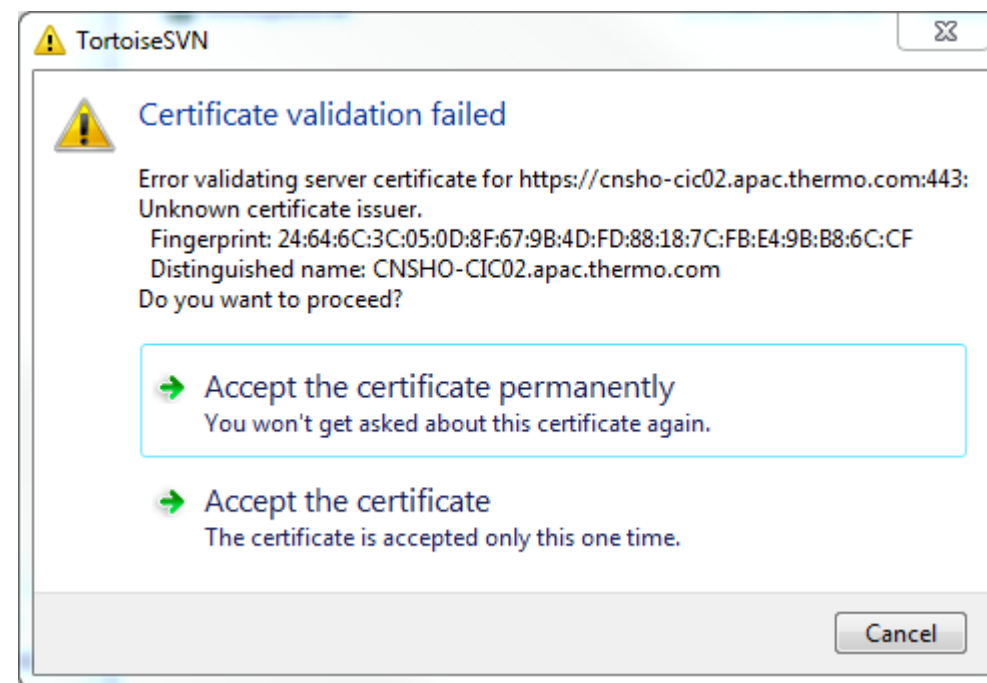
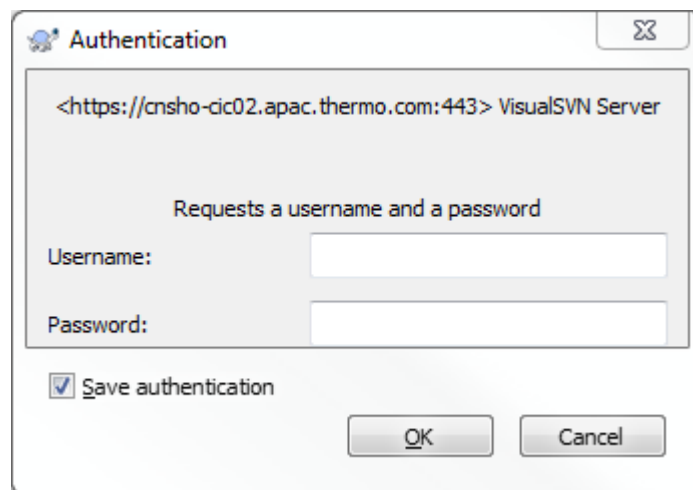
If you are first time access the repo, following window will pump up:

Certificate validation failed

- Select “Accept the certificate permanently”

Authentication

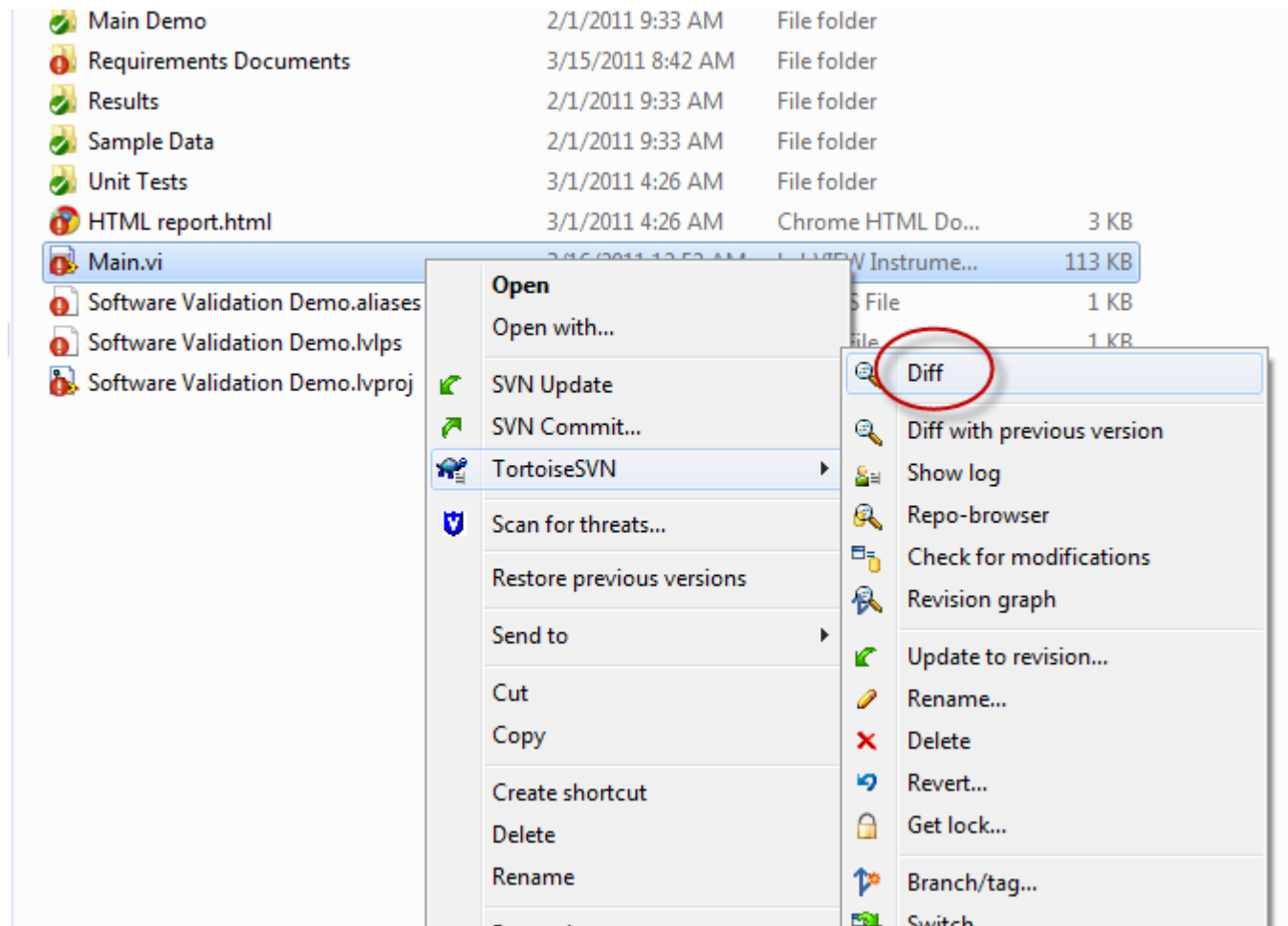
- Use your SVN account to login
- Example
 - Username: Developer1
 - Password:

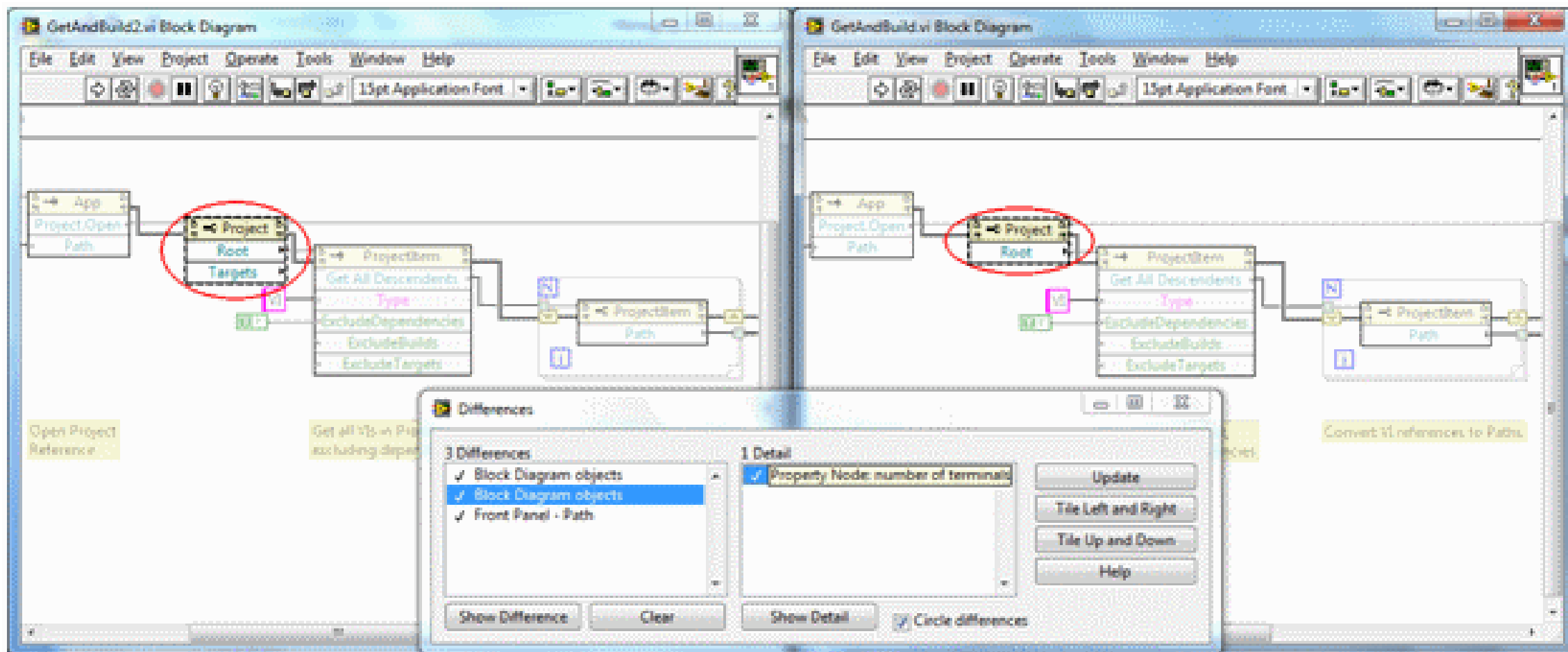


Configuring TortoiseSVN for Graphical Diff

The binary nature of LabVIEW file-types requires the use of custom utilities for diff and merge, which are available in the professional version of LabVIEW 2009 or later.

1. Right-click in Windows Explorer to expose the TortoiseSVN options
2. Mouse-over 'TortoiseSVN >> Settings'
3. In the list of settings on the left, select 'Diff Viewer' under 'External Programs'
4. On the right side of the dialog, select 'Advanced'
5. When the 'Advanced diff settings' dialog appears, select 'Add'
6. In the dialog that appears, type .vi as the extension
7. Where it prompts you for the path to the external program, type the following: "C:\Program Files\National Instruments\Shared\LabVIEW Compare\LVCompare.exe" %mine %base -nobdcosm -nobdpos
8. Repeat this operation for the file-type .ctl





Configuring TortoiseSVN for Graphical Merge

1. Right-click in Windows Explorer to expose the TortoiseSVN options
2. Mouse-over 'TortoiseSVN >> Settings'
3. In the list of settings on the left, select 'Merge Tool' under 'External Programs'
4. On the right side of the dialog, select 'Advanced'
5. When the 'Advanced merge settings' dialog appears, select 'Add'
6. In the dialog that appears, type .vi as the extension
7. Where it prompts you for the path to the external program, type the following: "C:\Program Files\National Instruments\Shared\LabVIEW Merge\LVMerge.exe" %base %mine %theirs %merged
8. Repeat this operation for the file-type .ctl

Chapter 2

Modifying the Acquisition Code for New DAQ Hardware

Software Architecture

The Labview Framework for idea fast prototyping is forked from National Instrument Continuous Measurement and Logging sample project, with customization on instrumentation related hardware devices and algorithms.

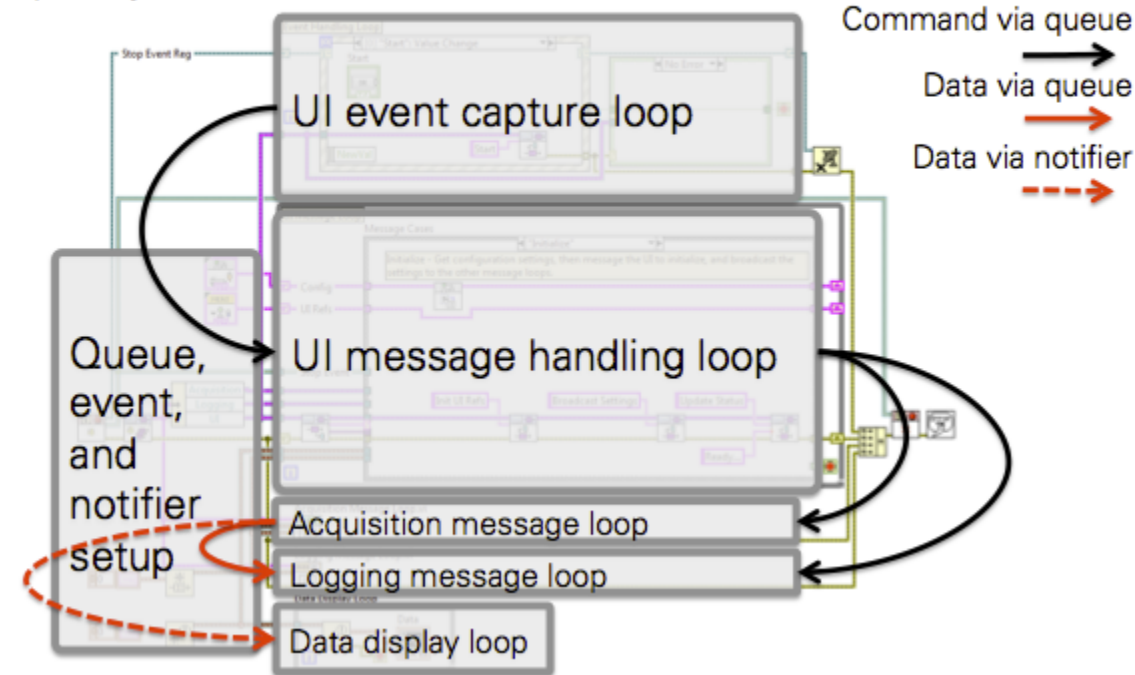
Features Queued Message Handler (QMH) mechanism w/ five loops execution in parallel:

- Event handling (Main.vi)
- User interface messaging (Main.vi)
- Acquiring data (Acquisition.lvlib:Acquisition Message Loop.vi)
- Logging data (Logging.lvlib:Logging Message Loop.vi)
- Displaying data (Main.vi)

Features a Settings dialog box (Settings.lvlib) you can use to configure the application.

Features series of DAQ device & processing algorithm available for instrumentation idea fast prototyping

Continuous Measurement and Logging Sample Project



Support DAQ Hardware

Oscilloscope:

- Tektronics DPO2000/3000/4000 (USB)
- Agilent ag6000 (USB)

Picoscope:

- 2205A (USB)

Bitscope:

- BS05 (USB)

NI Compact RIO module

NI DAQ (AI, AO, DI, DO) modules (RS485)

Datalogger:

- Agilent 34972A (USB)

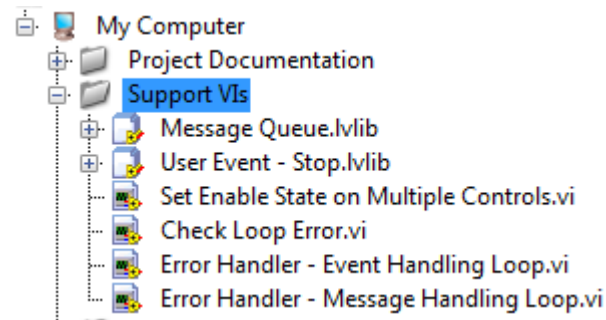
Proprietary module:

- MIB (RS485)
- PEB (SPI)

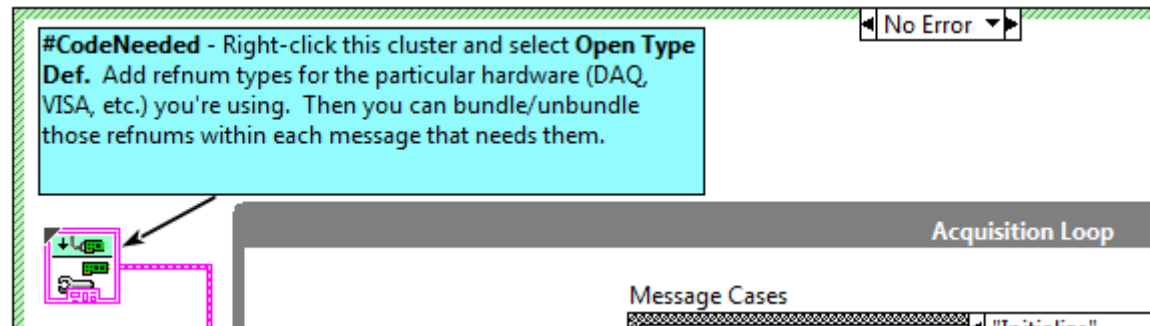
Modifying the Acquisition Code for New DAQ Hardware

You must modify the framework acquisition code to acquire data from new hardware. Complete the following steps to make these modifications:

1. Copy New DAQ hardware driver library or support Vis to InstruLabFramework/support/DAQ Drivers folder. Add the library into the Support Vis virtual folder in Labview project explorer.



2. Open Acquisition.lvlib:Acquisition Message Loop.vi and follow following instruction:

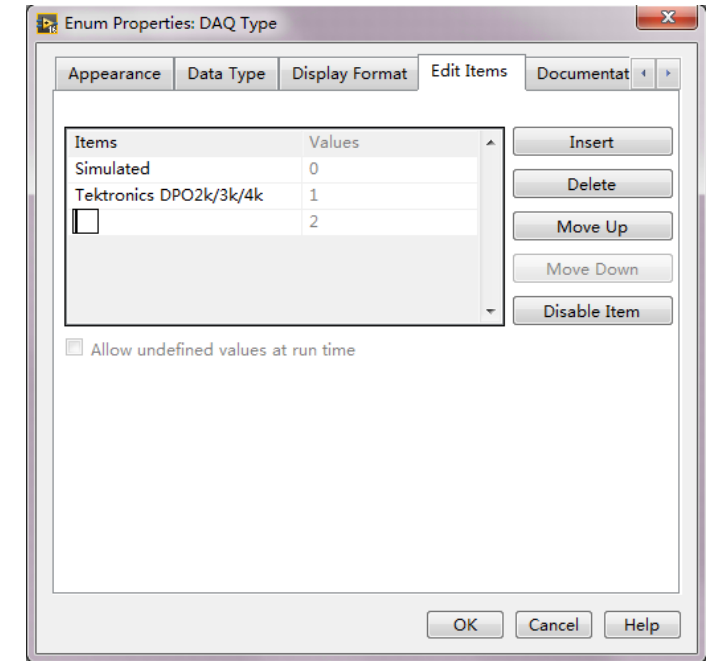
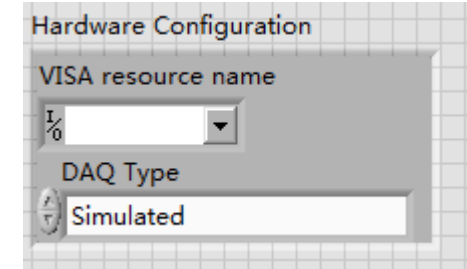


Modifying the Acquisition Code for New DAQ Hardware (cont.)

3. Add hardware refnums to Acquisition.lvlib:Hardware Configuration.ctl. For example, you can use the following objects here:

- DAQ tasks
- DAQ channels
- VISA sessions
- RS-232 ports

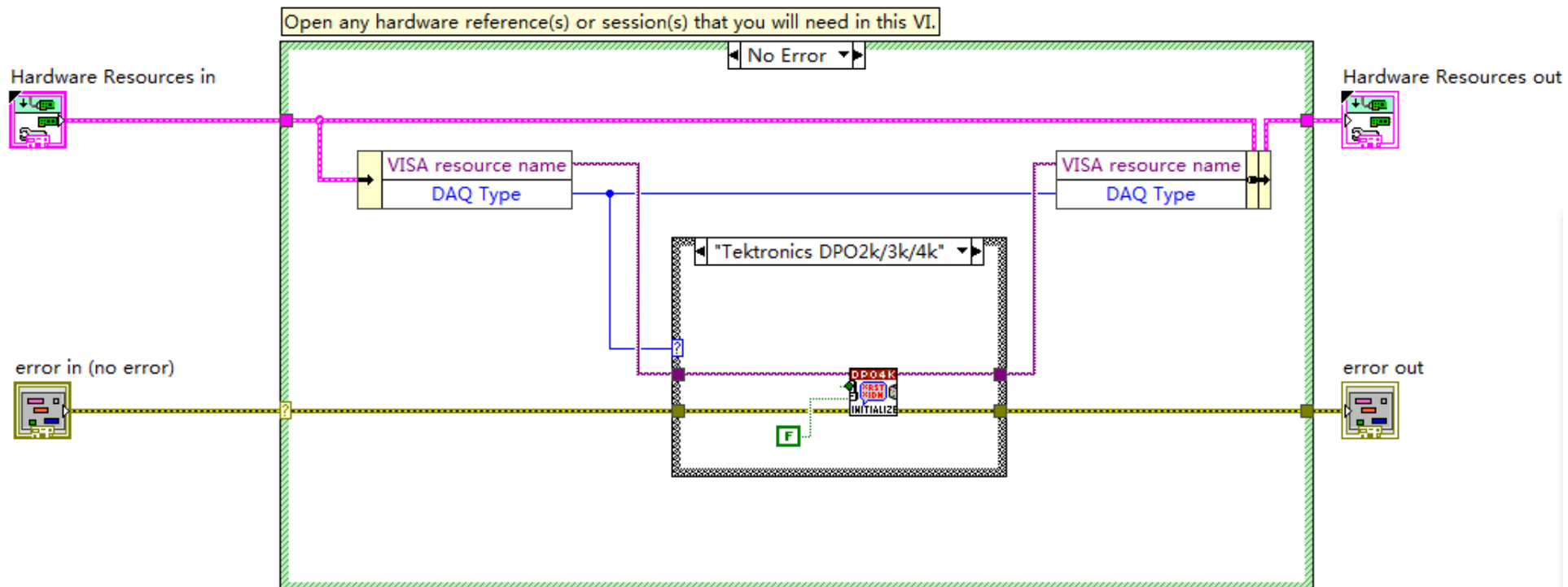
4. Right click DAQ Type enum control and select open def type. Add new hardware option in “DAQ Type” enum list. This option will be used in following case structure.



Modifying the Acquisition Code for New DAQ Hardware (cont.)

5. Add hardware initialization code to Acquisition.lvlib:Initialize Hardware References.vi with new case option. For example, you can use the following objects here:

- DAQmx Task Name constants
- DAQmx Create Virtual Channel VI
- (Instrument Driver) Initialize VI



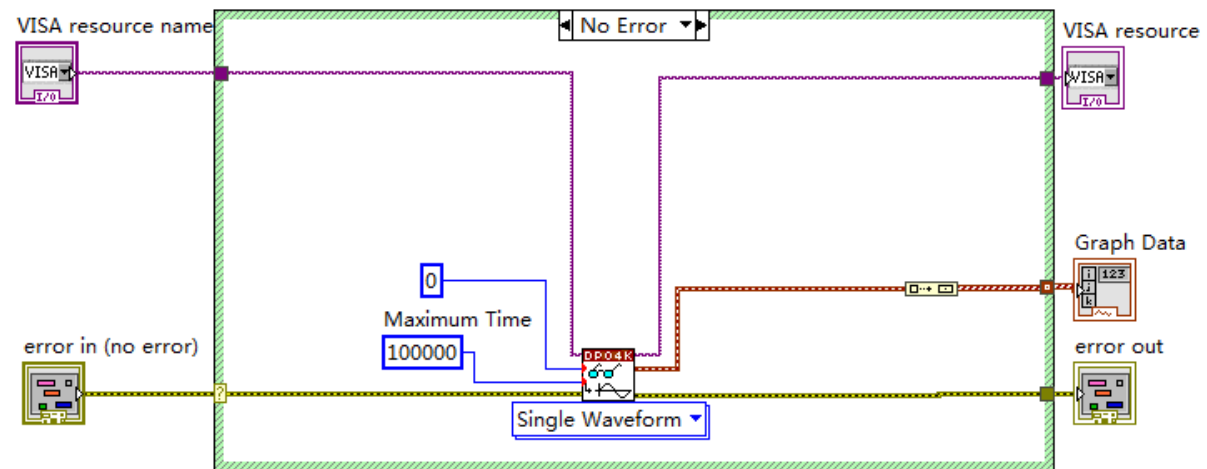
Modifying the Acquisition Code for New DAQ Hardware (cont.)

5. Add hardware configuration code to Acquisition.lvlib:Configure Hardware.vi. For example, you can use the following VIs here:

- DAQmx Timing VI
- DAQmx Trigger VI
- (Instrument Driver) Configure Measurement VI
- (Instrument Driver) Configure Autozero VI

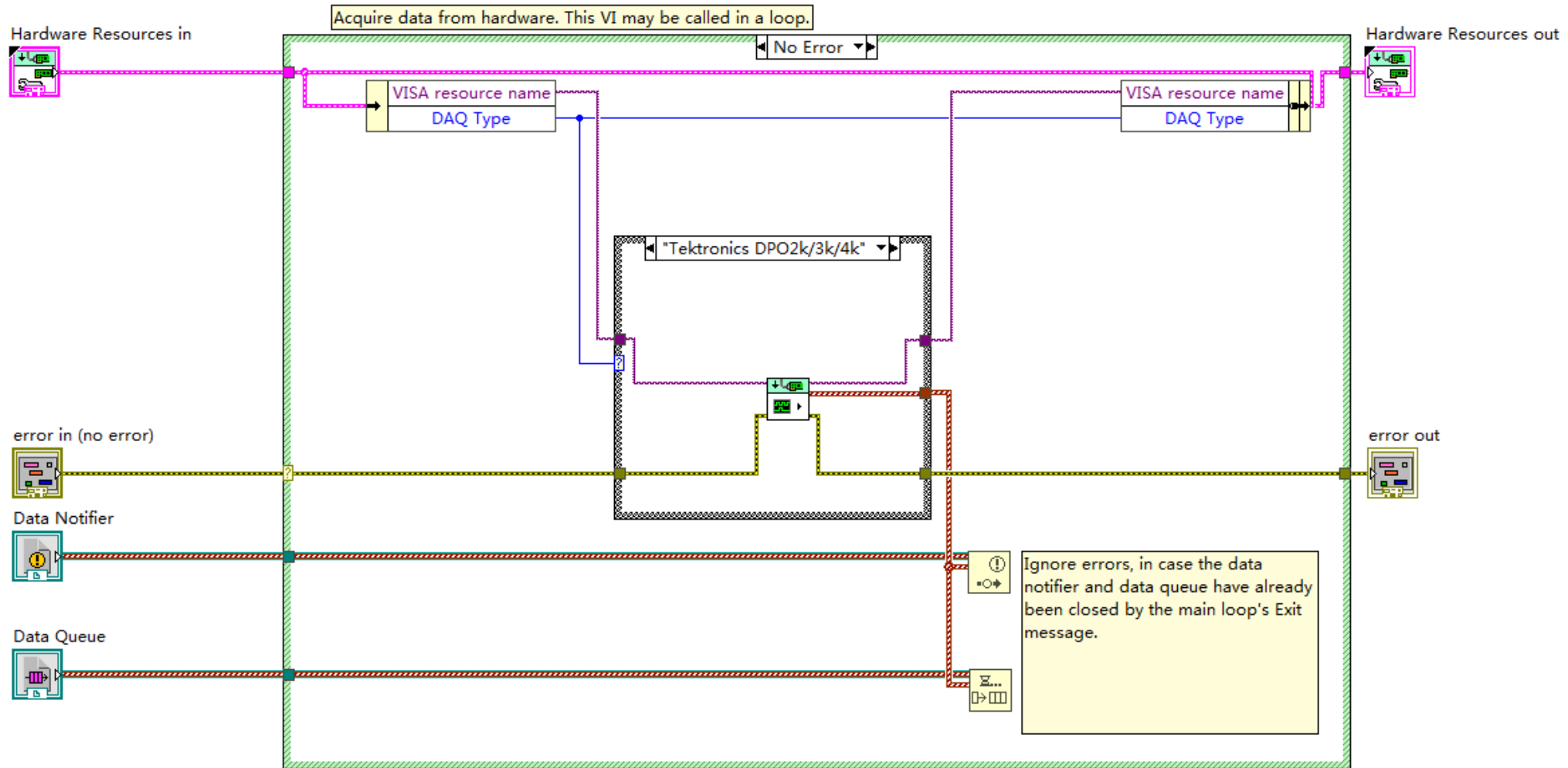
6. Create a subVI by copy “Generate Tektronix DPO MSO 2000 4000 Series Data.vi” and save as “Generate [New DAQ Hardware] Data.vi”. Add hardware configuration code. For example, you can use the following VIs here:

- DAQmx Read VI
- (Instrument Driver) Read VI



Modifying the Acquisition Code for New DAQ Hardware (cont.)

7. Add subVI “Generate [New DAQ Hardware] Data.vi” to Acquisition.lvlib:Acquire.vi for data acquisition.



Modifying the Acquisition Code for New DAQ Hardware (cont.)

8. Add code that stops data acquisition to Acquisition.lvlib:Stop Acquisition.vi. For example, you can use the following VIs here:

- DAQmx Clear Task VI
- (Instrument Driver) Close VI

