ATLAS Internal Note
ATL-D-ES-0004, Part 3
Version 1.0
28 January 2003

C++ Wrapper for the VMEbus Library of ROD Crate DAQ

Author : R. Spiwoks

Abstract

This note describes the C++ wrapper for the VMEbus library of ROD Crate DAQ.

1 Introduction

This note describes the C++ wrapper for the VMEbus library of ROD Crate DAQ. The VMEbus library itself is documented in References [1] and [2].

2 Package

The C++ wrapper for the VMEbus library of ROD Crate DAQ is contained in package **RCDVme** of the TDAQ ROS software repository [3]. In order to build the package the instructions given in the reference have to be followed.

The package allows a **DEBUG** flag to be set in the **cmt/requirements** file. When the package is compiled with this flag on, then the C++ wrapper will generate print-out to **cout**, in particular for all accesses on the VMEbus.

3 Application Program Interface

The C++ wrapper of the VMEbus library for ROD Crate DAQ implements the specification in Reference [1], Chapter 5 “Ideas for a C++ Binding”. The definition files are available in directory **RCDVme/**. They define two levels of classes:

1. The **VME** class is a singleton class created and deleted using static methods. It contains members for return codes, CR/CSR space access, bus error handling and printing of general information.
2. The **VMEMasterMap**, **VMESlaveMap**, **VMEBlockTransfer** and **VMEInterrupt** classes implement master mapping, slave mapping, block transfer and interrupts, respectively. They correspond to the identifiers used in the C binding, cf. [1].

Most methods return an unsigned integer (u_int). This contains the return value used by the VMEbus library, cf. [1].

The methods for single-cycle reading and writing of master mappings use polymorphism for the different data types, e.g. VMEMasterMap::ReadSafe() for reading safely unsigned char (u_char), unsigned short (u_short) and unsigned integer (u_int).

4 Programming Examples

Programming examples are available for the test programs **src/test/menuRCDVme.cc** and **src/test/testRCDVme.cc**, see also Section 5.

5 Test Programs

Two test programs are provided:

- **menuRCDVme** provides simple text-driven menus which allow to call all methods of the C++ wrapper for the VMEbus library of ROD Crate DAQ. This program is intended for testing the C++ wrapper itself but can also be used to debug the user program or the VMEbus modules, especially if the package is compiled with the DEBUG flag on.

- **testRCDVme** provides a simple text-driven menu for reading and writing of a master mapping. This program is intended for low-level interactive communication with a VMEbus module. Commands to be used are:
  - **h** to print a help message;
  - **read** or **r <address> 0 <loop>** to read a short value from address *address loop* times;
  - **write** or **w <address> <data> <loop>** to write short value *data* to address *address loop* times;
  - **readl** or **rl <address> 0 <loop>** to read a long value from address *address loop* times;
  - **writel** or **wl <address> <data> <loop>** to write long value *data* to address *address loop* times;
  - **readcsr** or **rc <slot> 0 <field>** to read a value from CR/CSR space at slot *slot* and field *field*;
  - **writecsr** or **wc <slot> <data> <field>** to write value *data* to CR/CSR space at slot *slot* and field *field*;
  - **q** to quit the test program.

  If the value *loop* in the above commands is not filled, a default value of *1* is used. If the value *loop = -1* in the above commands is used, the command is executed in an infinite loop. This infinite loop can be interrupted using Ctrl-C.

6 Known Limitations

The return values of the class methods shall in the near future be replaced by using the C++ exception mechanism.

Please send any bug reports or requests for moddifications of the C++ wrapper for the VMEbus library of ROD Crate DAQ to Ralf.Spiwoks@cern.ch.

7 References

[1] R. Spiwoks et al., VMEbus Application Program Interface, ATLAS EDMS note ATL-D-ES-0004, https://edms.cern.ch/document/325729.

[2] M. Joos and J. Petersen, Implementation Notes for the VMEbus API on Linux-based Processors from Concurrent Technologies, ATLAS EDMS note ATL-D-ES-0004, https://edms.cern.ch/document/325729.

[3] The ATLAS TDAQ Read-out System, Code Repository, http://atlas.web.cern.ch/Atlas/GROUPS/DAQTRIG/ROS/ros.htm.