

MultiDtacq User Guide

Prepared By: Peter Milne

Date: 20 September 2010

Table of Contents

1	Introduction.....	5
1.1	What is MultiDtacq.....	5
1.2	Intended Audience.....	5
1.3	Scope.....	5
1.4	References.....	5
1.4.1	MDSplus: http://www.mdsplus.org : Data archive and plot system.....	5
1.4.2	Kst : http://kst-plot.kde.org : Cross platform, standalone plot tool.....	5
1.5	Glossary.....	5
1.6	Notation.....	6
2	Description.....	7
2.1	Linux GTK.....	7
2.2	MS-Windows XP:.....	7
2.3	MS-Windows 7:.....	7
2.4	MAC.....	8
3	Installation.....	9
3.1	Pre-Requisite.....	9
3.2	Download the .zip file.....	9
3.3	Install Rack and Scenario.....	9
3.4	Run.....	9
3.4.1	Check it's working.....	9
3.4.2	Init.....	9
3.4.3	Arm.....	9
3.4.4	Trigger / Event.....	9
3.4.5	Stop	10
3.4.6	Exit.....	10
4	Customization.....	11
4.1	Properties File:.....	11
4.2	Rack Definition file.....	12
4.3	Capture Definition File.....	13

Copyright and Attribution.

Document created using OpenOffice.Org www.openoffice.org.

This document and D-TACQ Software comprising platform Linux port, Linux kernel modules and most applications are released under GNU GPL/FDL:

Document:

Copyright (c) 2004-6 Peter Milne, D-TACQ Solutions Ltd.

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.2, with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts.

Software:

Copyright (C) 2004-6 Peter Milne, D-TACQ Solutions Ltd.

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License version 2 as published by the Free Software Foundation.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

1 Introduction

1.1 What is MultiDtacq

MultiDtacq is a host side client platform designed to control multiple *D-TACQ ACQxxx* intelligent digitizers concurrently.

MultiDtacq allows the user to monitor and control a population of cards (a “Rack”) and to control the cards in a pre-programmed *Scenario*.

Both the *Rack* definition and the *Scenario* definition are defined in *XML* data files. This means that one program works for any *Rack*, many *Scenarios*. Currently, the configuration files are supplied direct to users by *D-TACQ*, however, we plan to release the *ACQOMATIC Rack/Scenario* generation tool to allow users to do it themselves.

MultiDtacq does allow users to set 3 common operating parameters (*PRE*, *POST*, *CLK*), and to control the shot cycle.

MultiDtacq is CROSS PLATFORM – requires a Java 1.6 runtime, works on MS-Windows, MAC OSX and any flavour of Linux. *MultiDtacq* uses the *Web-Service* interface to control the digitizers; this is a “universal” remote control interface, works from any modern system.

MultiDtacq does not attempt to plot data. Highly capable tools such as MDSPLUS 1.4.1 and KST 1.4.2 make an excellent job of this.

1.2 Intended Audience

End-Users

1.3 Scope

Software client application guide. Does not attempt to specify what the scenarios are.

1.4 References

1.4.1 MDSplus: <http://www.mdsplus.org> : Data archive and plot system.

1.4.2 Kst : <http://kst-plot.kde.org> : Cross platform, standalone plot tool

1.5 Glossary

- Rack: set of ACQxxx cards in a system
- Scenario: capture scenario, determines nature of the capture.
- Web-Service : a generic remote procedure call mechanism

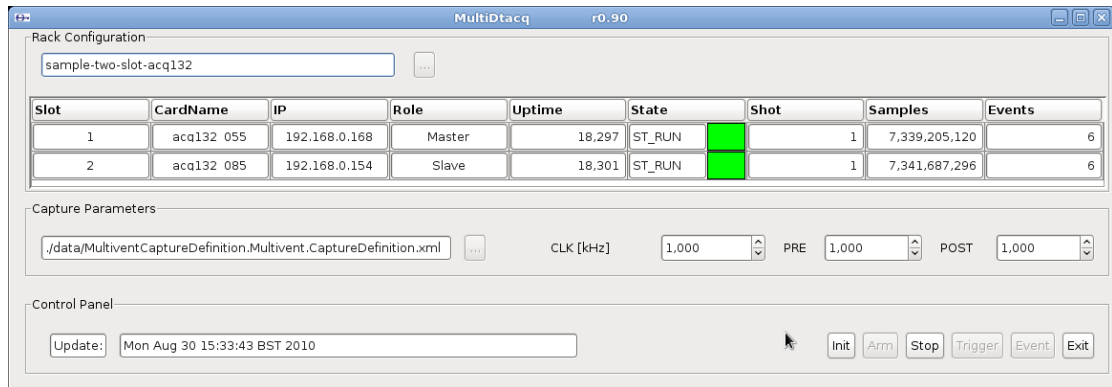
1.6 Notation

- **command** : indicates name of a program (command)
- preformatted text : literal input or output from terminal session.
- *Defined Term* : some term or acronym specific to this domain (perhaps referenced in the glossary)

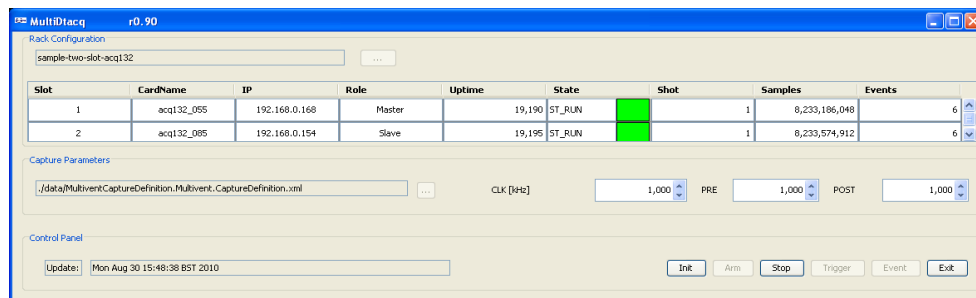
2 Description

MultiDtaq comprises a single Main Screen. The content of the program is completely configured with configuration files.

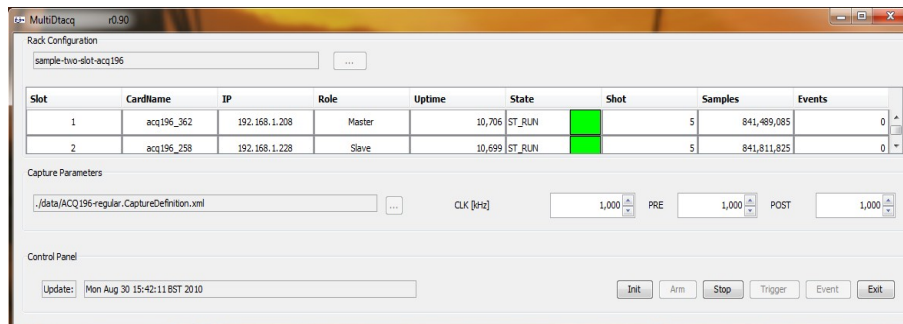
2.1 Linux GTK



2.2 MS-Windows XP:

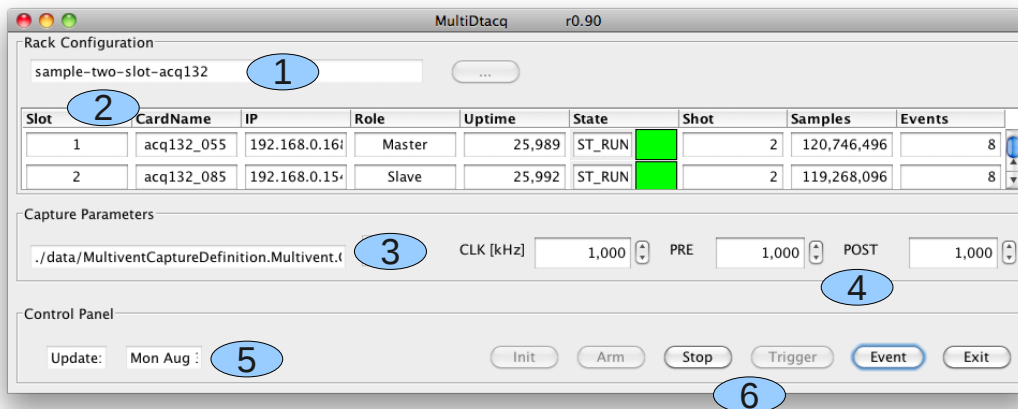


2.3 MS-Windows 7:



2.4 MAC

With main features annotated:



- 1 Rack Definition - name
- 2 Rack Status display – one row per Card “Slot”,
Live update of state, samples
- 3 Scenario Definition Name
- 4 User Parameters : CLK, PRE, POST
- 5 Status Indicator
- 6 Control Buttons:
Init : initialises the population
Arm : Arms the population
Stop : Stop a Capture
Trigger, Event : Start, assert an Event
Exit : Quit the app

Control buttons enable in sequence

3 Installation

3.1 Pre-Requisite

Install Java 1.6 run time or better.

3.2 Download the .zip file

(Check downloads page for latest version)

<http://www.d-tacq.com/swrel/multidtacq-1009210854.zip>

Create directory [C:\MULTIDTACQ](#) aka MDQ

Extract all to [C:\MULTIDTACQ](#)

Right click on **MULTIDTACQ.BAT**, create short cut, drag to desktop.

3.3 Install Rack and Scenario

D-TACQ may supply a customized rack definition and capture scenario.

Unpack this in the same directory.

3.4 Run.

Execute **MULTIDTACQ.BAT** in the same directory, either by double clicking the icon in the directory or double click the short-cut icon on the desktop.

Control buttons should highlight in sequence, it should be quite difficult to follow the wrong sequence.

3.4.1 Check it's working

Each card in the rack definition should show updating status.

If not, likely the ip-addresses are not correct.

The serial numbers of the cards should be updated to reflect the actual cards in use.

See 4.2 for description of appropriate data file.

3.4.2 Init

Initialises the cards. Runs the `initAction` defined in the `CaptureDefinition`.

User is free to modify Capture Parameters. Do this before pressing Init.

3.4.3 Arm

Arms the cards. Runs the `armAction` defined in the `Capture Definition`.

3.4.4 Trigger / Event

Optionally trigger and send events to the cards. Use of these buttons is definition-dependent.

3.4.5 Stop

Stop a capture. Some types of capture will stop as a matter of course.

3.4.6 Exit

Quit the application. Saves current CaptureParameters for next use.

4 Customization

MultiDtacq is customized by 3 configuration files.

In addition, it will store last used state. The first, a properties file is flat text. Other files are *XML*.

The *XML* files are a software generated definition of internal object state, however, since it's *XML*, it can be modified with a standard text editor.

D-TACQ strongly recommends using an effective text editor with syntax highlighting like [jEdit](#). This is a very small download given that the Java runtime is already installed.

Do not use “notepad”. “Wordpad” may be OK, but check that it does not add an unwanted extra “.txt” extension to the file name.

4.1 Properties File:

MDQ/com.d_tacq.multicon.Multicon.

Defines:

- rackDefinition : defines the ACQxxx cards in the rack
- captureDefinition : defines the capture scenario
- captureParameters: defines default user-modifiable parameters.

Example:

data/ACQ132-multivent.com.d_tacq.multicon.Multicon

```
# MULTIDTACQ default properties for ACQ132-multivent
# output Period in msec
outputPeriod=1000
# monitorPeriod in msec
monitorPeriod=1000
# run this remote command
monitorCommand=/usr/local/bin/monitor2
# output file
outputFile=stdout
rackDefinition=\
./data/sample-two-slot-acq132.Rack.xml

captureParameters=\
./data/MultiventCaptureParameters.Multivent.CaptureParameters.xml

captureDefinition=\
./data/MultiventCaptureDefinition.Multivent.CaptureDefinition.xml
```


4.3 Capture Definition File

Example: MultiventCaptureDefinition.Multivent.CaptureDefinition.xml

File defines actions to take on each button press. It's unlikely that you will need to modify this file.

```
<?xml version="1.0" encoding="UTF-8"?>
<java version="1.6.0_21" class="java.beans.XMLDecoder">
  <object class="com.d_tacq.multicon.CaptureDefinition">
    <void property="armAction">
      <string>set.dio32_bit 0 0;multivent-init $ROLE $CLK</string>
    </void>
    <void property="initAction">
      <string>daemon multivent-run-demux $PRE $POST</string>
    </void>
    <void property="initEveryTime">
      <boolean>>true</boolean>
    </void>
    <void property="name">
      <string>Multivent</string>
    </void>
    <void property="stopAction">
      <string>acqcmd setAbort</string>
    </void>
    <void property="triggerAction">
      <string>set.dio32_bit 0 1</string>
    </void>
    <void property="eventAction">
      <string>set.dio32_bit 0 N</string>
    </void>
    <void property="type">
      <string>CaptureDefinition</string>
    </void>
  </object>
</java>
```