Software for implementing trigger algorithms on the upgraded CMS Global Trigger System

This content has been downloaded from IOPscience. Please scroll down to see the full text.

(http://iopscience.iop.org/1742-6596/664/8/082031)

View the table of contents for this issue, or go to the journal homepage for more

Download details:

IP Address: 188.184.3.56
This content was downloaded on 17/02/2016 at 09:02

Please note that terms and conditions apply.
Software for implementing trigger algorithms on the upgraded CMS Global Trigger System

Takashi Matsushita and Bernhard Arnold

Institute of High Energy Physics, Austrian Academy of Sciences, Nikolsdorfer Gasse 18, 1050 Wien, Austria
E-mail: takashi.matsushita@cern.ch

Abstract. The Global Trigger is the final step of the CMS Level-1 Trigger and implements a trigger menu, a set of selection requirements applied to the final list of objects from calorimeter and muon triggers to meet the physics objectives [1]. The conditions for trigger object selection, with possible topological requirements on multi-object triggers, are combined by simple combinatorial logic to form the algorithms. The LHC has resumed its operation in 2015, the collision-energy will be increased to 13 TeV with the luminosity expected to go up to $2 \times 10^{34} \text{ cm}^{-2}\text{s}^{-1}$. The CMS Level-1 trigger system will be upgraded to improve its performance for selecting interesting physics events and to operate within the predefined data-acquisition rate in the challenging environment expected at LHC Run 2. The Global Trigger will be re-implemented on modern FPGAs on an Advanced Mezzanine Card in MicroTCA crate. The upgraded system will benefit from the ability to process complex algorithms with DSP slices and increased processing resources with optical links running at 10 Gbit/s, enabling more algorithms at a time than previously possible and allowing CMS to be more flexible in how it handles the trigger bandwidth. In order to handle the increased complexity of the trigger menu implemented on the upgraded Global Trigger, a set of new software has been developed. The software allows a physicist to define a menu with analysis-like triggers using intuitive user interface. The menu is then realised on FPGAs with further software processing, instantiating predefined firmware blocks. The design and implementation of the software for preparing a menu for the upgraded CMS Global Trigger system are presented.

1. Introduction

The Global Trigger is the final step of the CMS Level-1 trigger and implements a trigger menu, a set of selection requirements applied to the final list of objects from calorimeter and muon triggers to meet the physics objectives [1]. The conditions for trigger object selection, with possible topological requirements on multi-object triggers, are combined by simple combinatorial logic (AND-OR-NOT) to form the algorithms. The most basic algorithm consist of applying $E_T$ or $p_T$ threshold to single object. The present Global Trigger is comprised of several VME modules with FPGAs.

The LHC has resumed its operation in 2015, the collision-energy will be increased from 8 TeV to 13 TeV, with the luminosity expected to go up from $0.75 \times 10^{34} \text{ cm}^{-2}\text{s}^{-1}$ to $2 \times 10^{-34} \text{ cm}^{-2}\text{s}^{-1}$. These operating environments will provide new challenges for the CMS trigger system. The CMS Level-1 trigger system will be upgraded to improve its performance for selecting interesting physics events and to operate within the predefined data-acquisition rate [2].
The technical implementation of the upgrades revolves around standardisation and flexibility. The standardisation comes by means of using large, modern FPGAs, which will provide the needed flexibility by allowing the implementation of reconfigurable and highly sophisticated algorithms. This standardisation will help to minimise the future maintenance of the overall system and to share expertise across the Level-1 Trigger group. High-bandwidth optical data communication links also allow the system to be highly reconfigurable to accommodate future detector and electronic upgrades to the CMS experiment.

The Global Trigger will be re-implemented on modern FPGAs on an Advanced Mezzanine Card in MicroTCA crate. The MP7 (Master Processor board, Virtex-7) developed by Imperial College [3] will be used for the upgraded Global Trigger. The upgraded system will benefit from the ability to process complex algorithms with DSP slices and increased processing resources with optical links running at 10 Gbit/s, enabling more algorithms at a time than previously possible and allowing CMS to be more flexible in how it handles the trigger bandwidth. CMS also will be able to match different objects, e.g. muons with jets, with higher resolution and efficiency and be able to calculate more sophisticated quantities such as the mass of a pair of objects. In 2015, CMS plans to keep the present triggers running and commission the new ones simultaneously. Detailed comparisons will be performed between both to test everything from technical implementation to whether the new triggers perform better, before the older system will be turned off.

A software for handling trigger menu implementation on the present system is strongly coupled with the current hardware design. In order to handle the increased complexity of the trigger menu implemented on the upgraded Global Trigger, a set of new software has been developed. The software allows a physicist to define a menu with analysis-like triggers using intuitive user interface. The menu is then realised on FPGAs with further software processing, instantiating predefined firmware blocks. The menu plays a central role in trigger selection and is shared by the second level trigger, known as the High Level Trigger as well as the trigger emulating software in an offline software environment. The menu information is stored in either an XML file or in a database for sharing information with other systems.

2. Level-1 menu grammar

For hiding the underlying hardware implementation as well as to make the data structure of a trigger menu to be flexible for enabling future evolution of the Global Trigger, a set of rules for describing algorithms for the Global Trigger (Level-1 menu grammar) is introduced. The Level-1 menu grammar is mostly defined and parsed by Boost.Spirit [4]. The Boost.Spirit enables a target grammar to be written exclusively in C++ programming language using inline Extended Backus-Normal Form grammar specifications.

An algorithm is comprised of a combination of object, cut, function, external signal and logical operators. An object is a physical object created by the calorimeter and the muon trigger systems, i.e. $\mu$, $e\gamma$, jet, $\tau$, $E_T$, $H_T$, $E_T\text{miss}$ and $H_T\text{miss}$. In the Level-1 menu grammar, each object has its $p_T$ or $E_T$ threshold applied at a certain bunch crossing offset with respect to the bunch crossing ID of a collision. A function is a logical or mathematical computation based on objects. A cut is a condition applied on objects or on the value computed by a function. An external signal is a pre-defined set of binary input signals to the Global Trigger. Table 1 summarises each component.

A set of pre-defined functions are available. There are currently three types of functions, i.e. $\text{comb}$, mass and dist. The $\text{comb}$ function is used for a double-, triple- and quad-objects triggers. For example, a di-objects trigger is expressed as $\text{comb}\{\text{obj1}, \text{obj2}\}$. The mass function is used to compute invariant mass of two objects. The $\text{dist}$ function calculates distance between two objects with specified metric. The $\text{dist}$ function is expressed as $\text{dist}\{\text{obj1}, \text{obj2}\}\{\text{metric}\}$, here metric can be one of $\Delta\eta$, $\Delta\phi$ and $\Delta R = \sqrt{\Delta\eta^2 + \Delta\phi^2}$ with its range specified.
Table 1. Components of the Level-1 menu grammar.

<table>
<thead>
<tr>
<th>Object</th>
<th>Function</th>
<th>Cut</th>
<th>External signal</th>
<th>Logical operators</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\mu$, $e\gamma$, jet, $\tau$, $E_T$, $H_T$, $E_T^{\text{miss}}$, $H_T^{\text{miss}}$</td>
<td>logical or mathematical computation based on objects</td>
<td>condition applied on objects or on the return value of a function</td>
<td>a pre-defined set of binary input signal to the $\mu\text{GT}$</td>
<td>AND, OR, XOR and NOT</td>
</tr>
</tbody>
</table>

The simplest algorithm is an object with $p_T$ or $E_T$ threshold. A muon with $p_T \geq 20$ GeV is expressed as $\text{MU20}$ or $\text{MU.ge.20+0}$, here $+0$ denotes bunch crossing offset. One or more cuts can be applied to an object by appending cuts in square brackets as $\text{MU20[MU-PHI\_UPPER,MU-ETA\_BARREL]}$. Here the cuts $\text{MU-PHI\_UPPER}$ and $\text{MU-ETA\_BARREL}$ should be defined by a user to specify $\phi$ and $\eta$ ranges. Di-muon trigger with $p_T \geq 20$ GeV and $p_T \geq 10$ GeV can be defined using a combination function as $\text{comb}\{\text{MU20,MU10}\}$. Cuts can be applied to a function as well. For example, one can specify invariant mass range of di-muon as $\text{mass}\{\text{MU20,MU10}\}\{\text{MASS\_JPSI}\}$. Topological requirements on multi-object can be applied with dist function.

Due to its abstract nature, an algorithm expressed with the grammar is de-coupled from the details of the underlying hardware implementation. An intermediate software layer handles hardware specific implementation of the menu expressed in the grammar. In this way, the flexibility in the software layer has been newly introduced.

3. Level-1 trigger menu tools

Software libraries have been developed to build menu related tools for implementing a grammar based Level-1 trigger menu on hardware. Figure 1 shows software libraries and tools for menu implementation. Circles in the figure represents three software domains. In online software domain, the menu stored in the online database is used for data taking purpose and the menu is transferred to the offline database with O2O process. In offline software domain, the menu stored in the offline database is used for the emulator in CMS offline software framework (CMSSW) as well as raw data unpacker for the Global Trigger. In the HEPHY domain, the menu will be translated to VHDL code with VHDL producer and the code will be synthesized to build a firmware.

3.1. Trigger Menu Editor

Trigger menu editor provides a graphical user interface for a physicist to edit or to create a Level-1 trigger menu without exposing the complex hardware logic of the Global Trigger. The editor helps a user to express algorithms in the Level-1 menu grammar for composing a menu. Figure 2 shows the use case diagram of the trigger menu editor. Level-1 trigger menus are stored in the CMS online database for persistency. Users can retrieve a menu in an XML format, then use the XML file for developing a menu without accessing the online database which is located in a private network. Expert users can upload a Level-1 trigger menu to the online database using the Web-based graphical user interface for the database.

3.2. VHDL producer

The VHDL producer assembles VHDL codes by populating parameters of VHDL templates according to a Level-1 trigger menu. A template engine for Python, Jinja2 [9], is used to instantiate VHDL template. For accessing the information stored in a Level-1 trigger menu,
common data structures and C++ classes are developed to share the codes between VHDL producer and the emulator in CMSSW, both of which need to retrieve the information from a Level-1 trigger menu. The VHDL producer uses Python binding created by SWIG and the emulator uses shared libraries as external libraries of CMSSW.

4. Summary
The Level-1 Trigger system of the CMS experiment will be upgraded for LHC Run 2, with emphasis on standardisation and flexibility. The Global Trigger will be re-implemented on modern FPGAs on an Advanced Mezzanine Card in MicroTCA crate. For hiding the underlying hardware implementation as well as to make the data structure for a trigger menu to be flexible for enabling future evolution of the Global Trigger, a set of rules for describing an algorithm for the Global Trigger has been introduced. New software tools have been developed to help implementing the event selection algorithms of increased complexity on the Global Trigger.

Acknowledgments
We acknowledge the enduring support for the construction and operation of the LHC and the CMS detector provided by the Bundesministerium für Wissenschaft, Forschung und Wirtschaft (Austria).

References