

# Table of Contents

<b>L1 HF Trigger Emulation.....</b>	<b>1</b>
Triggers.....	1
HF tower count.....	1
HF ring sums.....	1
Running the emulators.....	1
Adjusting parameters.....	1
Accessing the output.....	1
CMSSW 22X/30X.....	1
CMSSW 20X/21X.....	2

# L1 HF Trigger Emulation

This page contains instructions for accessing new HF trigger quantities in the L1 emulator.

## Triggers

### HF tower count

The HF trigger primitive generation sets a single bit for each HF *region* if one of the six towers within the region is over threshold. The number of set bits is counted in the GCT for each of the eta ranges listed below.

0. HF+ ring 1 ( $4.5 < \eta < 5.0$ )
1. HF- ring 1 ( $-5.0 < \eta < -4.5$ )
2. HF+ ring 2 ( $4.0 < \eta < 4.5$ )
3. HF- ring 2 ( $-4.5 < \eta < -4.0$ )

### HF ring sums

The GCT also sums the Et deposited in HF regions in rings around the beampipe for the above ranges of pseudorapidity.

## Running the emulators

The HF quantities are produced by default when running the L1 emulator in all CMSSW versions since CMSSW 200. Instructions for re-running the full L1 Emulator chain (including trigger primitives) can be found at <https://twiki.cern.ch/twiki/bin/view/CMS/SWGuideReRunningL1>

## Adjusting parameters

The feature-bit threshold is set using a unsigned int parameter with LSB = 1/16 GeV. The default value is 1, but can be reset in your job using :

```
replace hcalTriggerPrimitiveDigis.FG_threshold = 32 // for a 2 GeV threshold
```

## Accessing the output

The HF quantities are stored differently in the 22X/30X series from earlier releases.

### CMSSW 22X/30X

The HF bit counts are stored in the L1GctHFBitCounts class, and the ring Et sums are stored in the L1GctHFRingEtSums class. Note that the collections may contain objects from more than one bunch-crossing - in general the user will need to check they have the correct object from the triggered BX. The code snippet below will achieve this.

```
Handle< L1GctHFBitCountsCollection > bitCounts ;
iEvent.getByLabel( "gctDigis", bitCounts ) ;
L1GctHFBitCountsCollection::const_iterator bitCountItr ;
for (bitCountItr=bitCounts->begin(); bitCountItr.bx()==0 && bitCountItr!=bitCounts.end(); ++bitCountItr)

Handle< L1GctHFRingEtSumsCollection > ringSums ;
iEvent.getByLabel( "gctDigis", ringSums ) ;
```

```
L1GctHFRingEtSumsCollection::const_iterator ringSumItr ;
for (ringSumItr=ringSums->begin(); ringSumItr.bx()==0 && ringSumItr!=ringSums.end(); ++ringSumItr)
```

The individual bit counts and ring sums can be accessed using the `L1GctHFBitCounts::bitCount(i)` and `L1GctHFRingEtSums::etSum(i)`, where the `i` argument may run from 0..3 for the 4 eta ranges above.

## CMSSW 20X/21X

The GCT output for HF tower counts and ring sums are transmitted to the GT using bits previously assigned for jet counts. They are therefore saved in the `L1GctJetCount` object (for 18X and 20X series at least). To retrieve this object from the Event, use :

```
Handle< L1GctHfBitCounts > jetCountDigi ;
iEvent.getByLabel( "gctDigis", jetCountDigi ) ;
```

The tower counts and ring sums can be accessed using the `L1GctJetCount::count(int i)` method with arguments as below :

Jet Count	Quantity
6	HFTowerCountPositiveEta
7	HFTowerCountNegativeEta
8	HFRing0EtSumPositiveEta
9	HFRing0EtSumNegativeEta
10	HFRing1EtSumPositiveEta
11	HFRing1EtSumNegativeEta

For convenience, the tower counts and ring sums can also be accessed using the following methods of `L1GctJetCounts` :

```
hfRing0EtSumPositiveEta()
hfRing0EtSumNegativeEta()
hfRing1EtSumPositiveEta()
hfRing1EtSumNegativeEta()
hfTowerCountPositiveEta()
hfTowerCountNegativeEta()
```

The default LSB for the ring sums is 0.5 GeV

-- JimBrooke - 26 Feb 2008

This topic: [CMSPublic > SWGuideL1MinBiasTriggers](#)

Topic revision: r11 - 2009-03-05 - JimBrooke



Copyright &© 2008-2022 by the contributing authors. All material on this collaboration platform is the property of the contributing authors.

or Ideas, requests, problems regarding TWiki? use [Discourse](#) or [Send feedback](#)