MU2E/PHYSICS/xxxxx
version 1.0 November 1, 2023

Commissioning of the Mu2e Data AcQuisition system and the Vertical Slice Test of the straw tracker

11. Mu2e ROC simulation

version 1.0 November 1, 2023

Abstract

This note presents the initial results of the tracker DAQ commissioning.

Contents

1 Notes for the authors **2**

1.1 Revision history 2

2 Introduction to the analysis **3**

3 Monte Carlo simulation **3**

4 Overflow mode: RUN281 **4**

4.1 Time distribution 4

4.2 Occupancy: Number of hits versus channel 4

4.3 Number of hits 6

5 Regular mode: RUN105038 **7**

5.1 Time distribution 7

5.2 Occupancy: Number of hits versus channel 7

5.3 Number of hits 8

6 rate **9**

7 Summary **10**

1 Notes for the authors

1.1 Revision history

- v1.01: initial version

2 Introduction to the analysis

In this note, we present ~~an analysis of the data derived from the readout teststands of the motherboard.~~ initial results of the tracker DAQ commissioning. ~~This analysis was performed with the aim of characterizing the functionality of the Data Acquisition (DAQ) system. A signal generator was employed to send pulses and we tried to understand the output and non-output of the DTC. Our study centered on testing the performance of ROCs and DTCs, actually we were reading 1 ROC (96 channels), which is the equivalent of one panel or 2 ROCs. The analysis was executed employing a single DTC. During the analysis, we had the capability to change different generator's features. We varied the event window duration between successive pulses and modulated the generator's operating frequency. Specifically, we could operate with two distinct frequencies: 31.29 MHz/(2^7+1), resulting in approximately 250 kHz, and 31.29 MHz/(2^9+1), 60 kHz.~~
~~The selection of the event window duration and the frequency played an important role in determining the number of hits per event, considering that the ROC buffer possessed a storage capacity for up to 255 hits. The relationship between the generator and readout counts can be summarized as follows:~~

The test stand included one DRAC card [**?**] and one DTC connected to the DAQ computer, 96 channels in total. The ROC was operated in the data readout mode, and the digi FPGAs were pulsed by the internal pulser. The pulser has two distinct frequencies, 31.29 MHz/(2^7+1) , or approximately 250 kHz, and 31.29 MHz/(2^9+1), or approximately 60 kHz. The ROC firmware has an internal hit buffer which can store up to 255 hits. That should be sufficient for the data taking. Testing therefore could proceed in two different modes:

- the event window is large enough , so the total number of generated hits is greater than 255. In this case the ROC hit buffer always gets filled up, and only the first 255 hits are read out
- the total number of hits within the event window is less than 255. in this case the ROC hit buffer doesn't get filled up and the total number of hits can vary from one event to another.

3 Monte Carlo simulation

To ensure a comprehensive understanding of our system, we initiated a Monte Carlo Simulation of our Data Acquisition (DAQ) system. Given that the maximum allowable number of hits per event is 255, the simulation protocol adheres to the following sequential steps:

- Within a time interval ranging from 0 μs to the reciprocal of the generator frequency, we generated the first event (t_0) by following a uniform distribution;
- After verified that hits from the same channel, are separated by an interval equivalent to the reciprocal of the generator frequency, we proceeded to generate the second event (t_1) by summing this specific time increment to t_0;
- Subsequently, we verified whether the second pulse remained within the predefined time window. If it did, we included this hit in the ongoing event under construction;
- The process of event creation continued until the count of hits reached the maximum threshold of 255, at which point the event construction was terminated.

Furthermore, the simulation accounts for the channel to channel and FPGA to FPGA delays. This comprehensive approach ensures a faithful representation of the real-world operational aspects of the DAQ system.

4 Overflow mode: RUN281

4.1 Time distribution

The analysis started examining the data time distribution of each channel. After a preliminary observation of the distributions, we saw different patterns for channels in the first FPGA and in the second FPGA, as illustrated in Fig. 1: the initial though was the occurrence of a cessation in data acquisition for specific channels at a certain time.

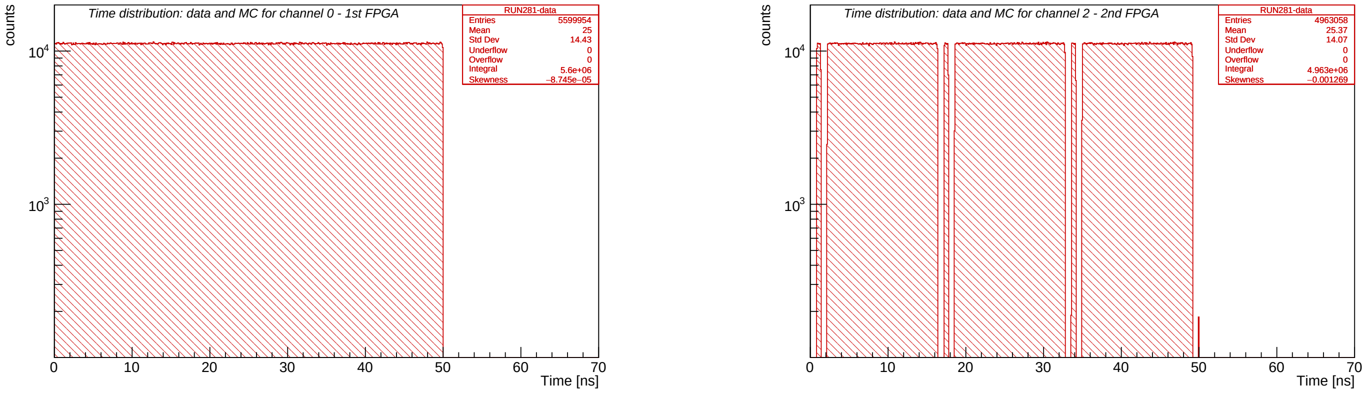

Figure 1: right: First FPGA's channel time distribution, left: Second FPGA's channel time distribution.

We thought it was necessary to characterize the apparatus with a Monte Carlo simulation for our Data Acquisition (DAQ) system, in order to understand the interruptions, so we redirect to section 3 for further information.

4.2 Occupancy: Number of hits versus channel

We conducted an analysis by plotting the number of hits in each channel as a function of the channel number, which revealed a non-uniform distribution. Specifically, channels were arranged in an ascending order, spanning from channel 0 to channel 95. As second step, we managed to generate a histogram with an alternative channel order, as we would expect the filling to occur. Our expectation was to observe a declining trend in the number of hits per channel, primarily due to the progressive filling of the buffer. However, contrary to our initial hypothesis, we encountered a distinctive pattern in which the number of hits exhibited a decrease until reaching zero hits at a specific channel, followed by a subsequent increase beyond the 72nd channel, which corresponds to channel 95.

Upon careful examination, we deduced that the initial channel ordering was incorrect. The first and the second lanes of the second FPGA were inverted and also some channels of the second FPGA. Consequently, we have identified a revised channel sequence, verified also by the Monte Carlo simulation, which is as follows:

FIST FPGA:

lane 1:
91,85,79,73,67,61,55,49,
43,37,31,25,19,13,7,1,
90,84,78,72,66,60,54,48,

lane 2:
42,36,30,24,18,12,6,0,
93,87,81,75,69,63,57,51,
45,39,33,27,21,15,9,3,

SECOND FPGA:

lane 1:
38,44,5,11,17,23,29,35,
41,92,2,8,14,20,26,32,
86,80,74,68,62,56,50,47,

lane 2:
95,89,83,22,16,28,34,40,
46,53,59,65,71,77,10,4,
94,88,82,76,70,64,58,52

At this point we have plotted the number of hits versus the channel number in the order that the filling occurred, as we can see in Fig. 2. We can see a perfect adherence between data and simulation. We can interpret the histogram as it follows: the initial group of channels in the histogram, corresponds to the ones in which we accumulate 4 hits (associated with the first FPGA) and 3 hits (associated with the second FPGA). Subsequently, we observe a contrasting group of channels where the pattern reverses, 3 hits from the first FPGA followed by 4 hits from the second FPGA. We can observe a little jump in the middle and it is due to channel to channel time differences. Time differences between channels are three order of magnitude less than the FPGA ones, so these little jumps are very few, in our case only one appears. Lastly, the concluding cluster of channels is comprised of those collecting 3 hits from the first FPGA and 3 hits from the second FPGA.

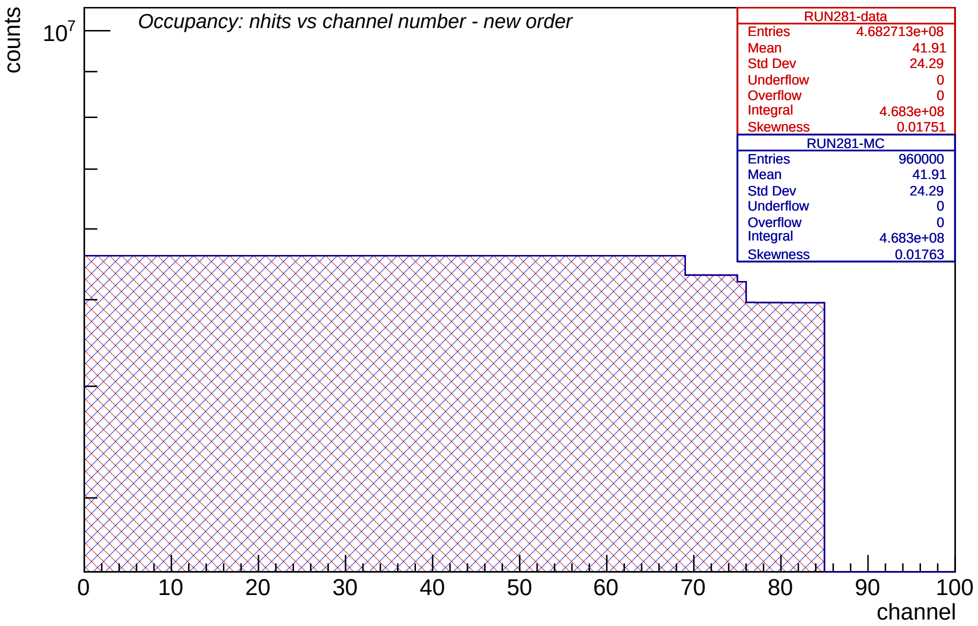

Figure 2: Occupancy: number of hits versus channel. The ordering of channels adheres to the sequence prescribed by the Monte Carlo simulation.

4.3 Number of hits

In conclusion, as in Fig. 3, a key characteristic of the overflow mode becomes evident: the number of hits is concentrated at the maximum value of 255. Consequently, this results in all events possessing the same dimension.

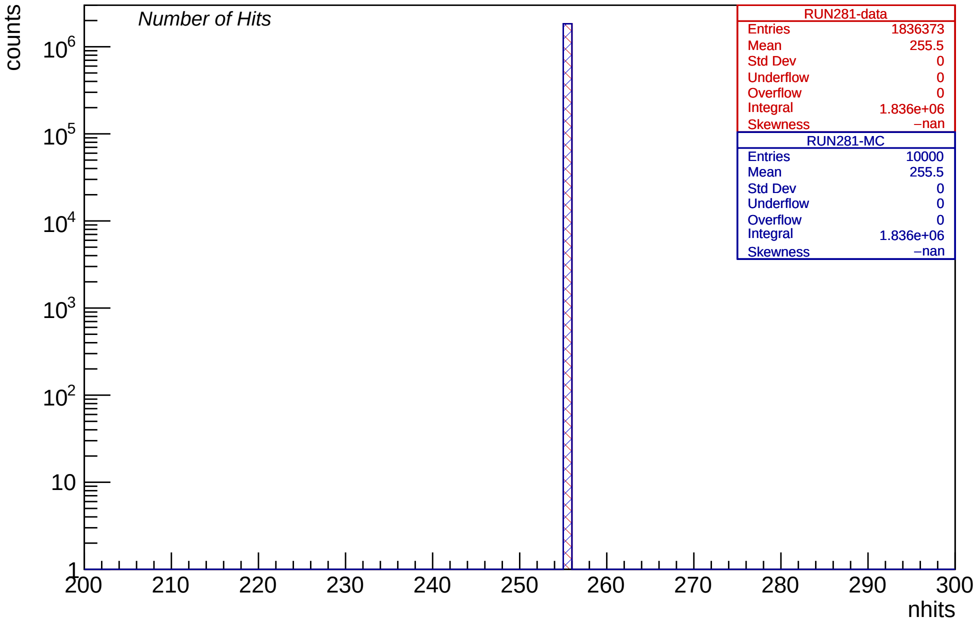

Figure 3: Number of hits distribution.

5 Regular mode: RUN105038

5.1 Time distribution

Similar to the approach outlined in Section 4.1, we commenced our analysis by investigating the temporal distribution of data for channels associated with the first and second FPGAs. Given that we are not operating in an overflow mode, our observations reveal a uniform temporal distribution for both cases.

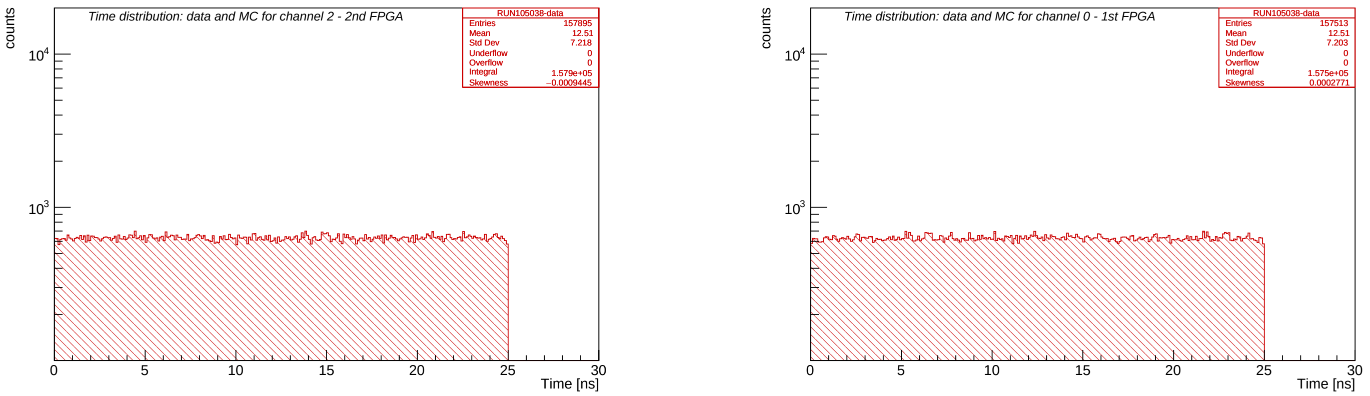

Figure 4: right: First FPGA's channel time distribution, left: Second FPGA's channel time distribution.

5.2 Occupancy: Number of hits versus channel

We conducted an analysis by reproducing the plot that shows the number of hits in relation to the channel number. The occupancy is a uniform distribution, primarily attributable to the fact that we are operating in a non-overflow mode. We can see a perfect adherence between data and simulation.

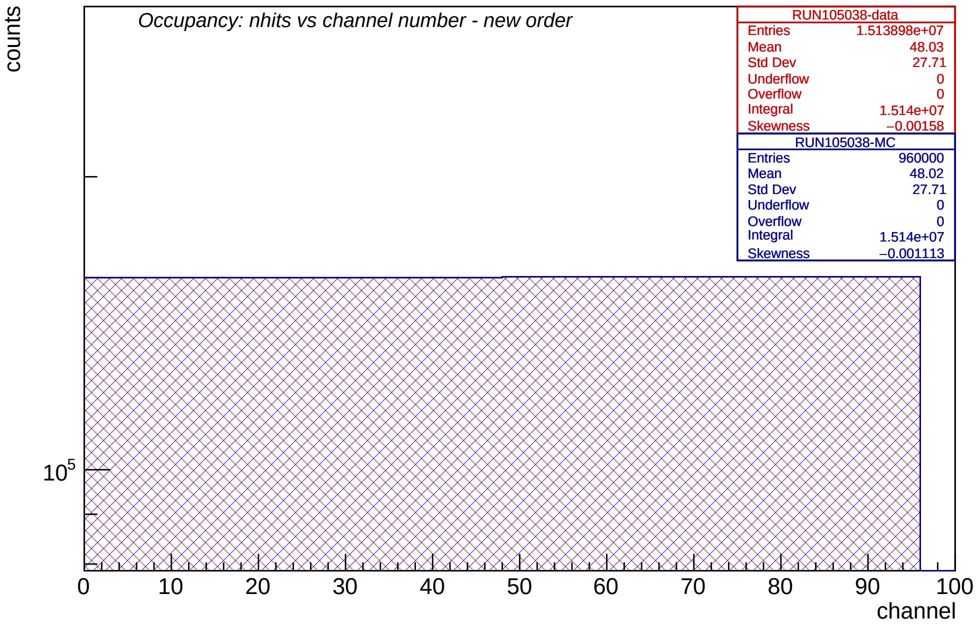

Figure 5: Occupancy: number of hits versus channel. The ordering of channels adheres to the sequence prescribed by the Monte Carlo simulation.

5.3 Number of hits

As conclusion we can see in Fig. 6, the main aspect of non-overflow mode: the number of hits are not anymore peaked in 255 and so this reflects in the fact that the number of bytes are not always the same.

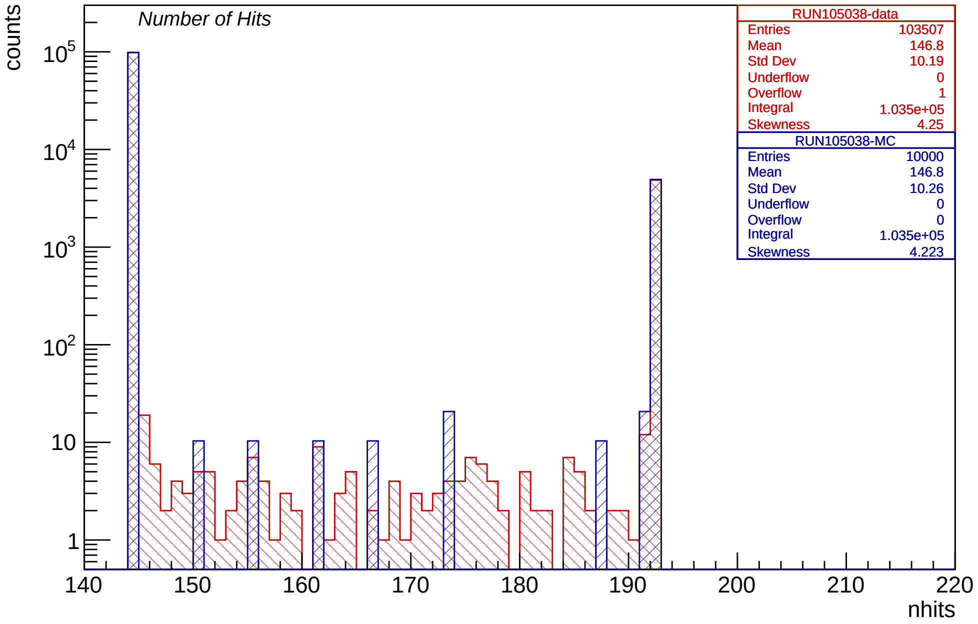

Figure 6: Number of hits distribution.

6 rate

7 Summary

Upper bounds on the direct beam-related backgrounds are as follows:

- background from beam electrons scattered in the stopping target < 1×10^{-3}
- background from muon decay in flights < 1×10^{-3}
- background from beam muons scattered in the stopping target < 1×10^{-5}