
Level-1 Trigger Development for the Multi-band Instability Monitor

By

Joshua ELLIS, School of Physics,
University of Melbourne, Australia

Under the supervision of

Ralph STEINHAGEN, CERN

This work has been sponsored by the Australian
Collaboration for Accelerator Science (ACAS).

21st October 2013

1 Acknowledgements

I would like to first acknowledge all the people who have helped in organising this project, and who have assisted me during my time at CERN. First of all, I wish to thank the Australian Collaboration for Accelerator Science (ACAS) for sponsoring me to go CERN and I have to also thank coordinators of the CERN Summer Student Programme; without them, none of this would have been possible. I must also thank my supervisor, Ralph Steinhagen; not only does he seem to have the answer to everything, but he always made sure that our time at CERN and in Geneva was as good as it could possibly be. Lastly, I also want to thank James Doherty — with whom Ralph and I shared an office — for being great company whilst completing this project, for the work we did together in other small projects, and for all the great times outside of CERN.

2 Introduction

The physics reach and luminosity performance of the Large Hadron Collider (LHC) fundamentally depends on the maximum beam particle intensity that can be injected, accelerated and stored in the accelerator and its proton injectors. The maximum achievable beam intensity and consequently luminosity is limited by self-amplifying beam instabilities due to intrinsic accelerator imperfections [1].

A notable collective interaction is the ‘head-tail’ instability phenomenon which is caused by a resonance condition in circular accelerators, created by the interplay between beam induced transverse wake fields and longitudinal synchrotron oscillation of the particles within a bunch. The effect has been first observed at the ACO and Adone lepton storage rings in Frascati, and independently been theorised by Sands and Pellegrini [2, 3]. The same phenomenon was later directly observed also at the CERN Proton Synchrotron (PS) and Booster (PSB) (both hadron accelerators), and the theory was further formalised by Gareyte and Sacherer [4, 5, 6]. Beside the finite impedances of the beam vacuum chamber that can drive head-tail instabilities, there are a multitude of other effects such as beam-beam interactions, effects related to electron cloud, or other transverse mode coupling instabilities (TMCI) that can drive these ‘intra-bunch’ motion and that are discussed in detail in [1].

Traditionally, intra-bunch or head-tail particle motion is measured in time-domain using fast digitizers operating at speeds of 20 GHz or above to resolve oscillations in the less than one nano-second short particle bunches. However, at these speeds, the effective resolution of even state-of-the-art technology limits the effective intra-bunch position resolution to about 100 μm .

At the LHC, beam particle oscillations at this scale – outside of special machine development studies – cause partial or a total loss of the beam due to the tight constraints imposed on transverse oscillations by the LHC collimation system to protect LHC’s sensitive cryogenic aperture. To improve on the present signal processing, a Multiband-Instability-Monitor (MIM) has been designed, constructed and tested at the CERN Super-Proton-Synchrotron (SPS) and LHC using a new alternate, frequency-domain based diagnostics approach: the system splits the beam signal into multiple equally-spaced narrow frequency bands that are processed and analysed using highly sensitive narrow-band RF detectors [7].

The work has been prepared as part of the CERN summer student project, and its aim is to design a trigger to detect beam instabilities as observed by the MIM during their onset and to also provide an estimate of the lifetime of the instability. The trigger cannot assume information about the growth rate of the instability, and should be able to detect instabilities or damped oscillations in response to an external excitation in the range of 50–50 000 turns.

2.1 Signal Generation

In developing the trigger algorithm, a signal containing instabilities is needed and the start time of these instabilities should be known in order to determine how late the trigger was. This information is not available at the required precision from beam measurements so a simulated signal was used in the development of this trigger.

One possibility would be to use an explicit form to describe the instability, such as eq. (1). This however becomes quickly inefficient if multiple instabilities are desired and if other phenomena are present.

$$y(t) = \sin(2\pi Qt) \times \begin{cases} e^{(t-t_0)/\tau} & t > t_0 \\ 1 & t < t_0 \end{cases} \quad (1)$$

The beam is exposed to a (usually) restoring force keeping it on the designed trajectory. Assuming this restoring force to be linear and symmetric about the designed trajectory, the oscillations of the beam can then be simulated with a harmonic oscillator eq. (2).

$$\frac{d^2y}{dt^2} + 2\zeta(t)\omega\frac{dy}{dt} + \omega^2y = f(t) \quad (2)$$

$\zeta(t)$ can be varied at precise times to make the beam stable or unstable as desired and $f(t)$ can provide sudden kicks and also provides small random variations in $f(t)$ act as a seed from which instabilities can form.

Lastly, Gaussian noise can be added to the signal in order to simulate the instrumental noises associated with detecting the signal’s position.

2.2 Instability Detection

Instabilities are characterised by a growth in the amplitude of the oscillations of the beam. An absolute oscillated amplitude threshold however cannot be used to detect instabilities as they beam may have very large but stable oscillations.

To circumvent this, the growth of the oscillations can be analysed. The idea behind this trigger is to use the standard deviation about the mean of the signal and three averaging windows of varying lengths. In the case of an instability, the oscillations will grow and so will the standard deviation, however the averaging windows will respond to this change with different delays. In particular, a short averaging window will respond quicker than a long average window and the following inequalities should hold:

$$\sigma_{\text{short}} > \sigma_{\text{medium}} > \sigma_{\text{long}} \quad (3)$$

2.2.1 Noise

The trigger will be working on signals which can have a lot of noise associated with them, and therefore looking solely at eq. (3) will not be accurate.

In order to prevent noise from meeting these conditions too often two parameters, α and β , can be introduced in eq. (4). The greater α, β , the harder it is for noise to activate the trigger.

$$\sigma_{\text{short}} - \alpha\sigma_{\text{med}} > 0 \quad (4a)$$

$$\sigma_{\text{med}} - \beta\sigma_{\text{long}} > 0 \quad (4b)$$

In the case of a short instability, eq. (4a) will be satisfied quickly, whilst eq. (4b) may not for a while, so the two conditions can be used independently.

Moreover, even with $\alpha, \beta > 1$, it is possible for the conditions to be met temporarily due to noise; however, it is very unlikely for these conditions to be met over several consecutive turns unless an instability is forming.

2.2.2 Kicks

Kicks present themselves as an instantaneous increase in the oscillations of the beam caused by a known and expected process. Following the kick, the amplitude should decrease as the beam stabilises. Though it is not a primary aim for this trigger, they will also be detected and hopefully identified as such.

Straight after a kick, the conditions eq. (4) will be met, and an “instability” will be detected, however this will be delayed, and presents itself quite differently to an instability as demonstrated later.

To differentiate kicks from instabilities, the absolute value of the signal can be taken into consideration. Directly after a kick, the moving averages will not have had time to respond and therefore the signal will be a lot larger than the standard deviation:

$$|S(t)| \gg \sigma_{\text{short}} \quad (5)$$

Following the same idea as with the instability trigger, a counter can keep track of how often eq. (6) has been met, and at a particular threshold, cause the trigger to activate.

$$(|S(t)| - n\sigma_{\text{short}}) > 0 \quad (6)$$

2.3 Lifetime Estimator

Assuming the instability to be growing exponentially, then the envelope of the signal can be approximated by $ae^{(t-t_0)/\tau}$ and the lifetime estimator should provide a fairly accurate guess for τ .

Given two points, (t_1, y_1) and (t_2, y_2) , an exponential fit through both points will satisfy:

$$ae^{(t_1-t_0)/\tau} = y_1 \quad ae^{(t_2-t_0)/\tau} = y_2 \quad (7)$$

Letting $t_0 = t_1$, we have $a = y_1$ and therefore:

$$y_1 e^{(t_2-t_1)/\tau} = y_2 \quad (8)$$

$$\tau = \frac{t_2 - t_1}{\log(y_2/y_1)} \quad (9)$$

In addition to this basic estimator, a linear regression fit could also be used and would provide a more accurate estimate to the lifetime.

It should also be noted that the estimated lifetime depends greatly on the length of the time window over which the data points are taken and therefore a good method of estimating this length is required.

3 Method

3.1 Signal Generation

The signal generation was originally done with analytical functions, but quickly the harmonic oscillator was used to provide a more realistic signal. It was implemented in C++ using the 4th order Runge-Kutta method.

As the instability grows exponentially, very large amplitudes can be quickly reached so the damping parameter was such that if the oscillations exceed 1 unit, the system is greatly damped. This limits the amplitude up to a maximum of 1 unit and Gaussian noise with a standard deviation of 0.1 units was added.

3.2 Instability Detection

The trigger was developed in C++, originally making use of very large arrays as this made it a lot easier to debug and see what happened at which time using ROOT's plotting abilities, though keeping in mind that the algorithm is not allowed to look forward in time so that eventually, it can be implemented as a running trigger.

Three window size are used in the trigger, and will be referred to as W_{short} , W_{med} and W_{long} . The standard deviation is calculated using the mean and mean squared which have been averaged over W_{short} :

$$\sigma = \sqrt{\langle x \rangle^2 - \langle x^2 \rangle} \quad (10)$$

This is then further averaged out over W_{short} , W_{med} and W_{long} to produce three variables: σ_{short} , σ_{med} and σ_{long} .

As mentioned in section 2.2, two conditions should be met in an instability: eqs. (4a) and (4b). Due to noise however, these conditions need to be met over several turns, and may not both be met at the same time.

A simple counter was initially used as described in algorithm 1, however this did not work very well and caused too many false positives.

The current implementation adds $(\sigma_{\text{short}} - \alpha\sigma_{\text{med}})$ and $(\sigma_{\text{med}} - \beta\sigma_{\text{long}})$ to the counter. Unlike the absolute “true/false” condition, this has the advantage that if the condition is barely met the contribution to the trigger is small, whilst if $\sigma_{\text{short}} \gg \sigma_{\text{med}}$, the trigger will be activated faster.

Additionally, if the conditions are not met the counter is decreased but as with the increase, the rate at which the counter reaches zero depends on how much smaller σ_{short} is to σ_{med} (and similarly with σ_{med} and σ_{long}).

```

foreach new input do
    Calculate  $\sigma_{\text{short}}, \sigma_{\text{med}}, \sigma_{\text{long}}$ ;
    if  $\sigma_{\text{short}} > \alpha\sigma_{\text{med}}$  then
        | Add 1 to counter;
    else
        | Take away 0.5 from counter;
    end
    if counter > threshold then
        | Activate Trigger;
    end
end

```

Algorithm 1: Initial implementation with a discrete counter.

```

foreach new input do
    Calculate  $\sigma_{\text{short}}, \sigma_{\text{med}}, \sigma_{\text{long}}$ ;
    Add  $w_{\alpha}(\sigma_{\text{short}} - \alpha\sigma_{\text{med}})/(\sigma_{\text{short}} + \sigma_{\text{med}})$  to the counter;
    Add  $w_{\beta}(\sigma_{\text{med}} - \beta\sigma_{\text{long}})/(\sigma_{\text{med}} + \sigma_{\text{long}})$  to the counter;
    if counter > threshold then
        | Activate Trigger;
    end
end

```

Algorithm 2: Second implementation with a continuous counter.

This implementation however introduces a sensitivity to the magnitude of σ and the threshold for the counter would need to be adjusted all the time. For this reason, they are weighted and normalised:

$$w_{\alpha} \frac{\sigma_{\text{short}} - \alpha\sigma_{\text{med}}}{\sigma_{\text{short}} + \sigma_{\text{med}}} \quad w_{\beta} \frac{\sigma_{\text{med}} - \beta\sigma_{\text{long}}}{\sigma_{\text{med}} + \sigma_{\text{long}}} \quad (11)$$

So that the trigger works roughly as described in algorithm 2. In this case, the threshold has been set to 1 and w_{α}, w_{β} can be adjusted manually.

In testing the trigger, it has been found that letting $\alpha = \beta$ and $w_{\alpha} = w_{\beta}$ provides quite satisfactory results, though more thorough testing of the effects of varying these parameters independently should be investigated.

3.3 Kicks

For the kicks, the same counter as the instabilities is used (though they could very well be implemented separately). The following is added to the counter if $|S(t)| > n\sigma_{\text{short}}$ for some constant n :

$$\text{counter} \pm \min \left(\frac{|S(t)|}{\sigma_{\text{short}}} - n, k_{\text{cut off}} \right) \quad (12)$$

where $k_{\text{cut off}}$ is some maximum contribution that this can have to the overall counter. This avoids single very large oscillations from instantaneously activating the trigger.

Another possibility which was explored, was to have some function which scales down the contribution of large deviation to the counter, but was still monotonically increasing so that a very large deviations would contribute more than a very small deviations.

$$\text{counter} \pm f\left(\frac{|S(t)|}{\sigma_{\text{short}}}\right) \quad (13)$$

What worked quite well was taking the logarithm, however as this function is used is very frequently, it may use too much processing time and an alternative was thought out. Other slow but monotonically increasing functions, such as taking the n th root, can also be costly to implement, hence a simple cut-off was used.

In order to differentiate kicks from instabilities, one could keep two separate counters, though after the kick trigger, the instability trigger will also be activated and this must be dealt with.

Alternatively, this implementation uses the same counter for both kicks and instabilities. Kicks can be identified as they occur very quickly and generally faster than the smallest of the averaging windows. Knowing this, the trigger looks at the time passed between the counter first starting to increasing, and the counter activating the trigger. If that time is smaller than W_{short} , it must be a kick since σ_{short} cannot have responded quickly enough. This does however require that $\text{max contrib} < \text{threshold}/W_{\text{short}}$.

3.4 Lifetime Estimate

The 2-point lifetime estimator described in section 2.3 has been used as the linear regression fit has not yet been implemented. When the trigger is activated, the 2-point lifetime estimate is made between σ_{short} at the time of the trigger and every σ_{short} in a buffer beforehand.

The length of the buffer is estimated by looking at the counter. Once the counter is seen to be increasing, the time is recorded, and then the time between the trigger and this time corresponds to how quickly the trigger was activated. This also corresponds roughly to the growth rate of the instability as it is linked to the separation between σ_{short} , σ_{med} and σ_{long} .

A drawback of the current 2-point lifetime estimator is that it is subject to biases due to first point being used for every estimate. This would not be an issue with a linear regression fit.

3.5 Dead Time

When the counter reaches the threshold to activate the trigger, the counter is reset to zero so that it is ready for the next instability; however, this caused issues as the instability which activated the trigger is still present and will fill the counter very quickly causing it to trigger multiple times. Instead of having a nice clean signal at an instability, the trigger is continuously being activated.

In order to stop this, the activation of the trigger deactivates the trigger and the trigger is only activated when the conditions are false once again. Since σ_{med} and σ_{long} respond more slowly, the trigger is reactivated once $\sigma_{\text{med}} < \sigma_{\text{long}}$.

4 Results

The algorithm was tested on both simulated signals and on real data. In the development itself, the parameters were varied using trial and error. For example, the parameters for the noise in eq. (4) were simply tested over a noisy but instability-free signal and changed until the number of false positives reached an acceptable level. Once the algorithm appeared to work, a more systematic way of testing for these parameters was developed.

4.1 Simulated Signals

Figure 1 shows the trigger working over a large simulation with multiple instabilities and a region of only noise. The trigger is only activated when the counter reaches 1, though as can be seen, noise causes the counter to increase at times.

In fig. 2, the kick is activated very quickly, and it is clear that σ_{short} has only just had time to increase by the time the trigger is activated. This is why the signal itself is considered in some cases and not solely σ_{short} , σ_{med} and σ_{long} . As for figs. 3 and 4, the trigger activates before the oscillations have reached very large amplitudes.

For figs. 1 to 4, the parameters are described in table 1

Parameter	Value
W_{short}	10
W_{med}	200
W_{long}	2000
$w_{\alpha} = w_{\beta}$	0.1
$\alpha = \beta$	1.07
$k_{\text{cut off}}$	0.1
n	5

Table 1: Parameters used in figs. 1 to 4.

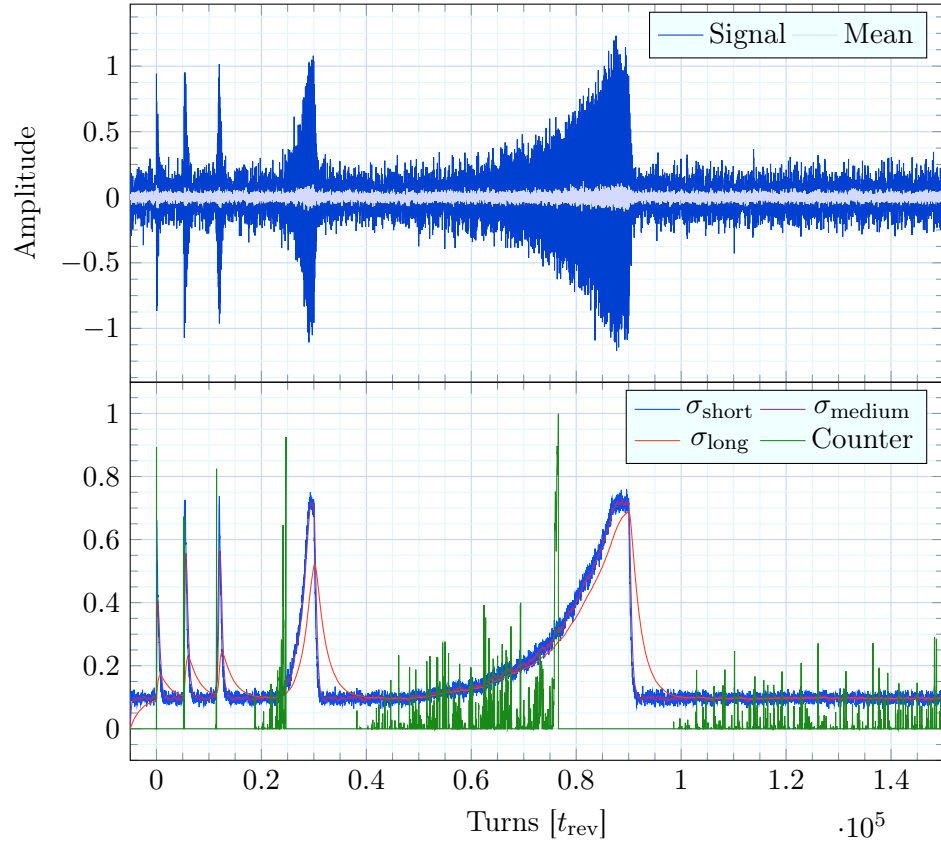


Figure 1: The trigger over various lengths instabilities and some noise at the end. The trigger is activated when the counter reaches 1 (though due to the aliasing, it may not reach 1 on the graph). The dead times can be identified easily as the regions where the counter is perfectly 0.

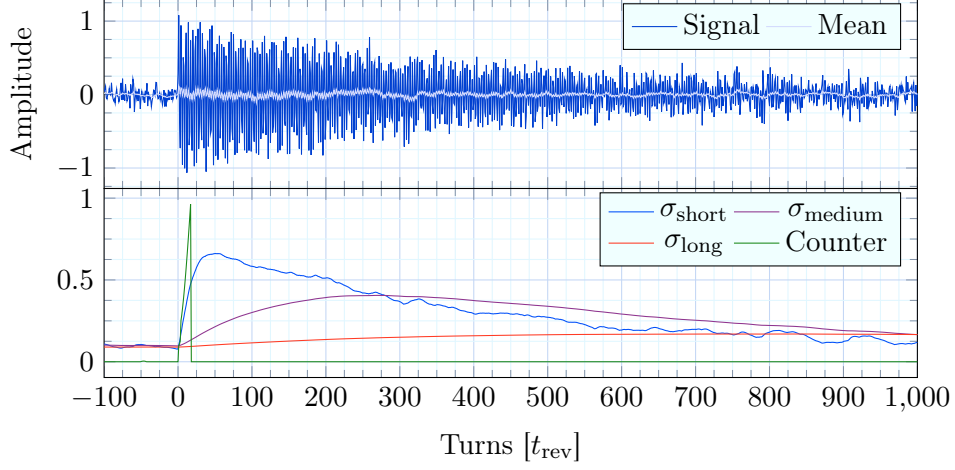


Figure 2: Kick taken from fig. 1. Trigger delay: 18 turns.

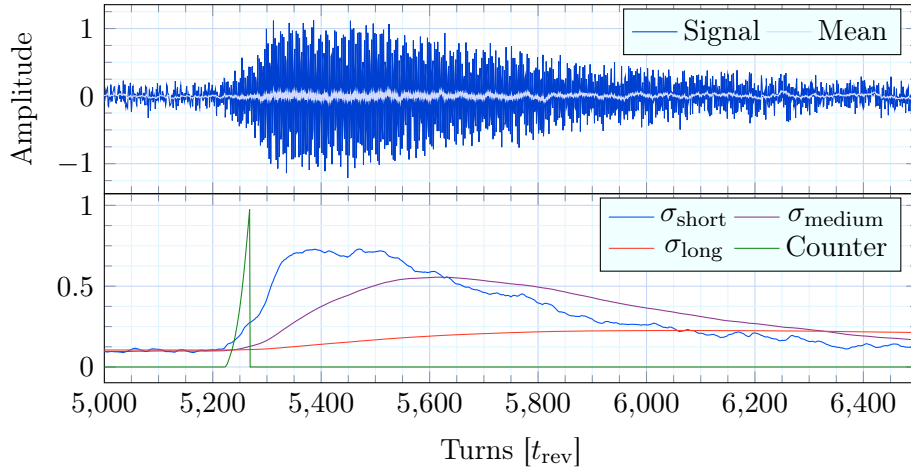


Figure 3: Short instability taken from fig. 1. Instability starts at y -axis. Trigger delay: 269 turns, Estimated lifetime: 147 turns.

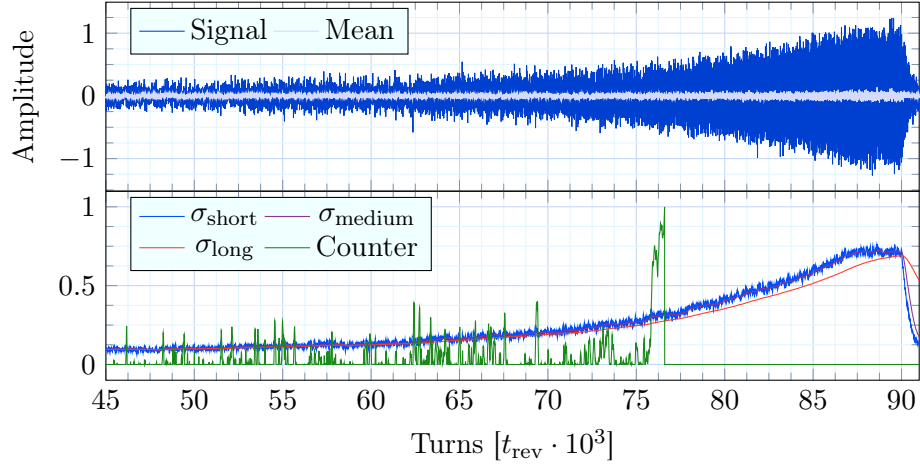


Figure 4: Long instability taken from fig. 1. Instability start at y -axis. Trigger delay: 31 600 turns, Estimated lifetime: 10 342 turns.

4.2 Real Signal

After the trigger was seen to work quite well on the generated signals, it was used on a real signal. An immediate observation is that the time scales involved are a lot larger. The simulation from fig. 1 had a total of $\sim 10^5$ turns, whilst the sample data is an order of magnitude longer. Additionally, as can be seen in fig. 7, the amplitude of the signal even outside of instabilities can vary considerably — a feature which was not simulated in the generated signal. To prevent each of these variations from activating the trigger, the small window had to be increased, though this is at the cost of not being able to detect short instabilities, unless they actually activate the kick trigger.

The parameters used were:

Parameter	Value
W_{short}	3000
W_{med}	10 000
W_{long}	100 000
$w_{\alpha} = w_{\beta}$	0.1
$\alpha = \beta$	1.07
$k_{\text{cut off}}$	0.1
n	5

The power spectrum of the studied signal can be seen in fig. 5, where the instability can be seen just above the main tune line, starting at about 2:7:0 and increasing in amplitude. The instability is allowed to grow due to reduced octupole settings and is initiated by a frequency sweep. The frequency sweeps also cause tune resonances.

The triggers can be further adjusted, but the signal can also be changed. Figure 8 is the same signal as fig. 6 but keeping only the frequencies around the tune ($Q \in [0.3, 0.308]$); the tune resonance is clearly visible and the instability is no longer present. The resonance is detected, though a few false positives are detected as well which seem to coincide with where the instability was before the band-pass filter.

Figure 9 shows the same signal again, but this time keeping only tunes above the main tune line in the range of $[0.308, 0.5]$. The tune resonance is no longer seen and the instability is clearly visible. The trigger finds the trigger easily with only a single false positive. Interestingly, the tune resonance is still detected even though it is not visible.

Looking at different frequency bands and the correlation between them may provide a way of identifying false positives. Additionally, what may be a higher concentration of false positive in one band may indicate the presence of an instability in some other frequency band, though this will need to be further tested.

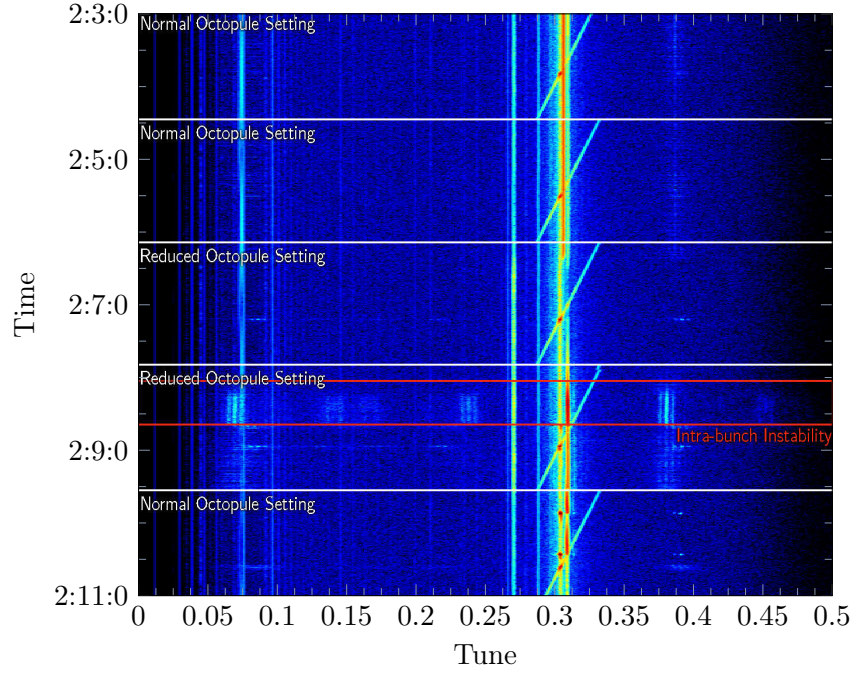


Figure 5: The power spectrum of the signal shown in figs. 6 to 9. The LHC tune line is seen just above $Q = 0.3$ and the diagonal lines crossing it is a frequency sweep to try and excite the beam. The first two panels have normal octupole setting and other than a tune resonance, the frequency sweep does nothing. In the second pane, the octupole settings are reduced and beam is less stable. At about 2:7:0, an instability forms just above the main tune line and keeps growing until resonance lines start to be seen in other parts of the spectrum.

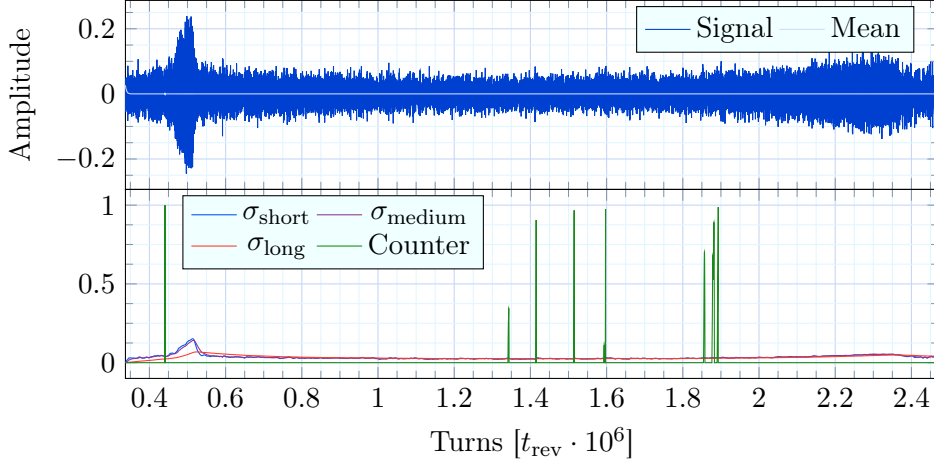


Figure 6: The trigger on real unfiltered data from the LHC. There are two instabilities: the tune resonance near $0.5 \cdot 10^6$, and an actual instability culminating near $2.2 \cdot 10^6$. The trigger easily finds the resonance, and does trigger properly at about $1.9 \cdot 10^6$.

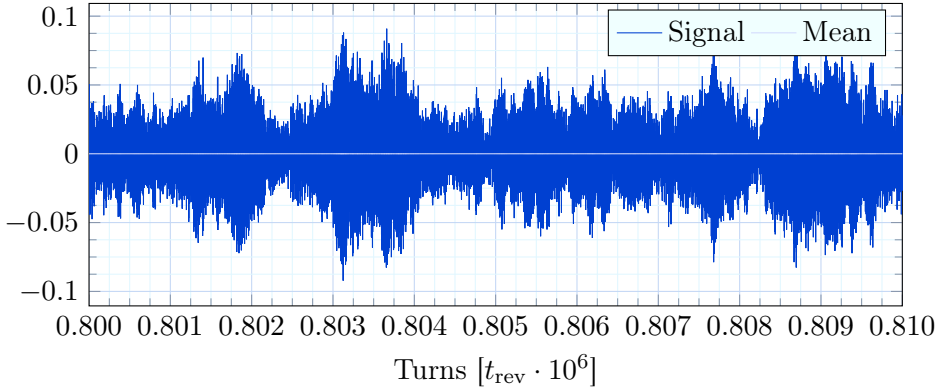


Figure 7: A subset of the signal, showing variations in the amplitude of the signal of such a form that they were not accounted for in the simulated signals.

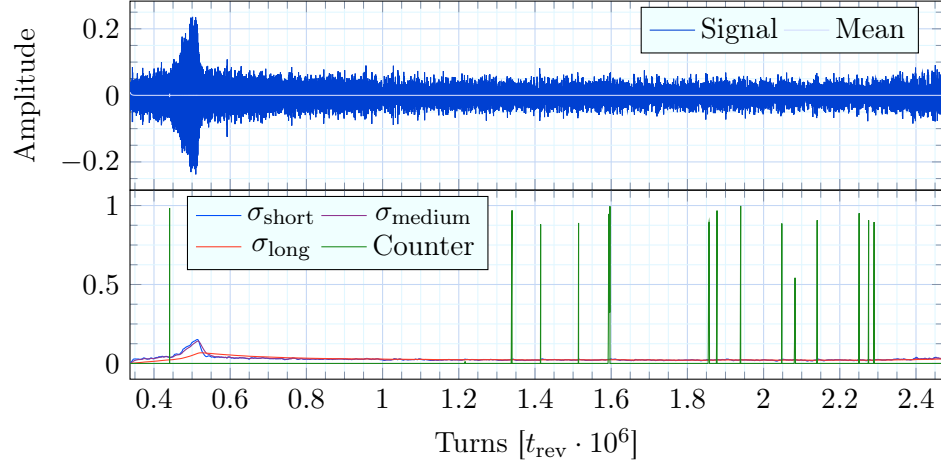


Figure 8: The same as fig. 6, but keeping only frequencies in range $Q \in [0.3, 0.308]$. The tune resonance is still there, whilst the instability has disappeared.

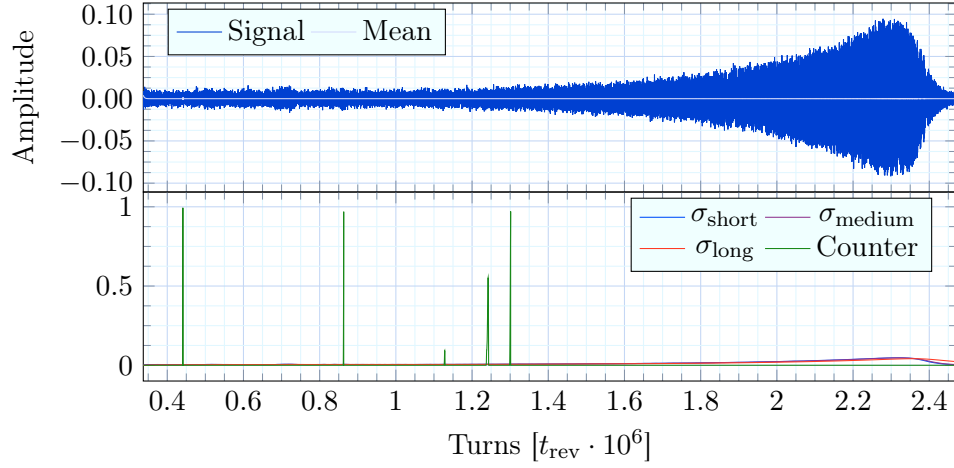


Figure 9: The same data as fig. 6, but selecting only tunes just above the main tune line: $Q \in [0.308, 0.5]$. The tune resonance is not longer visible, though it still causes the trigger to activate. The trigger properly activates at $1.3 \cdot 10^6$.

5 Conclusion

This project's aim was to design a trigger to detect beam instabilities as soon as possible after they have started and also provide an estimate to the lifetime of the instability without assuming any information about the growth rate of the instability, and should be able to detect a wide range of instabilities.

As was demonstrated, it is possible to develop a trigger which fulfils these features, though depending on the shape of the signal, it may be harder to detect very fast instabilities.

It was also seen that filtering out undesired frequencies, such as the main tune line, can reduce the number of false-positives and make instabilities a lot clearer and easier for the trigger to detect. Additionally, it appears that comparing different band may be advantageous as false positive in one frequency band may be linked to an instability in some other band, though this observation has not been verified.

References

- [1] Stephen Myers ed. and Herwig Schopper ed., "Elementary Particles - Accelerators and Colliders – Chapter 4", Landolt-Bornstein: Numerical Data and Functional Relationships in Science and Technology - New Series / Elementary Particles, Nuclei and Atoms, Vol. 21, Springer, London, ISBN-9783642230523, 2013
- [2] M. Sands, "The Head-Tail Effect: An Instability Mechanism in Storage Rings", SLAC-TN-69-008 & SLAC-TN-69-010, SLAC, 1969,
- [3] C. Pellegrini, "On a New Instability in Electron-Positron Storage Rings (The Head-Tail Effect)", Nuovo Cim. A64 (1969) 447-473
- [4] J. Gareyte, "Head-Tail Type Instabilities in the PS and Booster", CERN-MPS-BR-74-7, CERN, 1974
- [5] F. J. Sacherer, "Transverse Bunched Beam Instabilities - Theory", CERN-MPS-INT-BR-74-8, CERN, 1974,
- [6] F. J. Sacherer, "Transverse Bunched Beam Instabilities", CERN/PS/BR 76-21, CERN, 1976,
- [7] R. J. Steinhagen, M.J. Boland, and T. G. Lucas, "A Multiband-Instability-Monitor for High-Frequency Intra-Bunch Beam Diagnostics", IBIC'13, Oxford, 2013