# Table of Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Emulating Trigger Prescales in MC</td>
<td>1</td>
</tr>
<tr>
<td>Documentation</td>
<td>1</td>
</tr>
<tr>
<td>Getting TCKPrescaleEmulator</td>
<td>1</td>
</tr>
<tr>
<td>Configuring TCKPrescaleEmulator</td>
<td>1</td>
</tr>
<tr>
<td>IMPORTANT: Warnings</td>
<td>1</td>
</tr>
<tr>
<td>The TCK you want to emulate contains lines that don't exist in the MC.</td>
<td>2</td>
</tr>
<tr>
<td>The TCK you want to emulate contains lines with prescales higher than those in MC</td>
<td>2</td>
</tr>
<tr>
<td>The MCTCK changes mid-run</td>
<td>2</td>
</tr>
<tr>
<td>While running a line is found that wasn't in the data TCK</td>
<td>2</td>
</tr>
<tr>
<td>Advanced usage</td>
<td>3</td>
</tr>
<tr>
<td>Still to be implemented</td>
<td>3</td>
</tr>
</tbody>
</table>
**Emulating Trigger Prescales in MC**

The TCKs used in MC do not use prescales. In order to reliably determine trigger efficiencies in MC for direct comparison to data these prescales need to be emulated.

TCKPrescaleEmulator is a DaVinci algorithm that tries to do this. Give it a TCK corresponding to what was run in the pit for your data sample, and it will automatically rescale trigger decisions in the MC sample you run through it.

**Documentation**

This page serves as complete and current documentation. For a usage example with B->J/psi K candidates see the complementary LHCb internal note:

* LHCb-INT-2012-031

**Getting TCKPrescaleEmulator**

SVN revision 147252 of svn+ssh://svn.cern.ch/reps/lhcb/LHCb/trunk/Kernel/LHCbAlgs is known to work.

```
SetupProject DaVinci --build-env
getpack Kernel/LHCbAlgs head
cd Kernel/LHCbAlgs/cmt
cmt br cmt make
```

Make sure you're using the right platform when compiling on lxplus: It won't run on the grid otherwise. Best way to do this is perform the first 3 lines and then make in ganga using

```
j.application.make()
```

LHCbAlgs contains a lot of algorithms, so be prepared to wait a while for it to compile.

**Configuring TCKPrescaleEmulator**

Add the following to your main sequence:

```
from Configurables import TCKPrescaleEmulator

#emulate TCK:
SeqTCKEmu = TCKPrescaleEmulator("TCKPrescaleEmu");
SeqTCKEmu.TCKtoEmulate=0x790038
MyMainSequence.Members.append(SeqTCKEmu)
```

That's it! Substitute the TCK you want to emulate for 0x790038 above. Anything before SeqTCKEmu in your sequencer will see the unmodified HltDecReports, anything after will see the emulated ones. You can run eg: two instances of_decayTreeTuple to get the decisions with and without emulation.

**IMPORTANT: Warnings**

TCKPrescaleEmulator produces warnings for a reason. Don't ignore them! There are several conditions that you need to be aware of:
The TCK you want to emulate contains lines that don't exist in the MC

TCKPrescaleEmulator can't make decisions, it can only downscale decisions that have already been made. If you see the warning:

THIS TCK IS INCOMPATIBLE WITH MC! LineName IS IN THE TCK, BUT NOT IN THE MC!
I CANNOT EMULATE THIS LINE!

or:

THIS TCK IS INCOMPATIBLE WITH MC! PRESCALE FOR LineName IN MC IS 0.0 IN THE TCK IT IS: X
I CANNOT PRESCALE A DECISION THAT IS NEVER THERE!

Tread carefully. If the line is one you know for a fact your TisTos decision doesn't depend on, you can continue, but beware that HltXPhys_TIS uses all physics lines. Use a combination of lines manually instead if you see either of these.

The TCK you want to emulate contains lines with prescales higher than those in MC

As above, if a line in MC is prescaled harder than in data it can't be recovered. TCKPrescaleEmulator will not apply a further prescale, so if your TCK wants a prescale of 0.5 and MC was prescaled to 0.3, you'll get back the 0.3 that was already there. You'll see this warning:

THIS TCK IS INCOMPATIBLE WITH MC! PRESCALE FOR LineName IN MC IS 0.3 IN THE TCK IT IS: 0.5
NOT PRESCALING, RESULTS WILL BE APPROXIMATE!

Again, tread carefully. If this isn't a line you care about or that your other lines depend on you might be ok, but global decisions may be affected.

The MC TCK changes mid-run

If you combine MC samples into the same run for some reason, TCKPrescaleEmulator will handle the change and recompute the prescales for the new TCK. You should keep an eye out for the above warnings again though, so it will flag that the TCK has changed with the following message:

*************** DANGER ****************
THE TCK HAS CHANGED!!!!!!!!!!!!!!!
PREV TCK: XXXXXXXX
NEW TCK : YYYYYYY
MAKE SURE YOU EXPECTED THIS TO HAPPEN
*************** DANGER ****************

After this point watch for other warnings.

While running a line is found that wasn't in the data TCK

If you see the warning

DecReport found for line LineName but no prescaler found! Killing this line

Don't worry unless it's a line you wanted to study- it means that at initialisation this line wasn't found in the data TCK, so no prescaler was initialised. Because this line is never in the data, it's fine for it to be just zeroed in the MC. This should probably be an INFO instead of a WARNING as it isn't critical, just handy to know.
Advanced usage

To kill specific lines that would otherwise run, use the following:

```
SeqTCKEmu.LinesToAlwaysKill=\["LineName1","LineName2".....\]
```

If you want to read and write HltDecReports in a location that is not the default, use the following:

```
SeqTCKEmu.HltDecReportsLocation=/Event/Somewhere/
```

Still to be implemented

-Pre/Post scale emulation for L0? Is the L0 ever scaled?

-Rate limited lines: Find them, warn the user and maybe even try to rate limit them correctly?

-- ConorFitzpatrick - 21-Oct-2012