

## Using the data

\* The ToF data are stored in the AOD containers

```
#include <xAODForward/AFPToFHitContainer.h>
#include <xAODForward/AFPToFHit.h>
```

The ToF hit information is retrieved as:

```
const xAOD::AFPToFHitContainer* afpToFHitContainer = 0;
ANA_CHECK(evtStore()->retrieve( afpToFHitContainer, "AFPToFHitContainer" ) );
for (const xAOD::AFPToFHit* tofhit : *afpToFHitContainer) {
    int    station = tofhit->stationID();
    float  time    = tofhit->time();
    int    train   = tofhit->trainID();
    float  length  = tofhit->pulseLength();
    int    hptdcid = tofhit->hptdcID();
    bool   isA     = tofhit->isSideA();
    bool   isC     = tofhit->isSideC();
    int    channel = mapToFChan(tofhit->hptdcID(), tofhit->hptdcChannel());
    int    bar     = channel%m_nBarsPerTrain;

    // ... some code of yours here
}
```

where

```
const int m_nBarsPerTrain = 4;
```

and the **mapToFChan** function is implemented as (*functional in Run 2, I do know how about Run 3*):

```
int mapToFChan(int hptdcID, int hptdcChannel){
    if(hptdcID==1) {
        if    (hptdcChannel==0) return 0;
        else if(hptdcChannel==2) return 6;
        else if(hptdcChannel==3) return 3;
        else if(hptdcChannel==5) return 5;
        else if(hptdcChannel==6) return 2;
        else if(hptdcChannel==8) return 4;
        else if(hptdcChannel==9) return 1;
        else if(hptdcChannel==11) return 7;
        else return 16;
    } else if (hptdcID==2){
        if    (hptdcChannel==0) return 8;
        else if(hptdcChannel==2) return 14;
        else if(hptdcChannel==3) return 11;
        else if(hptdcChannel==5) return 13;
        else if(hptdcChannel==6) return 10;
        else if(hptdcChannel==8) return 12;
        else if(hptdcChannel==9) return 9;
        else if(hptdcChannel==11) return 15;
        else return 16;
    } else return 16;
}
```

The **time** stored in AOD is in nanoseconds although it is an uncalibrated HPTDC output (raw time). Check the implementation in

```
float xAOD::AFPToFHit_v1::time( ) const
```

The raw time in terms of one of the 1024 a HPTDC bin numbers (0 .. 1023) is translated to nanoseconds as **time [ns] = rawbin \* (25 [ns] / 1024)**. The 25 ns is 1 / 40 MHz LHC interval. This means if you want the

HPTDC time bin number you need to do **rawbin = time [ns] / (25 [ns] / 1024)**

-- KarelCerny - 2022-08-10

---

This topic: Sandbox > KarelCernySandbox

Topic revision: r1 - 2022-08-10 - KarelCerny



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](#)