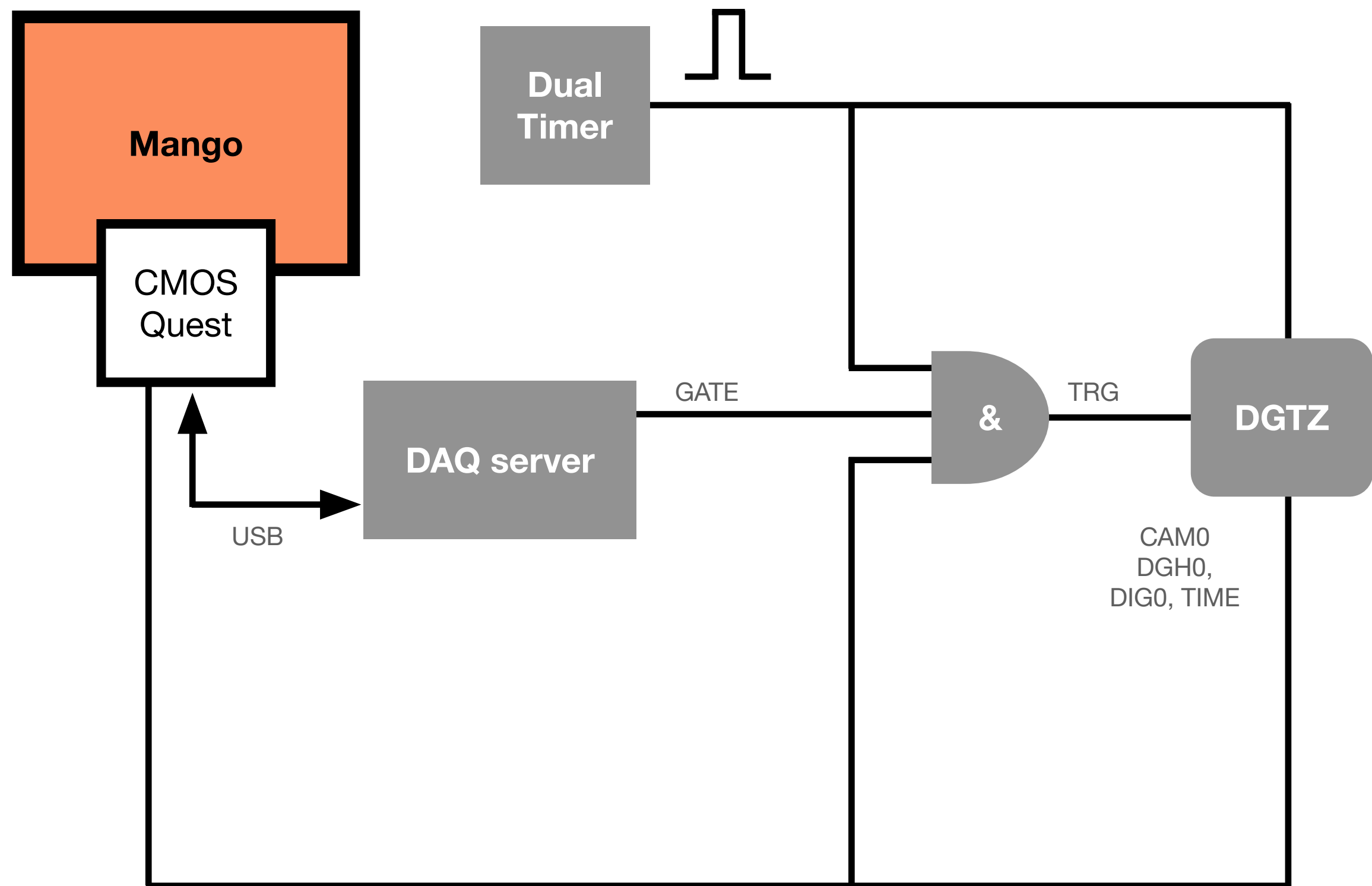


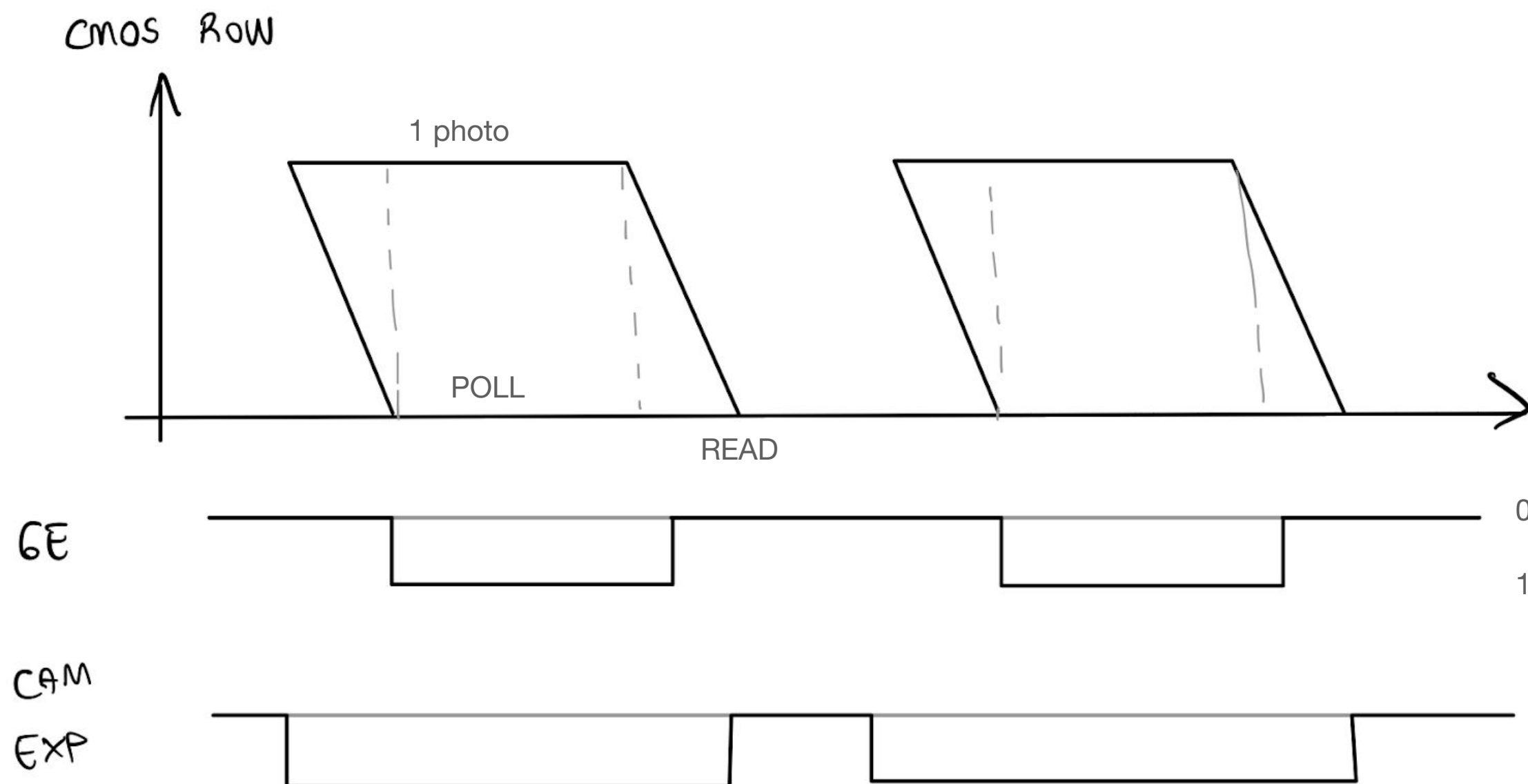
Mango DAQ update

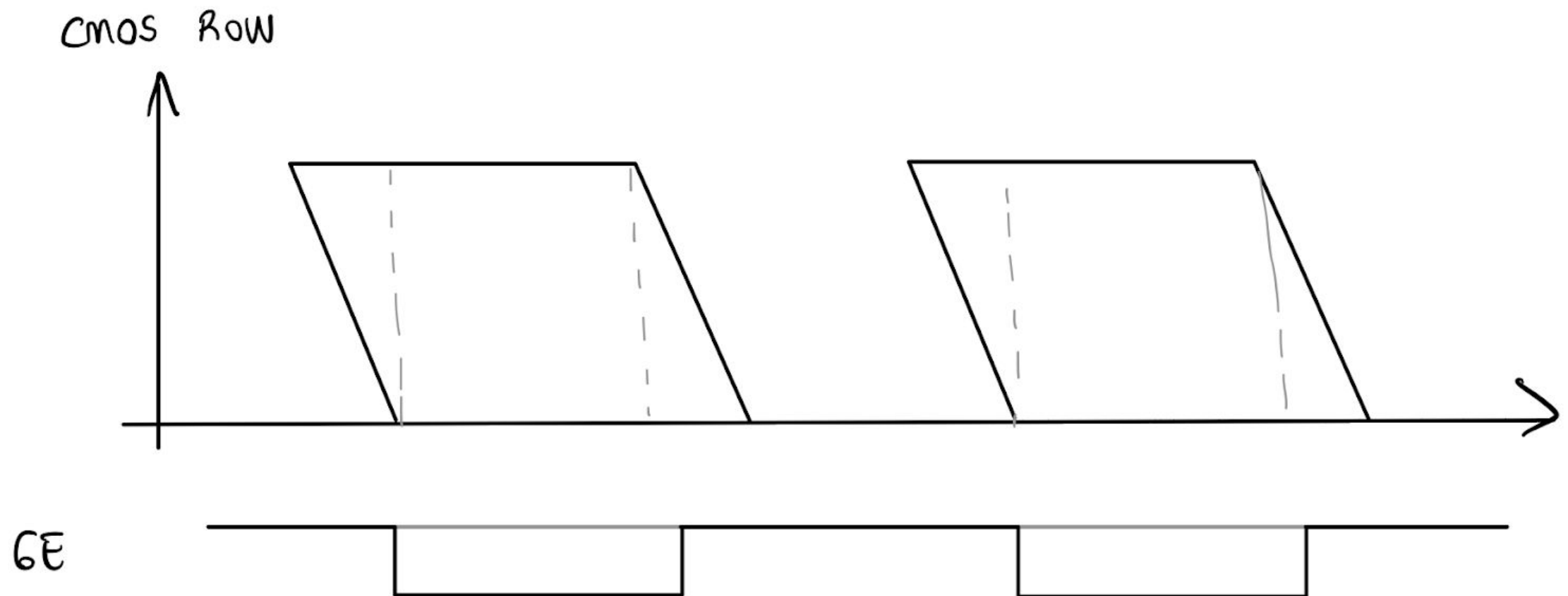
Towards continuous imaging

DAQ Scheme



Before

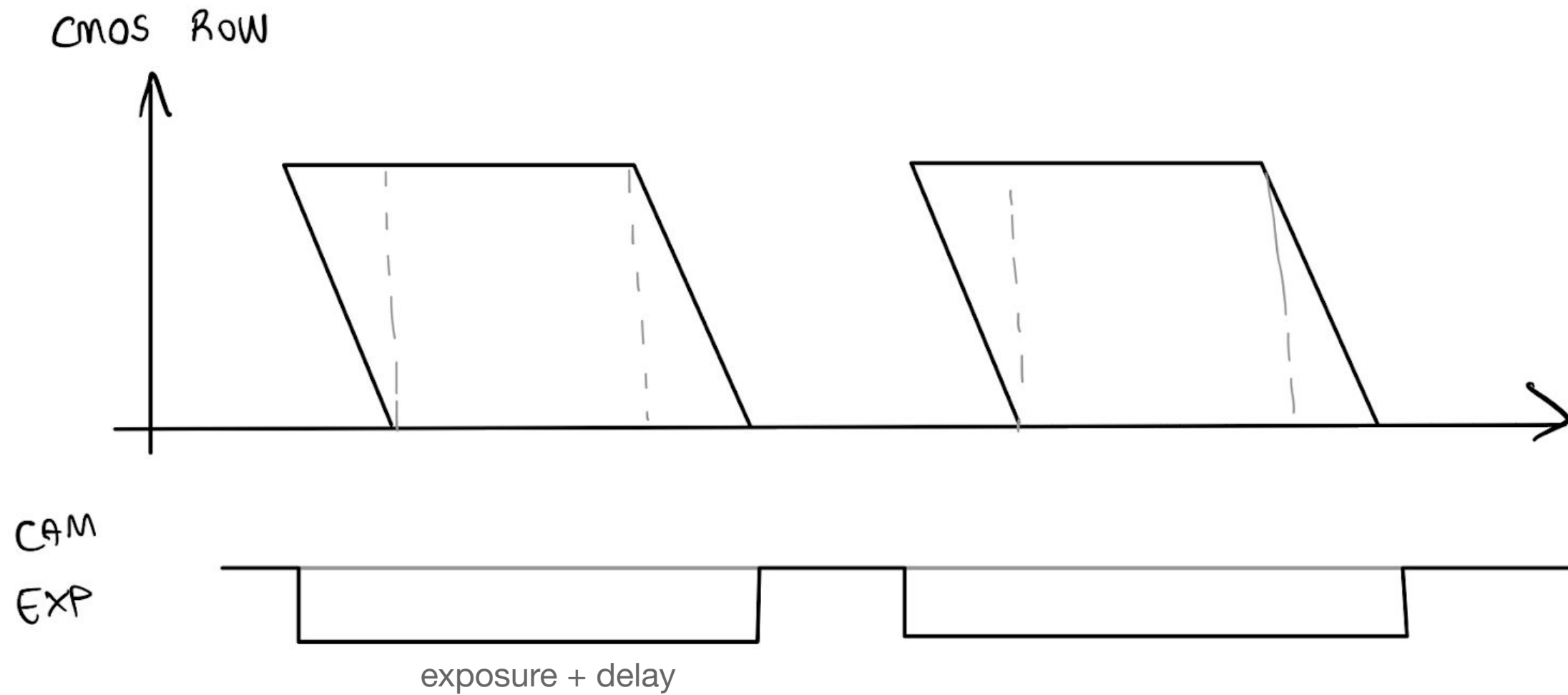




global exposure output 1

```
//Global exposure as output signal with positive polarity
err = dcamprop_setvalue( gCam, DCAM_IDPROP_OUTPUTTRIGGER_KIND, DCAMPROP_OUTPUTTRIGGER_KIND__EXPOSURE );
if(failed(err)) cout << "ERROR IN DCAM_IDPROP_OUTPUTTRIGGER_KIND" << endl;

err = dcamprop_setvalue( gCam, DCAM_IDPROP_OUTPUTTRIGGER_POLARITY, DCAMPROP_OUTPUTTRIGGER_POLARITY__POSITIVE );
if(failed(err)) cout << "ERROR IN DCAM_IDPROP_OUTPUTTRIGGER_POLARITY" << endl;
```



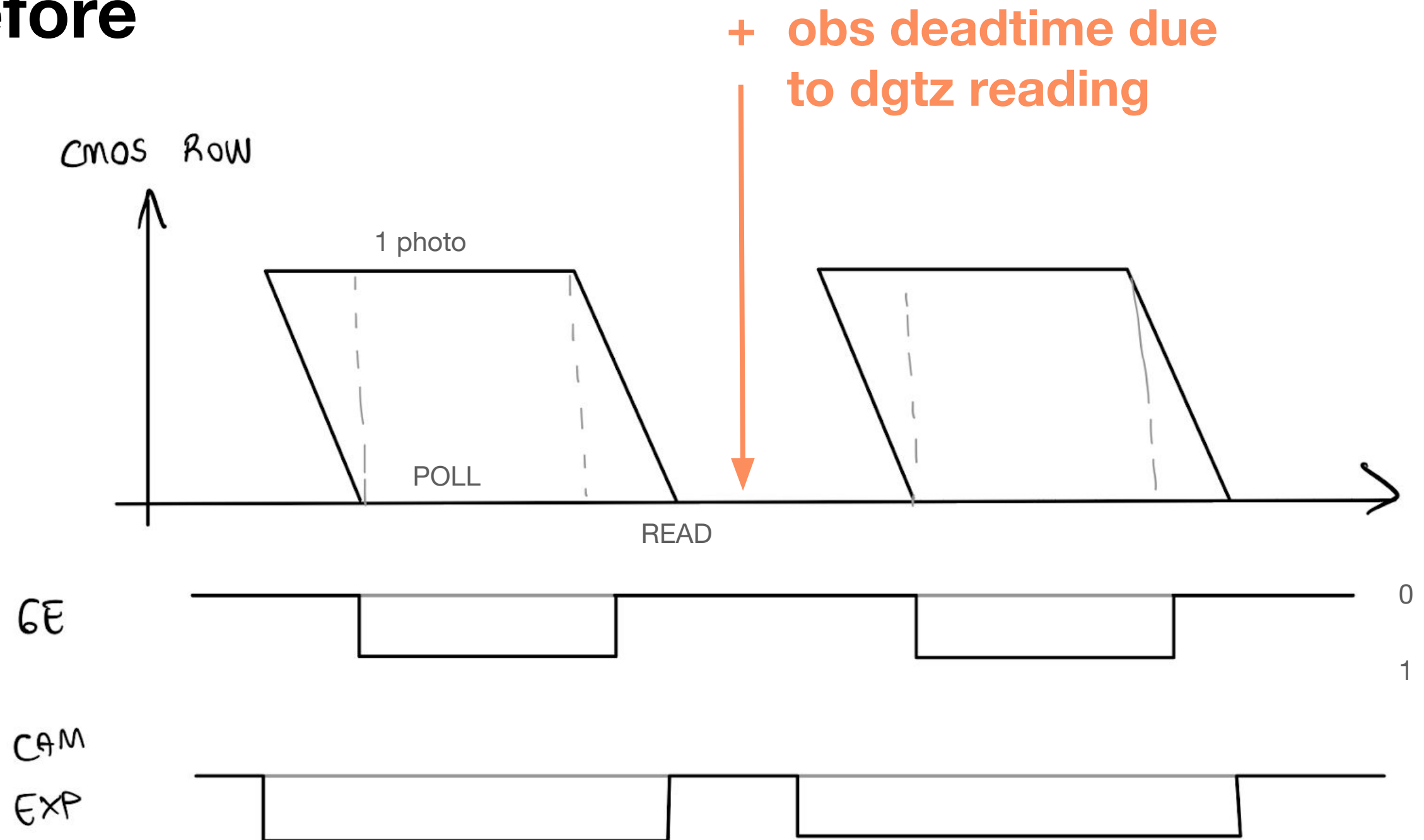
camera exposed as programmable output

```
err = dcamprop_setvalue( gCam, DCAM_IDPROP_OUTPUTTRIGGER_SOURCE + 256, DCAMPROP_OUTPUTTRIGGER_SOURCE__TRIGGER);
if(failed(err) && DEBUG) cout << "ERROR IN DCAM_IDPROP_OUTPUTTRIGGER_SOURCE + step" << endl;

//err = dcamprop_setvalue( gCam, DCAM_IDPROP_OUTPUTTRIGGER_ACTIVE + 256, DCAMPROP_OUTPUTTRIGGER_ACTIVE__EDGE);
//if(failed(err) && DEBUG) cout << "ERROR IN DCAM_IDPROP_OUTPUTTRIGGER_PERIOD + step" << endl;

err = dcamprop_setvalue( gCam, DCAM_IDPROP_OUTPUTTRIGGER_PERIOD + 256, exposure + delay);
if(failed(err) && DEBUG) cout << "ERROR IN DCAM_IDPROP_OUTPUTTRIGGER_PERIOD + step" << endl;
```

Before



40% deadtime on the fusion camera

➡ **TASK:** develop continuous imaging

NEW

[DAQ Github](#) branch: **mango2025**

set **mode = 3** in the run settings/ODB to use
continuous imaging mode

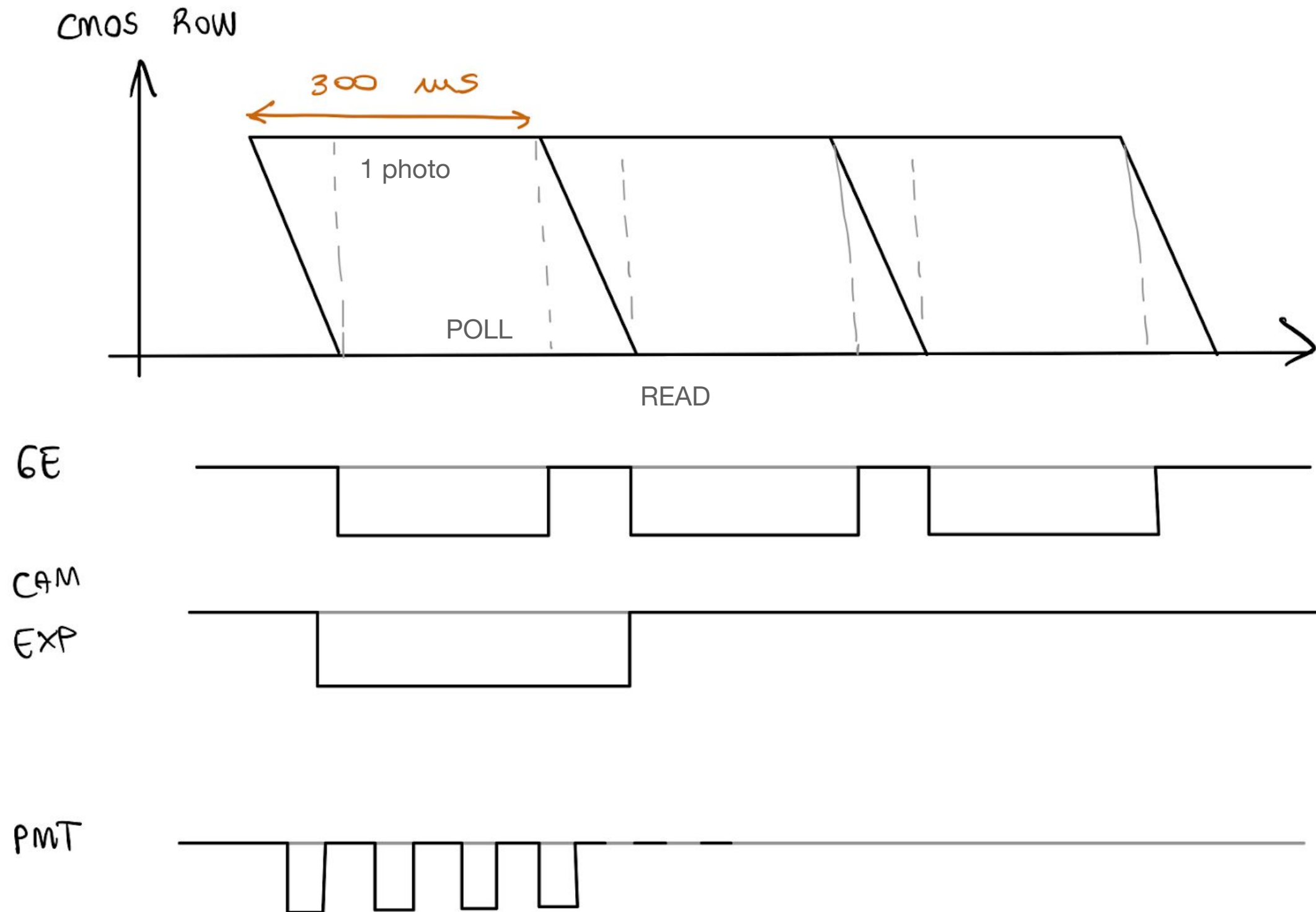
in ConfigCamera() :



```
//continous stream mode
if(mode==3) {
    err = dcamprop_setvalue( gCam, DCAM_IDPROP_TRIGGER_MODE, DCAMPROP_TRIGGER_MODE__START );
    if(failed(err)) cout << "ERROR IN DCAM_IDPROP_TRIGGER_MODE" << endl;
} else {
    err = dcamprop_setvalue( gCam, DCAM_IDPROP_TRIGGER_MODE, DCAMPROP_TRIGGER_MODE__NORMAL );
    if(failed(err)) cout << "ERROR IN DCAM_IDPROP_TRIGGER_MODE" << endl;
}
```

(all code showed is in `cygnus_fe.cxx`)

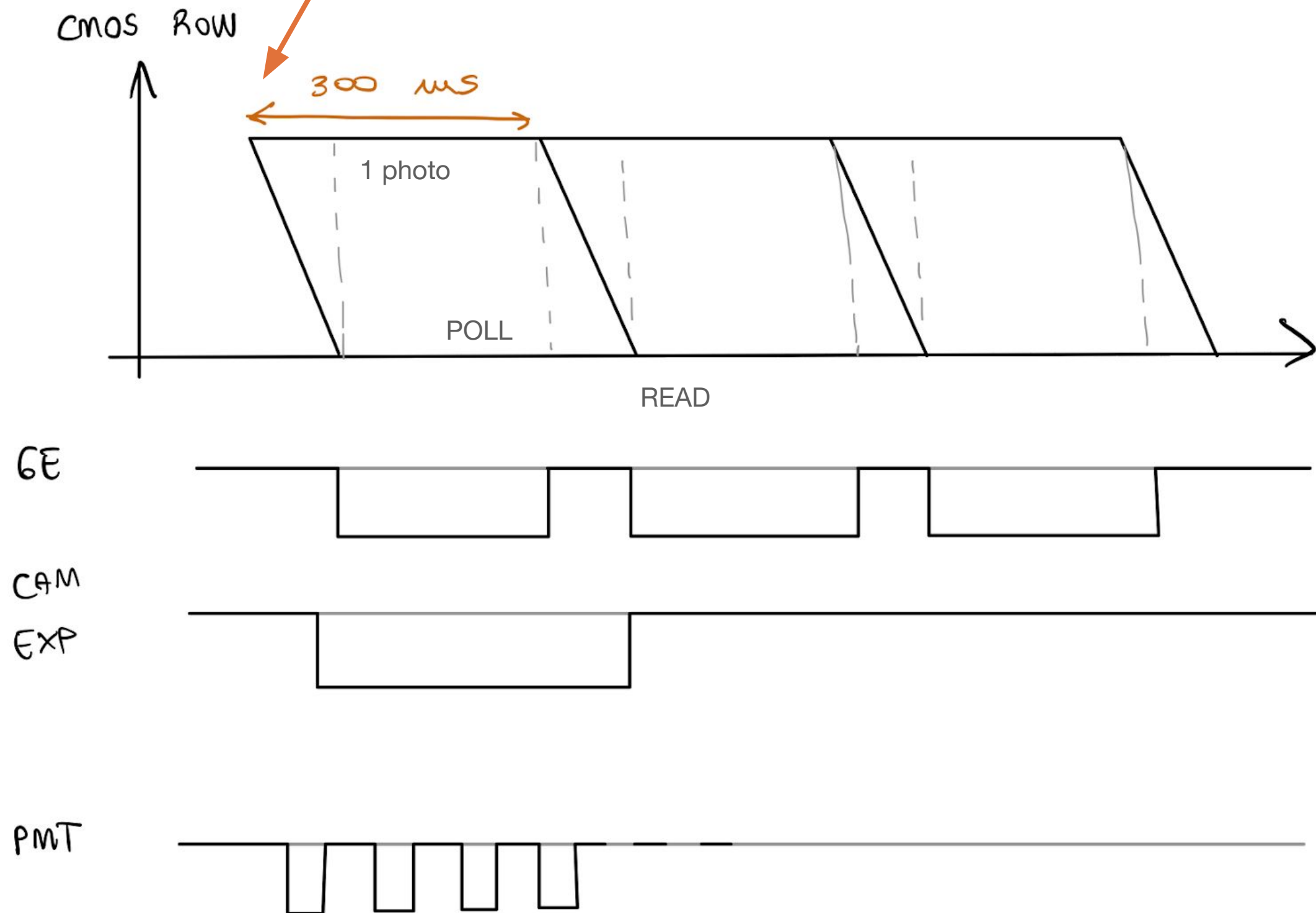
After



After

```
if(mode==3) {  
    dcamcap_firetrigger(gCam,0);  
}
```

in begin_of_run()



Data Bank info

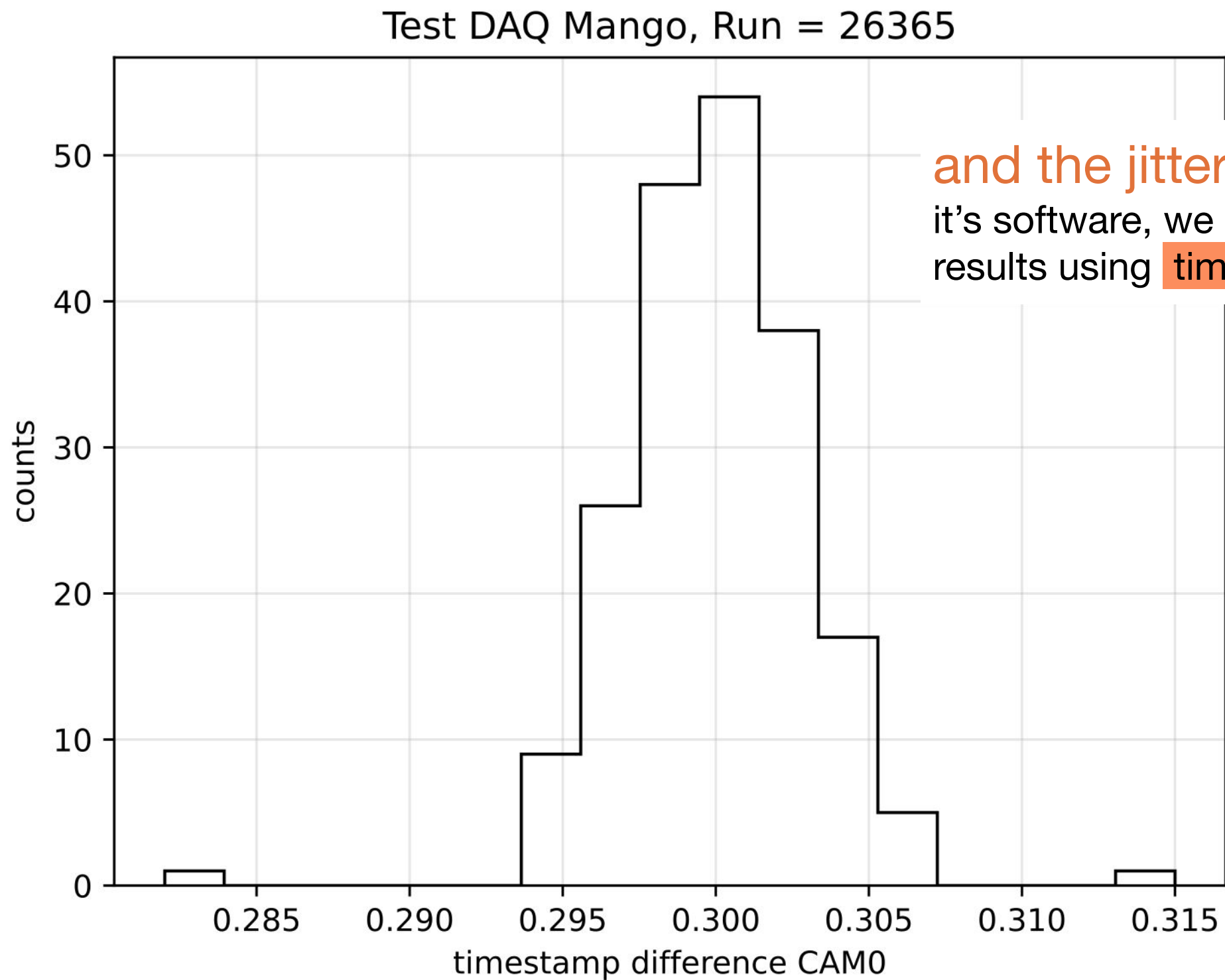
- TIME: only the first event, contains UTC time (bank.data[0])
- CAM0: time interval from TIME in ms (event.header.timestamp/1e3)
- DGH0, DIG0: Digitizer Bank, contains also the TTT
(fullhead[39+i]*8.5*1e-6)

N.B.

Before: DGH0, DIG0 bank in every event (Digitizer Ruled).

After: DGH0, DIG0 bank in **not** every event (CAM Rules).

Test on this: timestamp diff peaked at 300 ms

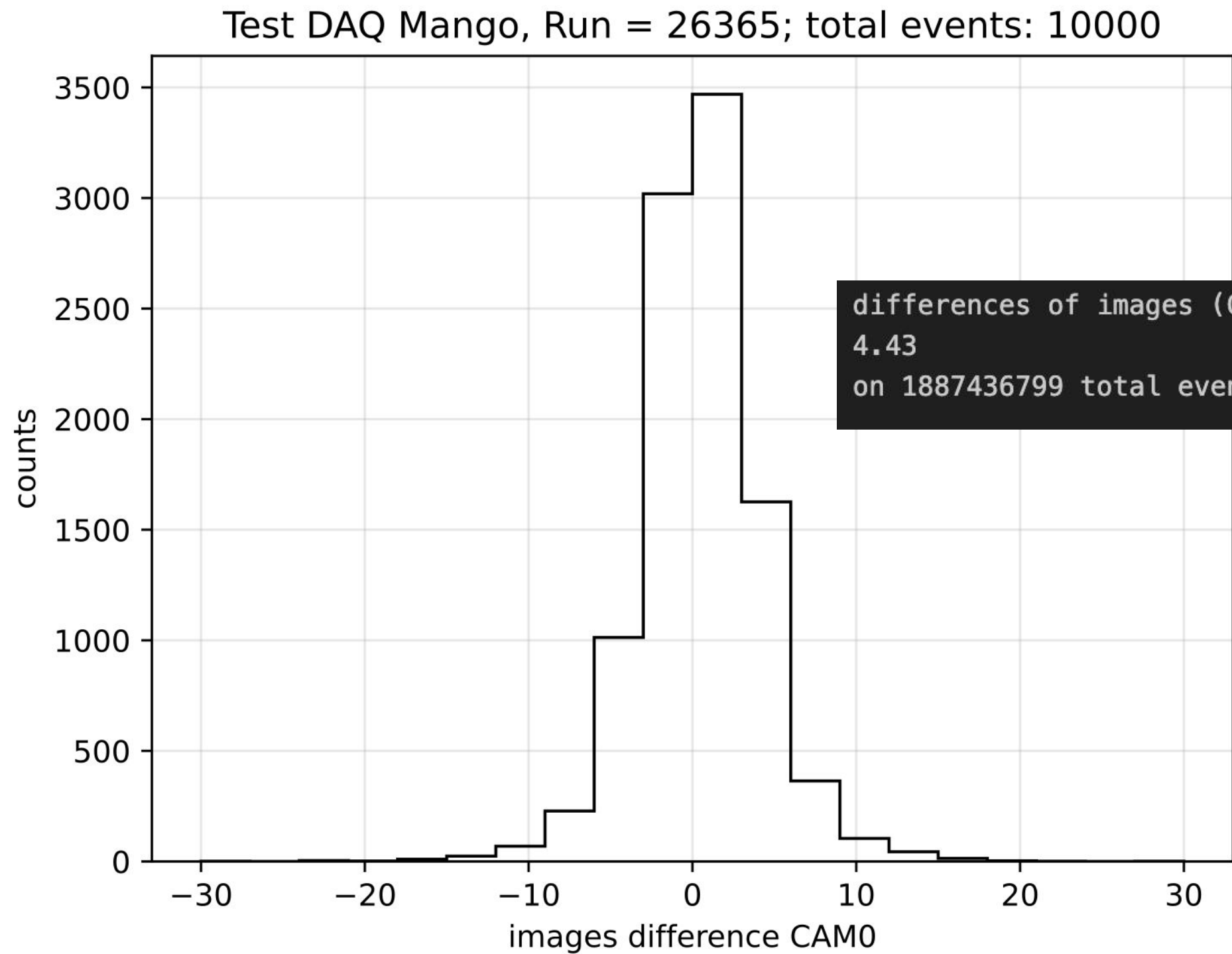


and the jitter?

it's software, we could achieve better results using timestamp from cam



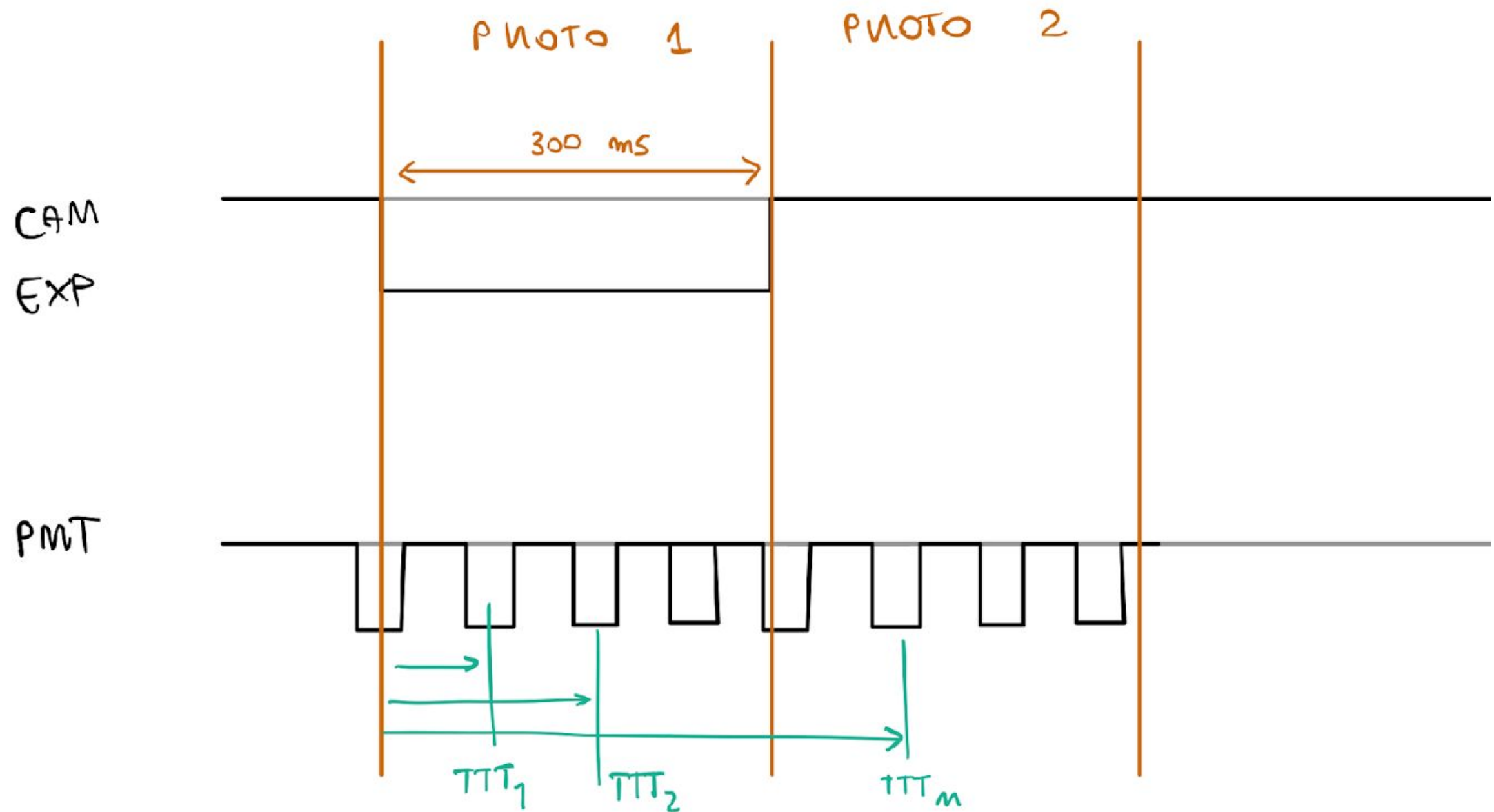
Test on this: images differences have std dev $\neq 0$



differences of images (CAM0) have the std dev:
4.43
on 1887436799 total events



What about the Trigger Time Tag?



Test on this: DGH0 banks in event given builded signal

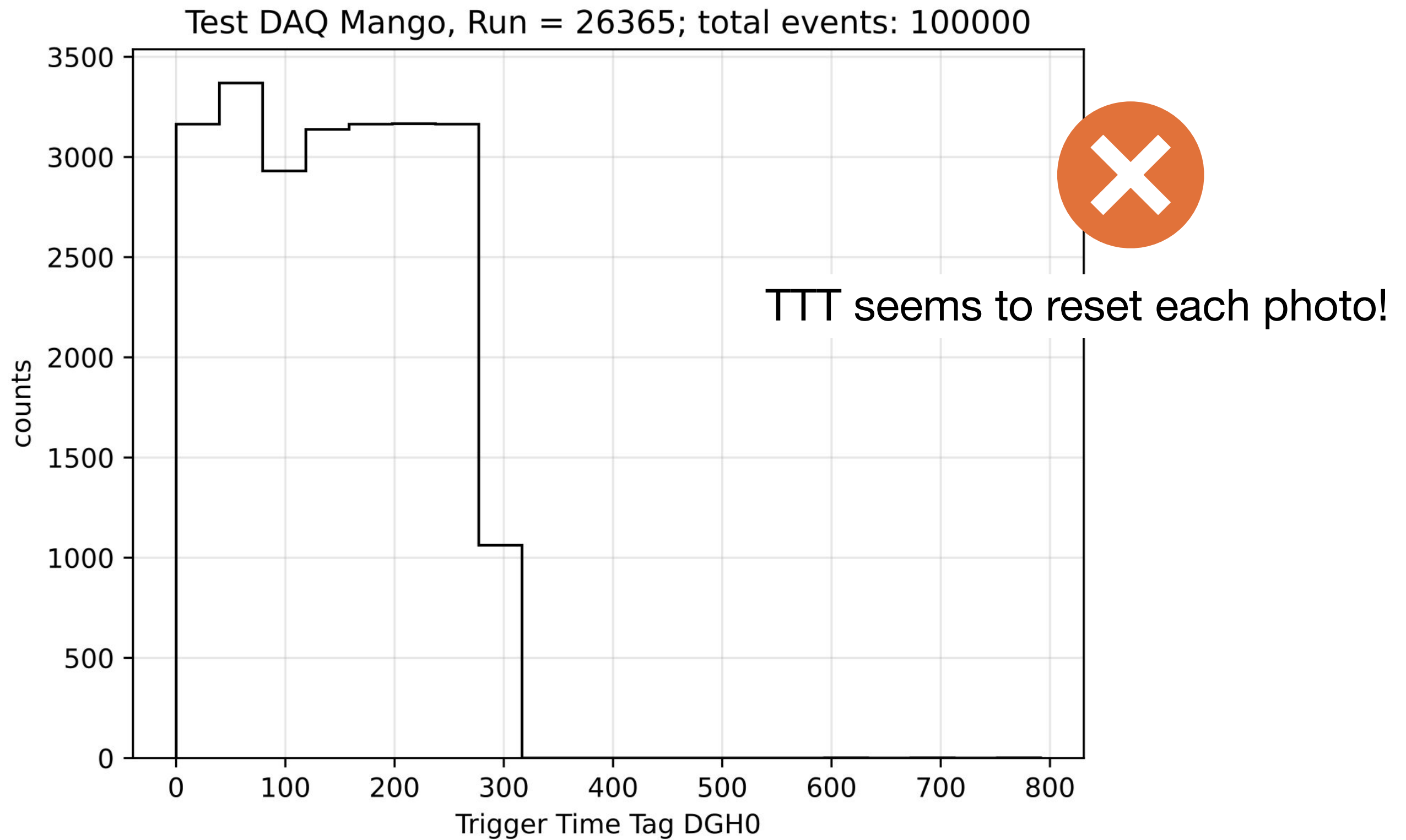
```
for the run 26365, frequency by oscilloscope: 854 ms  
rate of finding a dgtz event in a photo:  
expected: 0.3513  
measured: 0.3505
```



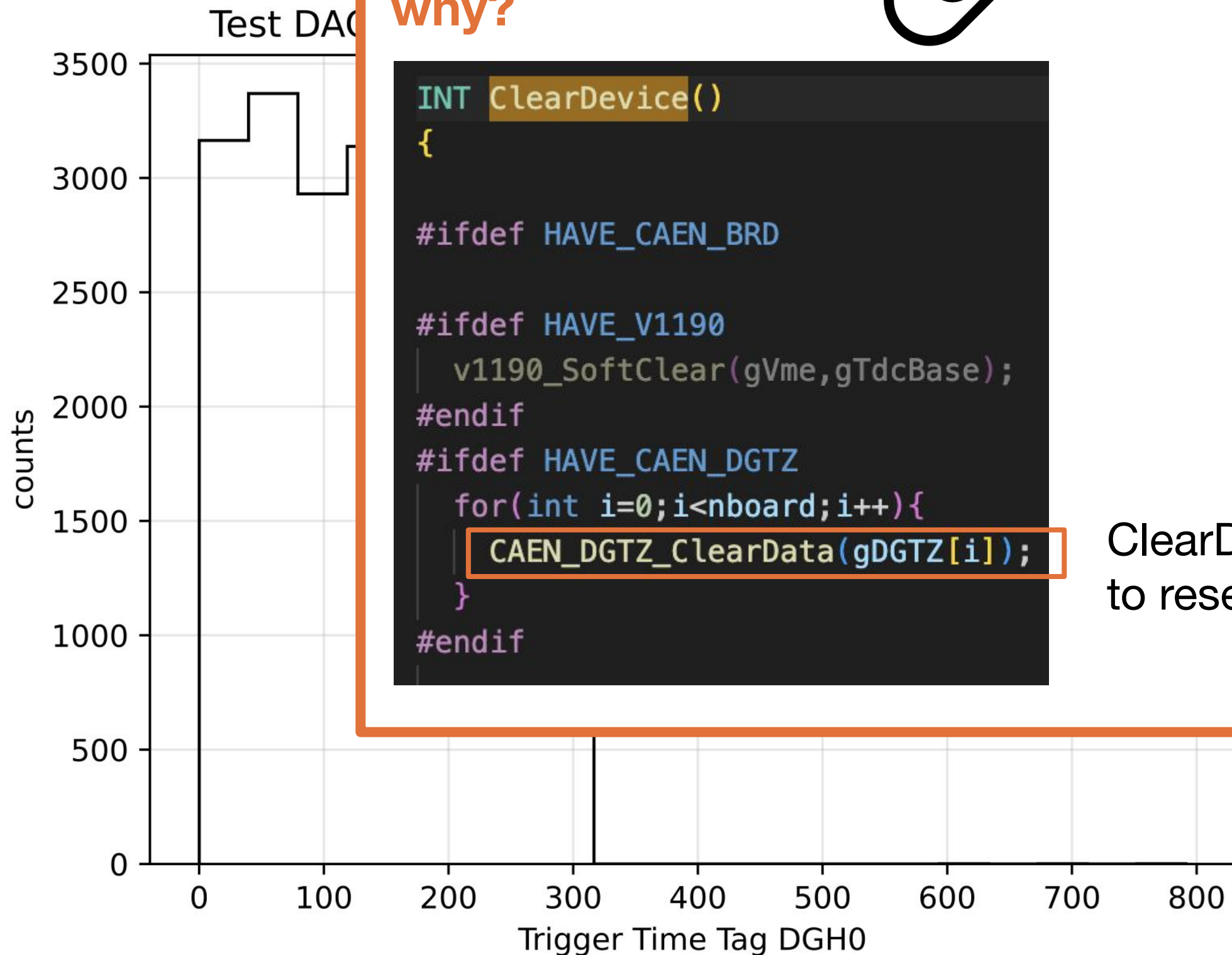
```
for the run 26366, frequency by oscilloscope: 94 ms  
rate of finding a dgtz event in a photo:  
expected: 3.1915  
measured: 3.1012
```



Test on this: TTT distribution

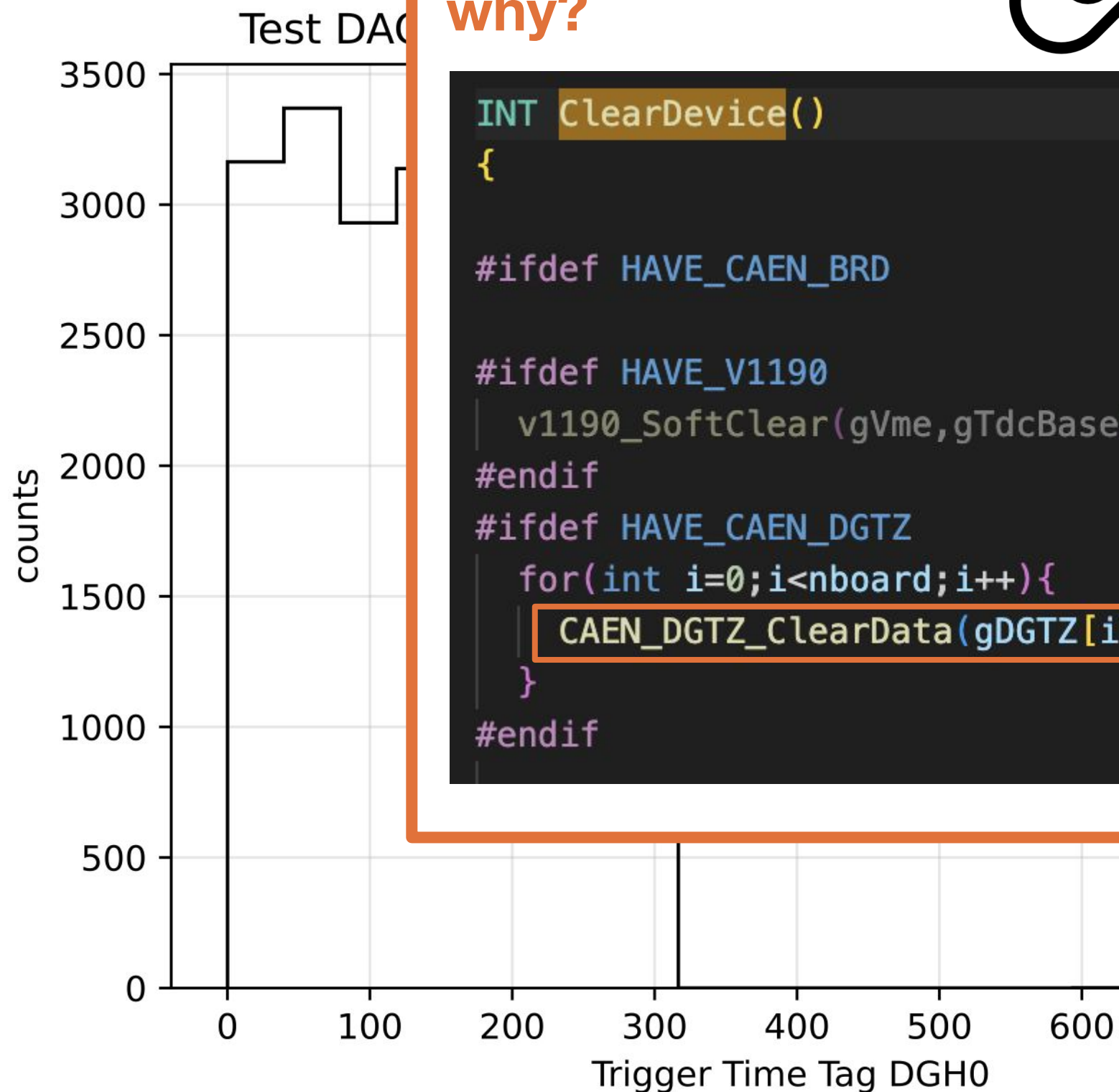


Test on this: TTT distribution



ClearData seems
to reset it

Test on this: TTT distribution



why?

```
INT ClearDevice()  
{  
  
#ifdef HAVE_CAEN_BRD  
  
#ifdef HAVE_V1190  
    v1190_SoftClear(gVme,gTdcBase);  
#endif  
#ifdef HAVE_CAEN_DGTZ  
    for(int i=0;i<nboard;i++){  
        CAEN_DGTZ_ClearData(gDGTZ[i]);  
    }  
#endif  
}
```

*up to now, we
open a ticket
to CAEN*

**thanks for your
attention :)**