




Kaon LT Status Update

November 4th, 2020

Richard Trotta


$$P_{\text{HMS}} = -3.266$$
$$\theta_{\text{HMS}} = 12.53$$

- 55 uA
 - 5149, 5150, 5152, 5154
- 45, 30, 15, 5 uA
 - 5155, 5156, 5157, 5158
- Tracking efficiency
 - (50 uA) 98,97,97,97
 - 98,99,99,99

$$P_{\text{SHMS}} = 6.842$$
$$\theta_{\text{SHMS}} = 6.495$$

- 55 uA
 - 5149, 5150, 5153, 5154
- 45, 30, 15, 5 uA
 - 5155, 5156, 5157, 5158
- Tracking efficiency
 - (55 uA) 95,90,93,96
 - 97,98,98,99


Cuts



- Current $< |2.5| \mu\text{A}$
- Delta
 - HMS, $(-8 < \text{delta} < 8)$
 - SHMS, $(-10 < \text{delta} < 20)$
- xpfp
 - HMS, $(-0.08 < \text{xpfp} < 0.08)$
 - SHMS, $(-0.06 < \text{xpfp} < 0.06)$
- ypfp
 - HMS, $(-0.045 < \text{ypfp} < 0.045)$
 - SHMS, $(-0.04 < \text{ypfp} < 0.04)$
- start time
 - `H(P).hod.goodstarttime == 1`

Equations

$$Q_{tot} = (H.BCM.scaler.charge)$$

- Scaler 
$$Y_{scaler} = \frac{N_{scaler}}{Q_{tot}} \quad N_{scaler} = \Sigma(trigscaler)$$

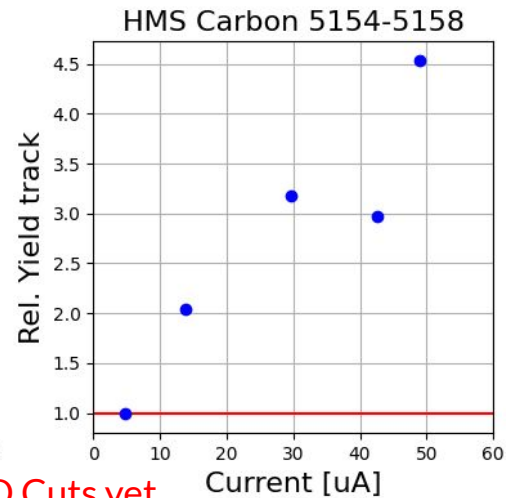
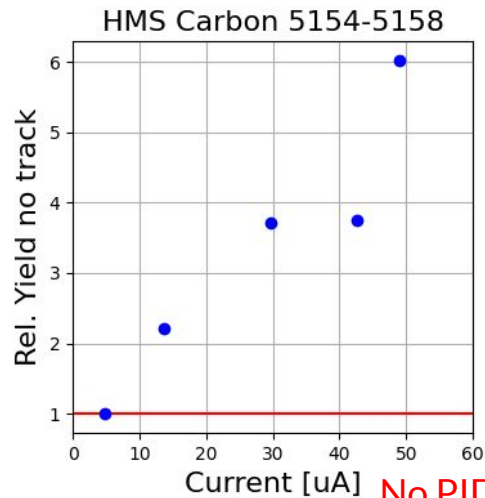
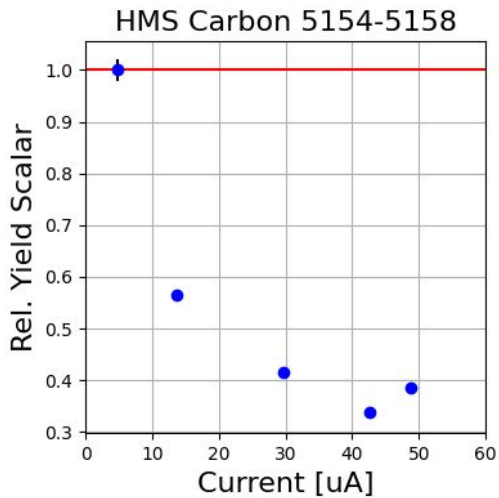
- No track
$$Y_{notrack} = \frac{PS * N_{electrons}}{Q_{tot} \epsilon_{cpuLT}} \quad N_{electrons} = \int (H/P.hod.goodscinhits)$$

Need to check if track cuts in replay script

$$cpuLT = \frac{EvtType}{TRIG_{accept} - edtm}$$

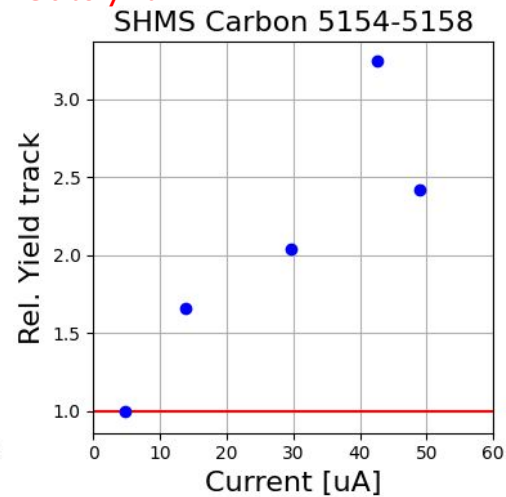
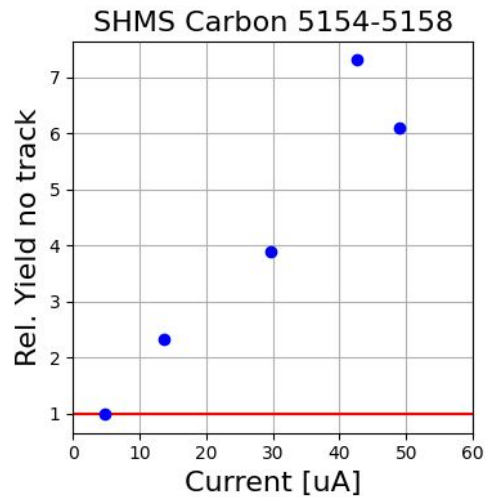
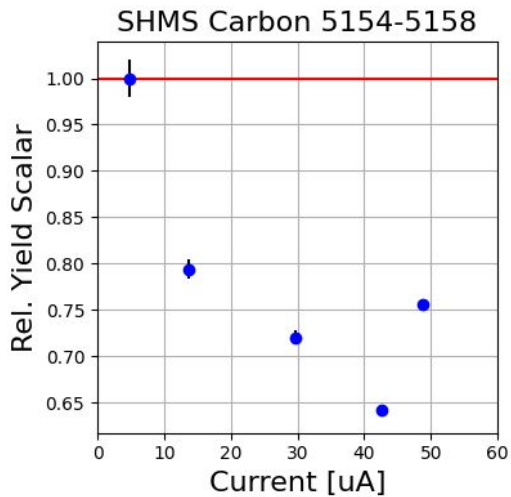
- Track
$$Y_{track} = \frac{PS * N_{track}}{Q_{tot} \epsilon_{track} \epsilon_{cpuLT}} \quad N_{track} = tracks$$

$$P_{HMS} = -3.266$$



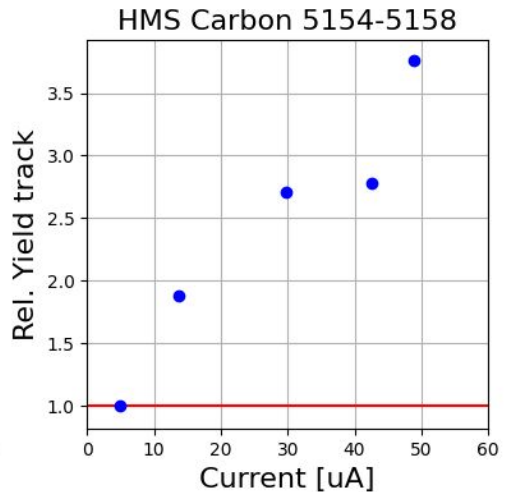
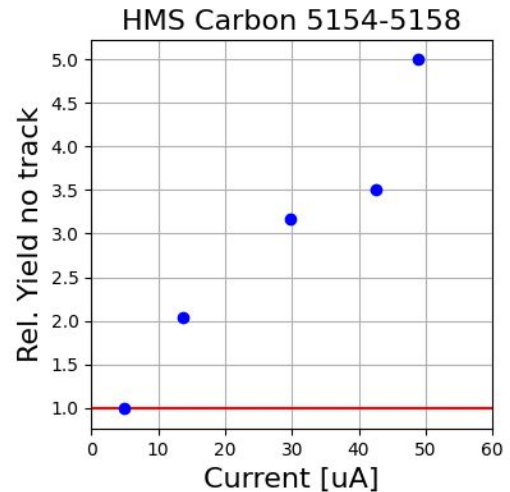
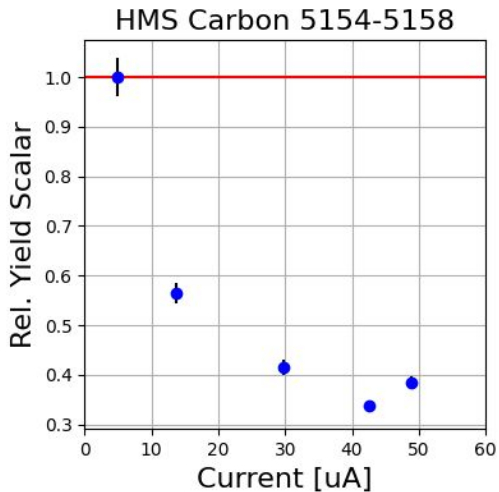
No PID Cuts yet

$$P_{SHMS} = 6.842$$



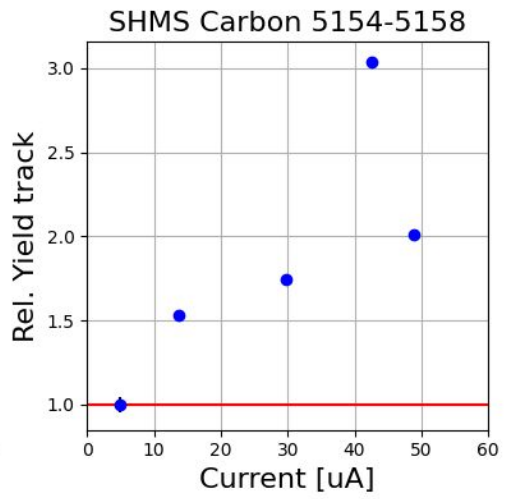
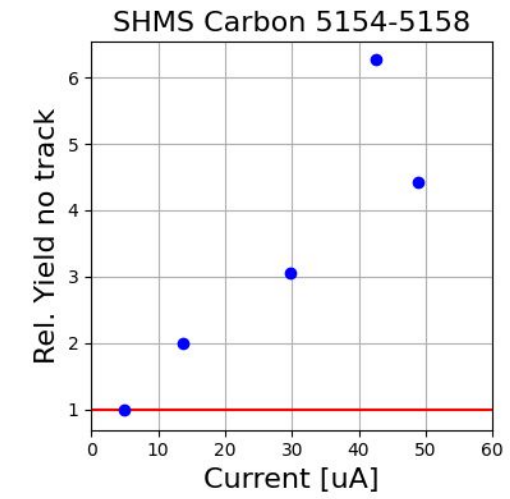
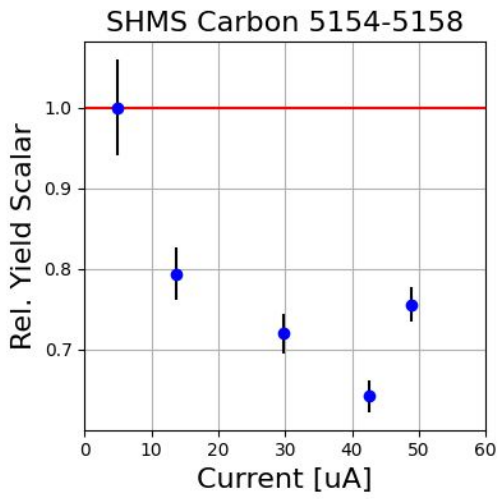
$$P_{\text{HMS}} = -3.266$$

Electron
Cer > 1.5
Cal > 0.7

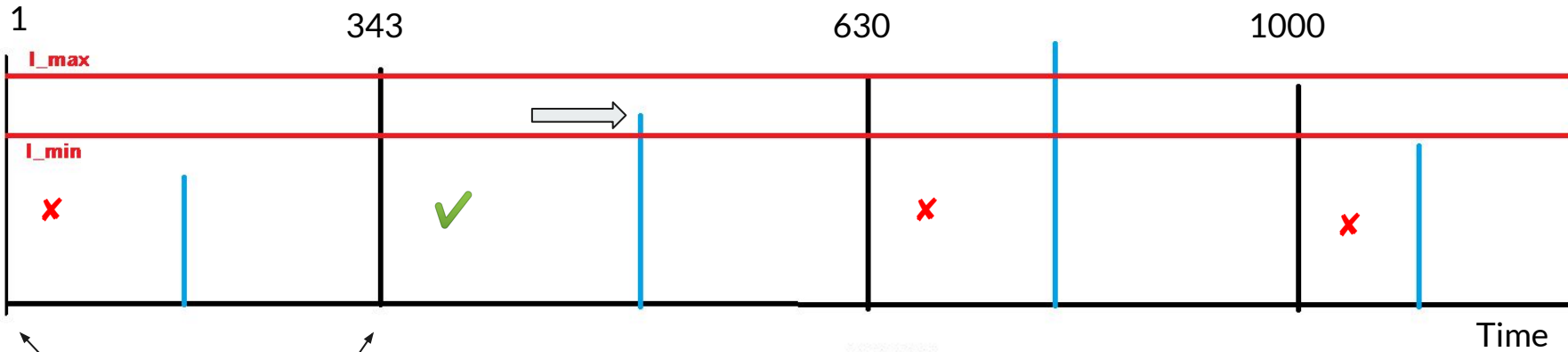


$$P_{\text{SHMS}} = 6.842$$

Hadron
Hgcer > 1.5
Aero > 1.5
Cal > 0.7



Events associated with scaler read



Scaler read events
(every 2s or 1000 events)

RUN: 5154
total_trig1_scaler_bcm_cut = 12850.7(Carlos), 201 [kHz]
total_trig3_scaler_bcm_cut = 2678.54 (Carlos), 48 [kHz]
total_charge_bcm_cut = 2548.83 (Carlos), 38.37 [mC]

RUN: 5158
total_trig1_scaler_bcm_cut = 842.502 (Carlos), 23 [kHz]
total_trig3_scaler_bcm_cut = 199.449 (Carlos), 11 [kHz]
total_charge_bcm_cut = 186.346 (Carlos), 3.7 [mC]

```

for ibcm in range(0, 5):
    current_I = 0
    for i, evt in enumerate(s_evts):
        if (time_value[i] != previous_time[ibcm]):
            current_I = (bcm_value[ibcm][i] -
                        previous_charge[ibcm])/(time_value[i] - previous_time[ibcm])
        if (( current_I-thres_curr < current_I < current_I+thres_curr )):
            charge_sum[ibcm] += (bcm_value[ibcm][i] - previous_charge[ibcm])
            time_sum[ibcm] += (time_value[i] - previous_time[ibcm])
        if (ibcm == 2 and ( current_I-thres_curr < current_I < current_I+thres_curr )):
            print("\n\ncurrent_I",current_I-thres_curr, "<", current_I, "<", current_I+thres_curr, "\n\n")
            EDMT_current = (EDTM_value[i] - previous_EDTM)
            EDMT_sum += EDMT_current
            acctrig_sum += ((acctrig_value[i] - EDMT_current) - previous_acctrig)
            for itrig in range(0, NTRIG):
                trig_sum[itrig] += (trig_value[itrig][i] - previous_trig[itrig])
            for iPRE in range(0, NPRE):
                PRE_sum[iPRE] += (PRE_value[iPRE][i] - previous_PRE[iPRE])
                SHMS_PRE_sum[iPRE] += (SHMS_PRE_value[iPRE][i] - SHMS_previous_PRE[iPRE])
            for iRATE in range(0, NRATE):
                rate_sum[iRATE] += (rate_value[iRATE][i] - previous_rate[iRATE])
            for iRATE in range(0, SHMSNRATE):
                SHMS_rate_sum[iRATE] += (SHMS_rate_value[iRATE][i] - SHMS_previous_rate[iRATE])
            previous_acctrig = (acctrig_value[i] - EDMT_current)
            previous_EDTM = EDMT_value[i]
            for itrig in range(0, NTRIG):
                previous_trig[itrig] = trig_value[itrig][i]
            for iPRE in range(0, NPRE):
                previous_PRE[iPRE] = PRE_value[iPRE][i]
                SHMS_previous_PRE[iPRE] = SHMS_PRE_value[iPRE][i]
            for iRATE in range(0, NRATE):
                previous_rate[iRATE] = rate_value[iRATE][i]
            for iRATE in range(0, SHMSNRATE):
                SHMS_previous_rate[iRATE] = SHMS_rate_value[iRATE][i]
            time_total += (time_value[i] - previous_time[ibcm])
            previous_time[ibcm] = time_value[i]
            previous_charge[ibcm] = bcm_value[ibcm][i]

```

```

//Scaler reads loop.
for (int i = 0; i < scal_entries; i++)
{
    /*(NOTE: Each scaler read is associated with as specific event number
    as (scaler read 1-> event 1000, scaler read 2 -> event 2300, ...)
    This means events up to 1000 correspond to scaler read 1, ...*/

    scaler_tree->GetEntry(i);

    //Set Event Flag to FALSE
    evt_flag_bcm[i] = 0;

    //Store event associated with scaler read
    scal_evt_num[i] = Scal_evtNum;

    //Store Cumulative Quantities
    total_charge_bcm = Scal_BCM_charge;
    total_time = Scal_time;
    total_ps1x_scaler = ps1X_scaler;
    total_ptrig1_scaler = pTRIG1_scaler;
    total_ptrig2_scaler = pTRIG2_scaler;
    total_ptrig3_scaler = pTRIG3_scaler;
    total_ptrig4_scaler = pTRIG4_scaler;
    total_ptrig6_scaler = pTRIG6_scaler;
    total_pedtm_scaler = pEDTM_scaler;

    //Check If BCM Beam Current in Between Reads is Over Threshold
    // if(abs(Scal_BCM_current-set_current)<current_thrs_bcm)
    if(Scal_BCM_current>current_thrs_bcm)

    {

        //Turn Event Flag ON, if beam current is within threshold
        evt_flag_bcm[i] = 1;

        //Store Quantities that Passed the Current Threshold
        total_time_bcm_cut = total_time_bcm_cut + (Scal_time - prev_time);
        total_charge_bcm_cut = total_charge_bcm_cut + (Scal_BCM_charge - prev_charge_bcm);
        total_ps1x_scaler_bcm_cut = total_ps1x_scaler_bcm_cut + (ps1X_scaler-prev_ps1x_scaler);
        total_ptrig1_scaler_bcm_cut = total_ptrig1_scaler_bcm_cut + (pTRIG1_scaler-prev_ptrig1_scaler);
        total_ptrig2_scaler_bcm_cut = total_ptrig2_scaler_bcm_cut + (pTRIG2_scaler-prev_ptrig2_scaler);
        total_ptrig3_scaler_bcm_cut = total_ptrig3_scaler_bcm_cut + (pTRIG3_scaler-prev_ptrig3_scaler);
        total_ptrig4_scaler_bcm_cut = total_ptrig4_scaler_bcm_cut + (pTRIG4_scaler-prev_ptrig4_scaler);
        total_ptrig6_scaler_bcm_cut = total_ptrig6_scaler_bcm_cut + (pTRIG6_scaler-prev_ptrig6_scaler);
        total_pedtm_scaler_bcm_cut = total_pedtm_scaler_bcm_cut + (pEDTM_scaler - prev_pedtm_scaler);

    } //End BCM Current Cut
}

```




Extra