

roi1_detector

June 11, 2024

```
[ ]: import databroker
      from bluesky.run_engine import RunEngine
      from bluesky.callbacks.best_effort import BestEffortCallback

      cat = databroker.temp().v2
      RE = RunEngine()
      RE.subscribe(cat.v1.insert)
      RE.subscribe(BestEffortCallback())
```

```
[ ]: 1
```

```
[ ]: from apstools.devices import SingleTrigger_V34
      from ophyd import ADComponent
      from ophyd.areadetector import DetectorBase
      from ophyd.areadetector import SimDetector
      from ophyd.areadetector.plugins import ImagePlugin_V34 as ImagePlugin
      from ophyd.areadetector.plugins import PvaPlugin_V34 as PvaPlugin
      from ophyd.areadetector.plugins import ROIPlugin_V34 as ROIPlugin
      from ophyd.areadetector.plugins import StatsPlugin_V34 as StatsPlugin

      class SimDetector_V34(SingleTrigger_V34, SimDetector):
          """
          ADSimDetector

          SingleTrigger:

          * stop any current acquisition
          * sets image_mode to 'Multiple'
          """

          # hdf1 = ADComponent(
          #     MyHDF5Plugin,
          #     "HDF1:",
          #     write_path_template=WRITE_PATH_TEMPLATE,
          #     read_path_template=READ_PATH_TEMPLATE,
          # )
          image = ADComponent(ImagePlugin, "image1:")
```

```

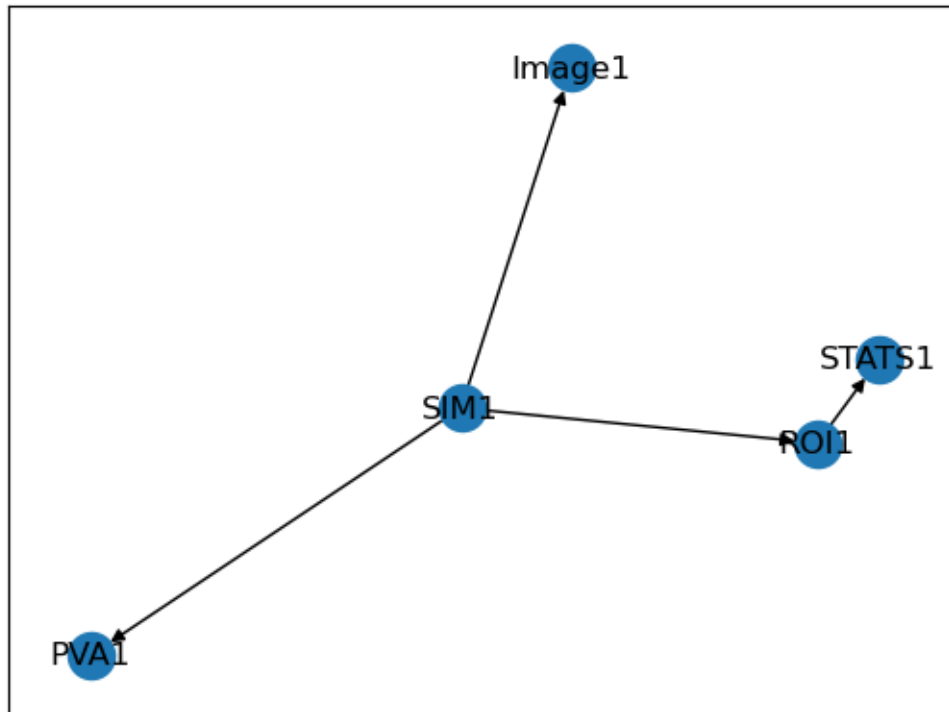
pva = ADComponent(PvaPlugin, "Pva1:")
roi1 = ADComponent(ROIPlugin, "ROI1:")
stats1 = ADComponent(StatsPlugin, "Stats1:")

det = SimDetector_V34("kad:", name="det")
det.missing_plugins()
det.visualize_asyn_digraph()
det.stats1.kind=3
det.stats1.max_value.kind = "hinted"
det.stats1.total.kind = "hinted"
print(f"{det.stats1.kind=}")

```

```
det.stats1.kind=<Kind.normal|config: 3>
```

AD port map for det



```
[ ]: det.read()
```

```
[ ]: OrderedDict([('det_stats1_max_value',
                  {'value': 160.0, 'timestamp': 1706733270.8807716}),
                 ('det_stats1_total',
                  {'value': 11429145.0, 'timestamp': 1706733270.8819215})])
```

```
[ ]: from bluesky import plans as bp
      RE(bp.count([det.stats1.total, det.stats1.max_value], num=5))
```

Transient Scan ID: 2 Time: 2024-01-31 14:34:37

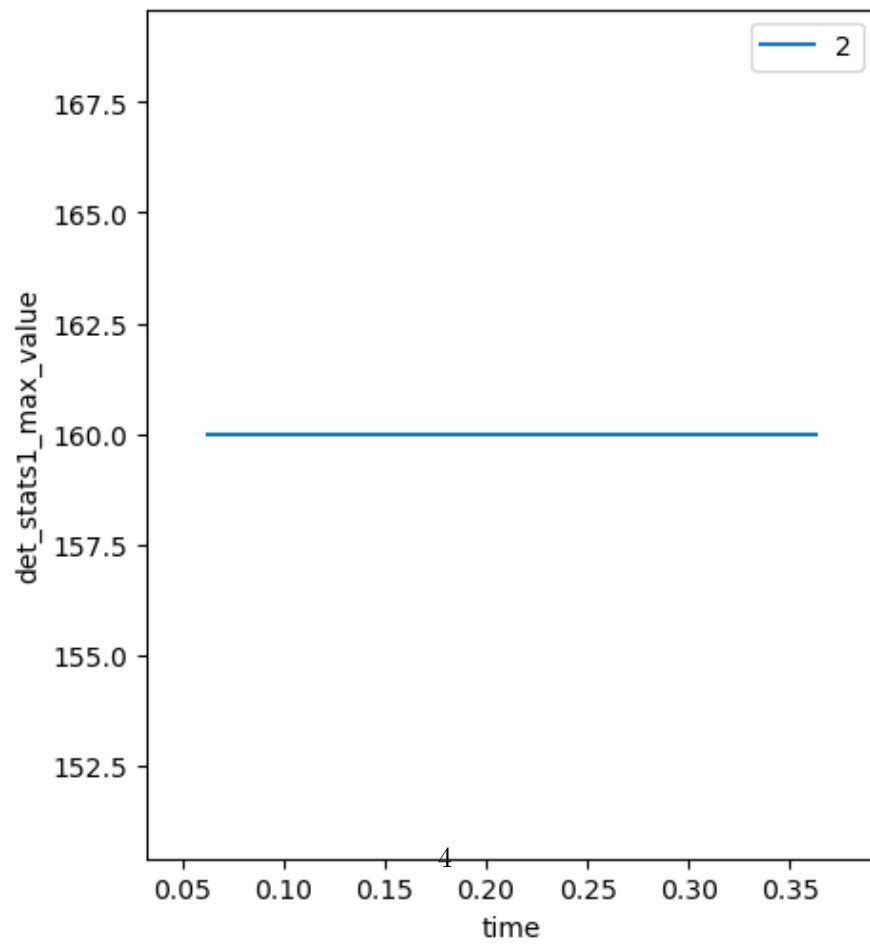
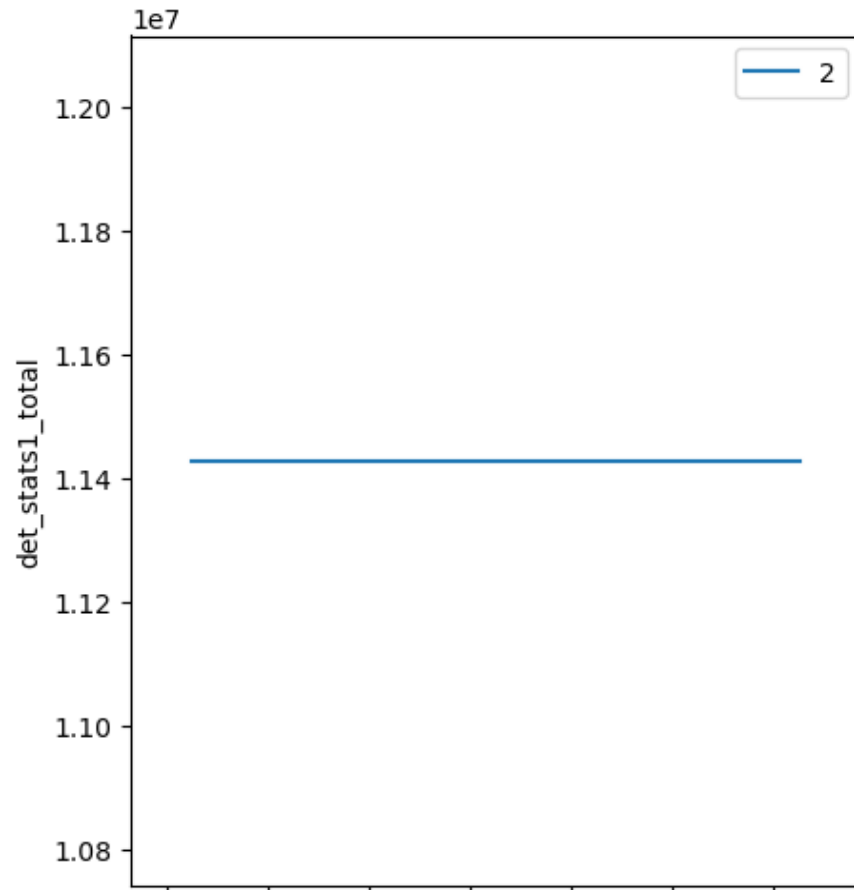
Persistent Unique Scan ID: 'b90b5bf4-bcf0-4f89-b92d-734cab21dbf2'

New stream: 'primary'

seq_num	time	det_stats1_total	det_stats1_max_value
1	14:34:37.0	11429145	160
2	14:34:37.1	11429145	160
3	14:34:37.2	11429145	160
4	14:34:37.3	11429145	160
5	14:34:37.3	11429145	160

generator count ['b90b5bf4'] (scan num: 2)

```
[ ]: ('b90b5bf4-bcf0-4f89-b92d-734cab21dbf2',)
```



```
[ ]: ds = cat[-1].primary.read()
ds
```

```
[ ]: <xarray.Dataset>
Dimensions:                (time: 3)
Coordinates:
  * time                    (time) float64 1.707e+09 1.707e+09 1.707e+09
Data variables:
  det_stats1_max_value     (time) float64 159.0 159.0 164.0
  det_stats1_total         (time) float64 1.136e+07 1.135e+07 1.172e+07
  m1                       (time) float64 -1.0 0.0 1.0
  m1_user_setpoint         (time) float64 -1.0 0.0 1.0
```

```
[ ]: from ophyd import EpicsMotor
m1 = EpicsMotor("kgp:m1", name="m1")
```

```
[ ]: m1.connected
```

```
[ ]: True
```

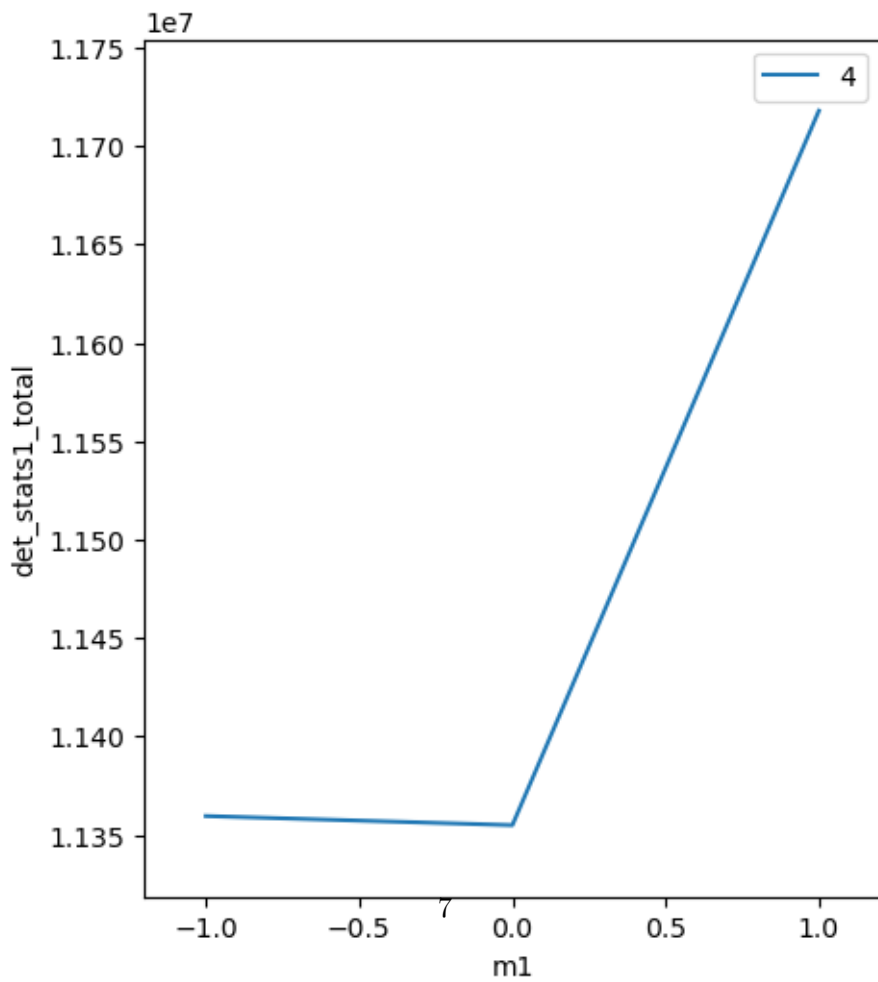
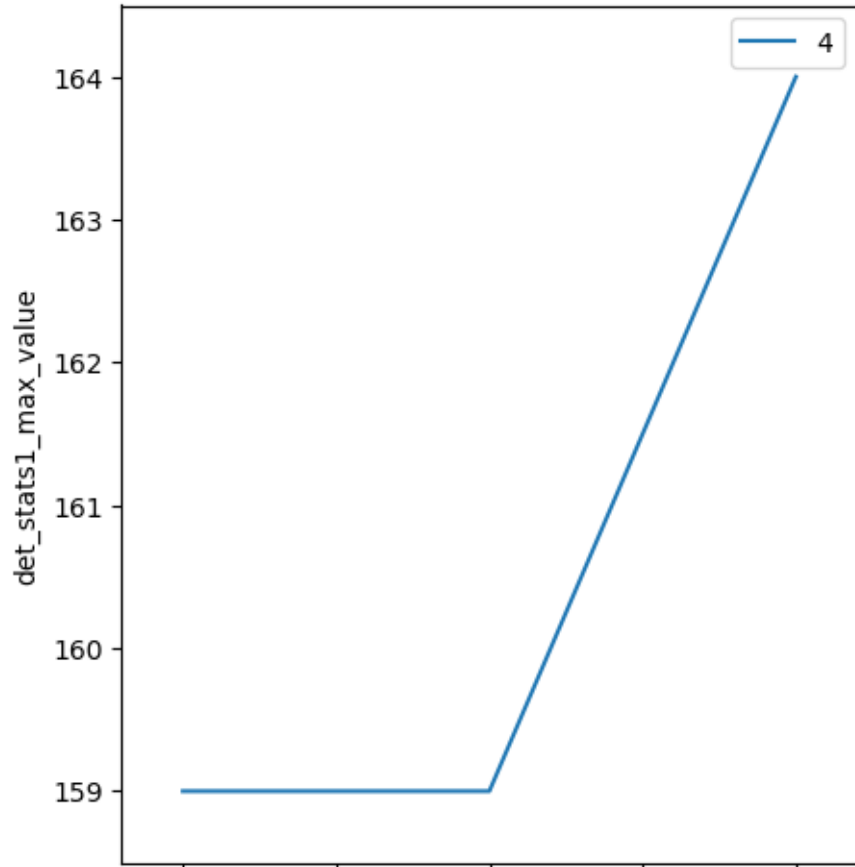
```
[ ]: RE(bp.scan([det], m1, -1, 1, 3))
```

```
Transient Scan ID: 4      Time: 2024-01-31 14:36:53
Persistent Unique Scan ID: 'cd57a32b-23e8-4856-aa89-032cf928c3c7'
New stream: 'primary'
```

```
+-----+-----+-----+-----+-----+
+
|  seq_num |      time |      m1 | det_stats1_max_value | det_stats1_total |
|          |          |          |                      |                  |
+-----+-----+-----+-----+-----+
+
|         1 | 14:36:55.8 | -1.0000 |          159 |          11359501 |
|          |          |          |                      |                  |
|         2 | 14:36:57.1 |  0.0000 |          159 |          11354861 |
|          |          |          |                      |                  |
|         3 | 14:36:58.4 |  1.0000 |          164 |          11717979 |
|          |          |          |                      |                  |
+-----+-----+-----+-----+-----+
+
```

```
generator scan ['cd57a32b'] (scan num: 4)
```

[]: ('cd57a32b-23e8-4856-aa89-032cf928c3c7',)



```
[ ]: from ophyd import EpicsSignalRO
stats_id = EpicsSignalRO("kad:Stats1:UniqueId_RBV", name="stats_id")

[ ]: stats_id.kind

[ ]: <Kind.hinted: 5>

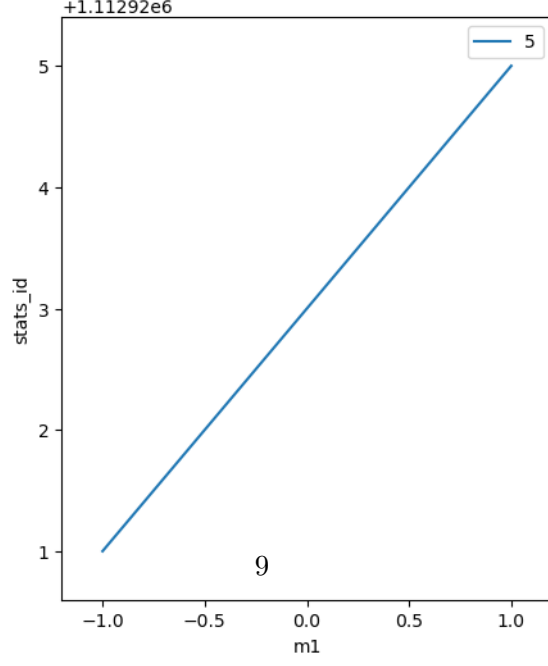
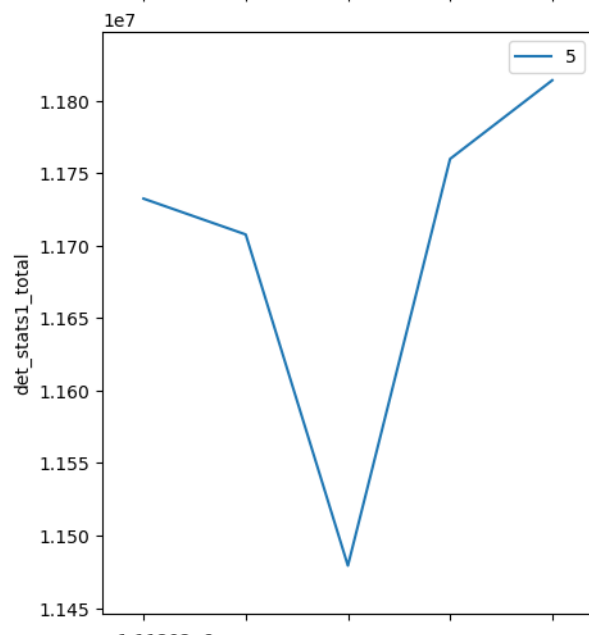
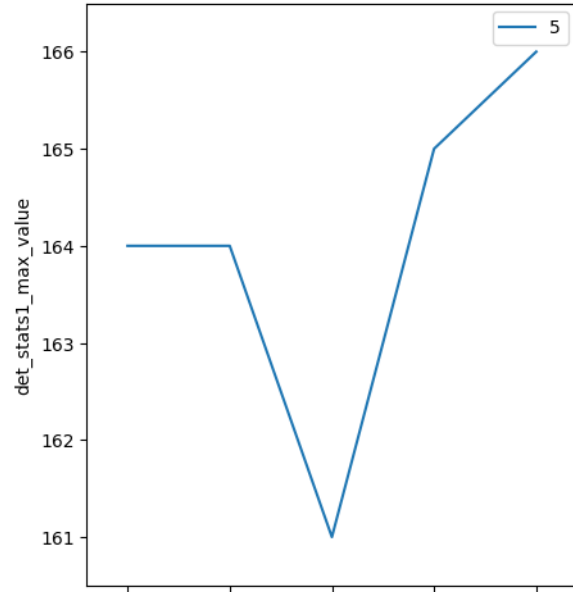
[ ]: RE(bp.scan([det, stats_id], m1, -1, 1, 5))
```

Transient Scan ID: 5 Time: 2024-01-31 14:43:59
 Persistent Unique Scan ID: '791d41d1-df31-4e1b-8401-4b6e92a58bbc'
 New stream: 'primary'

```
+-----+-----+-----+-----+-----+
+-----+
|  seq_num |      time |      m1 | det_stats1_max_value | det_stats1_total |
|  stats_id |
+-----+-----+-----+-----+-----+
+-----+
|      1 | 14:44:01.5 | -1.0000 |          164 |          11732442 |
| 1112921 |
|      2 | 14:44:02.4 | -0.5000 |          164 |          11707715 |
| 1112922 |
|      3 | 14:44:03.3 |  0.0000 |          161 |          11479265 |
| 1112923 |
|      4 | 14:44:04.2 |  0.5000 |          165 |          11759850 |
| 1112924 |
|      5 | 14:44:05.1 |  1.0000 |          166 |          11814100 |
| 1112925 |
+-----+-----+-----+-----+-----+
+-----+
```

generator scan ['791d41d1'] (scan num: 5)

```
[ ]: ('791d41d1-df31-4e1b-8401-4b6e92a58bbc',)
```

```
[ ]: det_list = []
det_list.append(det)
det_list.append(stats_id)
RE(bp.scan(det_list, m1, -1, 1, 5))
```

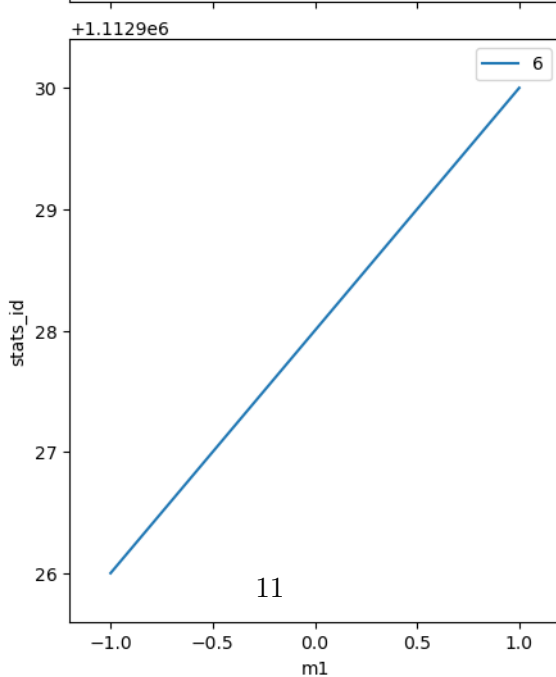
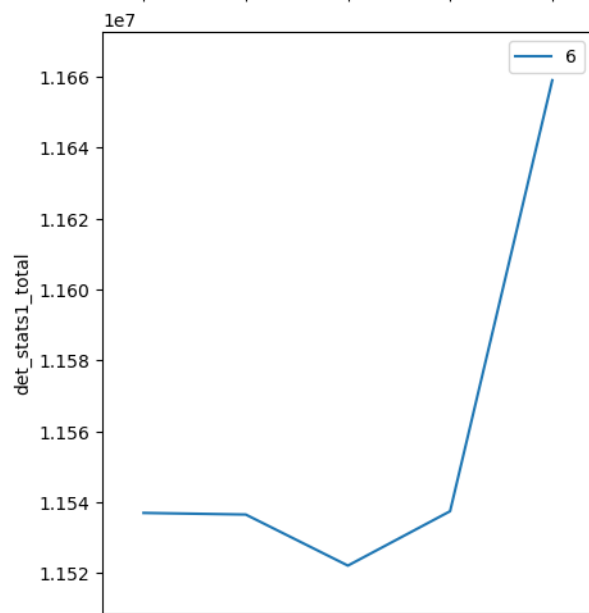
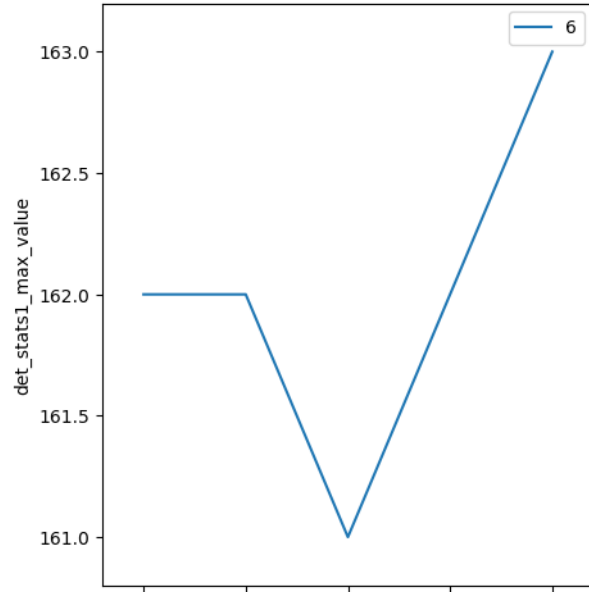
Transient Scan ID: 6 Time: 2024-01-31 14:47:59
Persistent Unique Scan ID: '132e4c4f-1c71-48b8-b3a9-f430d79480e5'
New stream: 'primary'

```

+-----+-----+-----+-----+-----+
+-----+
|  seq_num |      time |      m1 | det_stats1_max_value | det_stats1_total
|  stats_id |
+-----+-----+-----+-----+-----+
+-----+
|      1 | 14:48:02.3 | -1.0000 |          162 |          11536929
| 1112926 |
|      2 | 14:48:03.2 | -0.5000 |          162 |          11536476
| 1112927 |
|      3 | 14:48:04.1 |  0.0000 |          161 |          11522059
| 1112928 |
|      4 | 14:48:05.1 |  0.5000 |          162 |          11537405
| 1112929 |
|      5 | 14:48:06.1 |  1.0000 |          163 |          11659029
| 1112930 |
+-----+-----+-----+-----+-----+
+-----+
generator scan ['132e4c4f'] (scan num: 6)

```

```
[ ]: ('132e4c4f-1c71-48b8-b3a9-f430d79480e5',)
```



```
[ ]: det.read()
```

```
[ ]: OrderedDict([('det_stats1_max_value',  
                 {'value': 163.0, 'timestamp': 1706734086.168164}),  
                 ('det_stats1_total',  
                 {'value': 11659029.0, 'timestamp': 1706734086.168177})])
```

```
[ ]: det.stage()  
     det.trigger()  
     det.unstage()
```

```
cb_max_value: args=() kwargs={'old_value': 161.0, 'value': 165.0, 'timestamp':  
1706735231.362738, 'status': <AlarmStatus.NO_ALARM: 0>, 'severity':  
<AlarmSeverity.NO_ALARM: 0>, 'precision': 0, 'lower_ctrl_limit': 0.0,  
'upper_ctrl_limit': 0.0, 'units': '', 'sub_type': 'value', 'obj':  
EpicsSignalRO(read_pv='kad:Stats1:MaxValue_RBV', name='cdevice_max_value',  
parent='cdevice', value=165.0, timestamp=1706735231.362738, auto_monitor=True,  
string=False)}
```

```
[ ]: stats_id.read()
```

```
[ ]: {'stats_id': {'value': 1112930, 'timestamp': 1706734086.168137}}
```

```
[ ]: from ophyd import Signal, Device, Component  
  
class ComputedSignalDevice(Device):  
    max_value = Component(EpicsSignalRO, "kad:Stats1:MaxValue_RBV",  
↳kind="hinted")  
    min_value = Component(EpicsSignalRO, "kad:Stats1:MinValue_RBV",  
↳kind="hinted")  
    computed = Component(Signal, value=0, kind="hinted")  
  
    def __init__(self, *args, **kwargs):  
        super().__init__(*args, **kwargs)  
        self.max_value.subscribe(self.cb_max_value)  
  
    def cb_max_value(self, *args, **kwargs):  
        print(f"cb_max_value: {args=} {kwargs=}")  
  
    def trigger(self, *args, **kwargs):  
        super().trigger(*args, **kwargs)  
  
    # TODO: this device is incomplete  
    # Need to know how to wait for all signals to cominebereceived before  
    # computing the 'computed' signal.
```

```
# Then verify that RE(bp.scan([this_device])) will wait for  
# the new computed result.
```

```
cdevice = ComputedSignalDevice("", name="cdevice")
```

```
cb_max_value: args=() kwargs={'old_value': <object object at 0x7f5aac79e610>,  
'value': 163.0, 'timestamp': 1706734086.168164, 'status': <AlarmStatus.NO_ALARM:  
0>, 'severity': <AlarmSeverity.NO_ALARM: 0>, 'precision': 0, 'lower_ctrl_limit':  
0.0, 'upper_ctrl_limit': 0.0, 'units': '', 'sub_type': 'value', 'obj':  
EpicsSignalRO(read_pv='kad:Stats1:MaxValue_RBV', name='cdevice_max_value',  
parent='cdevice', value=163.0, timestamp=1706734086.168164, auto_monitor=True,  
string=False)}
```