NAME
> **ibv_attach_counters_point_flow** − attach individual counter definition to a flow object

SYNOPSIS
```
#include <infiniband/verbs.h>

int ibv_attach_counters_point_flow(struct ibv_counters *counters,
                                   struct ibv_counter_attach_attr *counter_
                                   struct ibv_flow *flow);
```

DESCRIPTION
> Attach counters point are a family of APIs to attach individual counter description definition to a verb object at a specific index location.

> Counters object will start collecting values after it is bound to the verb object resource.

> A static attach can be created when NULL is provided instead of the reference to the verbs object (e.g.: in case of flow providing NULL instead of *flow*). In this case, this counters object will only start collecting values after it is bound to the verbs resource, for flow this is when referencing the counters handle when creating a flow with **ibv_create_flow**().

> Once an ibv_counters is bound statically to a verbs resource, no additional attach is allowed till the counter object is not bound to any verb object.

> The argument counter_desc specifies which counter value should be collected. It is defined in verbs.h as one of the enum ibv_counter_description options.

> Supported capabilities of specific counter_desc values per verbs object can be tested by checking the return value for success or ENOTSUP errno.

> Attaching a counters handle to multiple objects of the same type will accumulate the values into a single index. e.g.: creating several ibv_flow(s) with the same ibv_counters handle will collect the values from all relevant flows into the relevant index location when reading the values from **ibv_read_counters**(), setting the index more than once with different or same counter_desc will aggregate the values from all relevant counters into the relevant index location.

> The runtime values of counters can be read from the hardware by calling **ibv_read_counters**().

ARGUMENTS
> *counters*
>> Existing counters to attach new counter point on.

> *counter_attach_attr*
>> An ibv_counter_attach_attr struct, as defined in verbs.h.

> *flow*   Existing flow to attach a new counters point on (in static mode it must be NULL).

counter_attach_attr **Argument**
```
struct ibv_counter_attach_attr {
    enum ibv_counter_description counter_desc;
    uint32_t index;
    uint32_t comp_mask;
};
```

counter_desc **Argument**
```
enum ibv_counter_description {
    IBV_COUNTER_PACKETS,
    IBV_COUNTER_BYTES,
};
```

> *index*   Desired location of the specific counter at the counters object.

comp_mask
> Bitmask specifying what fields in the structure are valid.

RETURN VALUE
ibv_attach_counters_point_flow() returns 0 on success, or the value of errno on failure (which indicates the failure reason)

ERRORS
EINVAL
> invalid argument(s) passed

ENOTSUP
> *counter_desc* is not supported on the requested object

EBUSY
> the counter object is already bound to a flow, additional attach calls is not allowed (valid for static attach only)

ENOMEM
> not enough memory

NOTES
Counter values in each index location are cleared upon creation when calling **ibv_create_counters**(). Attaching counters points will only increase these values accordingly.

EXAMPLE
An example of use of **ibv_attach_counters_point_flow**() is shown in **ibv_read_counters**

SEE ALSO
ibv_create_counters, **ibv_destroy_counters**, **ibv_read_counters**, **ibv_create_flow**

AUTHORS
Raed Salem <raeds@mellanox.com>

Alex Rosenbaum <alexr@mellanox.com>