

**NAME**

**bus\_adjust\_resource** - adjust resource allocated from a parent bus

**SYNOPSIS**

```
#include <sys/param.h>
```

```
#include <sys/bus.h>
```

```
#include <machine/bus.h>
```

```
#include <sys/rman.h>
```

```
#include <machine/resource.h>
```

```
int
```

```
bus_adjust_resource(device_t dev, int type, struct resource *r, rman_res_t start, rman_res_t end);
```

**DESCRIPTION**

This function is used to ask the parent bus to adjust the resource range assigned to an allocated resource. The resource *r* should have been allocated by a previous call to `bus_alloc_resource(9)`. The new resource range must overlap the existing range of *r*. The *type* argument should match the *type* argument passed to `bus_alloc_resource(9)` when the resource was initially allocated.

Note that none of the constraints of the original allocation request such as alignment or boundary restrictions are checked by `bus_adjust_resource()`. It is the caller's responsibility to enforce any such requirements.

**RETURN VALUES**

The `bus_adjust_resource()` method returns zero on success or an error code on failure.

**EXAMPLES**

Grow an existing memory resource by 4096 bytes.

```
struct resource *res;
```

```
int error;
```

```
error = bus_adjust_resource(dev, SYS_RES_MEMORY, res,
    rman_get_start(res), rman_get_end(res) + 0x1000);
```

**ERRORS**

`bus_adjust_resource()` will fail if:

[EINVAL]           The *dev* device does not have a parent device.

[EINVAL] The *r* resource is a shared resource.

[EINVAL] The new address range does not overlap with the existing address range of *r*.

[EBUSY] The new address range conflicts with another allocated resource.

**SEE ALSO**

bus\_alloc\_resource(9), bus\_release\_resource(9), device(9), driver(9)