

NAME

device_add_child, **device_add_child_ordered** - add a new device as a child of an existing device

SYNOPSIS

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

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

device_t

```
device_add_child(device_t dev, const char *name, int unit);
```

device_t

```
device_add_child_ordered(device_t dev, int order, const char *name, int unit);
```

DESCRIPTION

Create a new child device of *dev*. The *name* and *unit* arguments specify the name and unit number of the device. If the name is unknown then the caller should pass NULL. If the unit is unknown then the caller should pass -1 and the system will choose the next available unit number.

The name of the device is used to determine which drivers might be appropriate for the device. If a name is specified then only drivers of that name are probed. If no name is given then all drivers for the owning bus are probed. In any event, only the name of the device is stored so that one may safely unload/load a driver bound to that name.

This allows buses which can uniquely identify device instances (such as PCI) to allow each driver to check each device instance for a match. For buses which rely on supplied probe hints where only one driver can have a chance of probing the device, the driver name should be specified as the device name.

Normally unit numbers will be chosen automatically by the system and a unit number of -1 should be given. When a specific unit number is desired (e.g., for wiring a particular piece of hardware to a pre-configured unit number), that unit should be passed. If the specified unit number is already allocated, a new unit will be allocated and a diagnostic message printed.

If the devices attached to a bus must be probed in a specific order (e.g., for the ISA bus some devices are sensitive to failed probe attempts of unrelated drivers and therefore must be probed first), the *order* argument of **device_add_child_ordered**() should be used to specify a partial ordering. The new device will be added before any existing device with a greater order. If **device_add_child**() is used, then the new child will be added as if its order was zero.

When adding a device in the context of DEVICE_IDENTIFY(9) routine, the `device_find_child(9)` routine should be used to ensure that the device has not already been added to the tree. Because the

device name and *devclass_t* are associated at probe time (not child addition time), previous instances of the driver (say in a module that was later unloaded) may have already added the instance. Authors of bus drivers must likewise be careful when adding children when they are loaded and unloaded to avoid duplication of children devices.

When adding a child to another device node, such as in an identify routine, use `BUS_ADD_CHILD(9)` instead of `device_add_child(9)`. `BUS_ADD_CHILD(9)` will call `device_add_child(9)` and add the proper bus-specific data to the new child. `device_add_child()` does not call `BUS_ADD_CHILD(9)`.

RETURN VALUES

The new device if successful, NULL otherwise.

SEE ALSO

`BUS_ADD_CHILD(9)`, `device(9)`, `device_delete_child(9)`, `device_find_child(9)`,
`DEVICE_IDENTIFY(9)`

AUTHORS

This manual page was written by Doug Rabson.