

**NAME**

**pca954x** - driver for PCA9548A I2C switch

**SYNOPSIS**

To compile this driver into the kernel, place the following line in your kernel configuration file:

```
device pca954x
device iicmux
device iicbus
```

Alternatively, to load the driver as a module at boot time, place the following line in loader.conf(5):

```
pca954x_load="YES"
```

**DESCRIPTION**

The **pca954x** driver supports the PCA9548A I2C bus switch and compatible chips such as TCA9548A. It automatically connects an upstream I2C bus to one of several downstream buses as needed when slave devices on the downstream buses initiate I/O. More information on the automatic switching behavior is available in iicmux(4).

**FDT CONFIGURATION**

On an FDT(4) based system, an **pca954x** device node is defined as a child node of its upstream I2C bus. The children of the **pca954x** node are additional I2C buses, which will have their own I2C slave devices described in their child nodes.

The **pca954x** driver attaches to nodes where the *compatible* property is set to one of

- "nxp,pca9548"

The **pca954x** driver supports the following optional properties in addition to the standard I2C mux properties:

*i2c-mux-idle-disconnect* if defined, forces the switch to disconnect all children in idle state.

**HINTS CONFIGURATION**

On a device.hints(5) based system, these values are configurable for **pca954x**:

*hint.pca954x.<unit>.at* The upstream iicbus(4) the **pca954x** instance is attached to.

*hint.pca954x.<unit>.chip\_type* The type of the chip. At present, only "pca9548" is supported.

When configured via hints, the driver automatically adds an `iicbus(4)` instance for every downstream bus supported by the chip. There is currently no way to indicate used versus unused channels.

**SEE ALSO**

`iicbus(4)`, `iicmux(4)`

**HISTORY**

The **pca954x** driver and this manual page was written by Andriy Gapon <[avg@FreeBSD.org](mailto:avg@FreeBSD.org)>.