

interrupt
interruptList
numInterrupts
setInterruptList:num:

Getting and setting the list of memory ranges

memoryRangeList
numMemoryRanges
setMemoryRangeList:num:

Getting and setting the port devicePort

setDevicePort:

Getting and setting the direct device

directDevice
setDirectDevice:

Getting and setting the configuration table

configTable
setConfigTable:

(IOConfigTable *)configTable

Returns the table of configuration information for this driver instance.

setConfigTable:

(port_t)devicePort

Returns the device port associated with the device. This port is used by the Driver Kit. You shouldn't need to invoke this method if your driver uses only supported Driver Kit API.

Holding send rights to the device port gives a task rights to access a device's registers, to program its DMA channel, and receive interrupt notification. The kernel responds to requests sent on this port to provide these services to the requesting

object to which this driver instance is connected. Usually, the returned object is an `IODirectDevice` object. If this `IODeviceDescription`'s `object` is a direct or pseudo device driver, this method returns `setDirectDevice`:

`(unsigned int)interrupt`

Returns the first interrupt (IRQ) associated with this device. The return value is undefined if this device is not associated with it.

`interruptList, numInterrupts, setInterruptList:num:`

`(unsigned int *)interruptList`

Returns all the interrupts (IRQs) associated with this device. You can get the number of items in the array by invoking `numInterrupts`. You should never free the data returned by this method.

`interrupt, numInterrupts, setInterruptList:num:`

`(IORange *)memoryRangeList`

Returns all the memory ranges associated with this device. You can get the number of items in the array by invoking `numMemoryRanges`. You should never free the data returned by this method.

`numMemoryRanges, setMemoryRangeList:num:`

`(unsigned int)numInterrupts`

Returns the total number of interrupts (IRQs) associated with this device.

`interrupt, interruptList, setInterruptList:num:`

`(unsigned int)numMemoryRanges`

Returns the total number of memory ranges associated with this device.

`memoryRangeList, setMemoryRangeList:num:`

`(void)setConfigTable:(IOConfigTable *)configTable`

Sets the table of configuration information for this driver instance. In normal use of the Driver Kit, you should not invoke this method.

`configTable`

Records `directDevice` as the `IODevice` object that is connected to the driver instance that this `IODevice` describes. In normal use of the Driver Kit, you should never invoke this method.

`directDevice`

`(IOReturn)setInterruptList:(unsigned int *)aList num:(unsigned int)numInterrupts`

Sets the array and number of interrupts (IRQs) associated with this device. You shouldn't normally invoke this method since it overrides the normal configuration scheme (which is documented in Chapter 4).

`interrupt, interruptList, numInterrupts`

`(IOReturn)setMemoryRangeList:(IORange *)aList num:(unsigned int)numMemoryRanges`

Sets the array and number of memory ranges associated with this device. You shouldn't normally invoke this method since it overrides the normal configuration scheme (which is documented in Chapter 4).

`memoryRangeList, numMemoryRanges`