

checkRangesForConflicts:num:
Setting and getting the start address
checkText:
setStartAddress:
startAddress
Action methods minus:
plus:
Setting and getting the range length
rangeLength
setRangeLength:
Limiting the address range setAddressLimits::
Assigning a delegate setDelegate:
delegate
Delegate methods rangeDidChange:

(BOOL)checkRangesForConflicts:(IOAddressRanger *)ranges num:(unsigned int)numRanges

A configuration inspector invokes this method to check whether this IOAddressRanger uses any addresses already used by the specified IOAddressRangers. If so, this method changes the color of the text in the text field to gray and sets the status button on. If no conflicts exist, this method sets the status button off and changes the color of the text to black. Returns NO if no conflicts exist and YES if conflicts exist.

checkText:sender

Checks whether sender's string value is an address and, if so, sets the range's start address (adjusted as described in the class description), updates the display, and sends the delegate a rangeDidChange: message. If the string isn't an address, the system beeps and updates the range's display. Returns self if the string is an address otherwise, returns nil.

delegate

Returns the IOAddressRanger's delegate, or nil if it doesn't have one.

plus:sender

This method is the target of the plus button in the IOAddressRanger. It moves the range higher by range's length (but not above the higher limit), updates the display, and sends the delegate a rangeDidChange message. As an example, if the range is currently from 0x000e00 to 0x000eff, this method changes the range to 0x000fff. Returns self.

(unsigned long)rangeLength

Returns the length of the range. This length should be set at initialization using setRangeLength:.

setAddressLimits:(unsigned long)low :(unsigned long)high

Limits the address range to values between low (inclusive) and high (inclusive) and adjusts the start address in the class description. Returns self.

setDelegate:anObject

Makes anObject the IOAddressRanger's delegate, and returns self. The delegate is sent a rangeDidChange message whenever the address range changes.

setRangeLength:(unsigned long)length

Sets the length of the range and returns self. The new length is displayed and the start address is adjusted in the class description.

setStartAddress:(unsigned long)address

Sets the start address of the range (adjusted as described in the class description) and returns self.

(unsigned long)startAddress

Returns the start of the range. This length should be set at initialization using setStartAddress:.

rangeDidChange:sender

Notifies the delegate that the range changed.

