# IO.Bindings Namespace

Wrapper classes for C shared libraries.

## Classes

| | Class | Description |
|---|---|---|
| | libsimpleio | Wrapper class for the Linux Simple I/O Library `libsimpleio.so` |

# libsimpleio Class

Wrapper class for the Linux Simple I/O Library `libsimpleio.so`

## Inheritance Hierarchy

SystemObject IO.Bindingslibsimpleio

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static class libsimpleio
```

The libsimpleio type exposes the following members.

## Methods

| | Name | Description |
|---|---|---|
| | ADC_close | Close a Linux IIO A/D converter input device. |
| | ADC_get_name | Get the subsystem name for the specified Linux IIO A/D converter device. |
| | ADC_open | Open a Linux IIO A/D converter input device. |
| | | |

| | | |
|---|---|---|
| | ADC_read | Read a Linux IIO A/D converter input device. |
| S | DAC_close | Close a Linux IIO D/A converter output device. |
| S | DAC_get_name | Get the subsystem name for the specified Linux IIO D/A converter device. |
| S | DAC_open | Open a Linux IIO D/A converter output device. |
| S | DAC_write | Write to a Linux IIO D/A converter output device. |
| S | EVENT_close | Close an `epoll` event dispatcher. |
| S | EVENT_modify_fd | Modify a file registration. |
| S | EVENT_open | Open an `epoll` event dispatcher. |
| S | EVENT_register_fd | Register a file descriptor with an `epoll` event dispatcher. |
| S | EVENT_unregister_fd | Unregister a file from an `epoll` dispatcher. |
| S | EVENT_wait | Wait for events from an `epoll` dispatcher. |
| S | GPIO_chip_info | Get GPIO chip information. |

| | | |
|---|---|---|
| | GPIO_close | Close a Linux GPIO pin device. |
| | GPIO_configure | Configure a Linux GPIO pin. |
| | GPIO_line_close | Close a single GPIO line. |
| | GPIO_line_event | Read an edge trigger event from single GPIO line. |
| | GPIO_line_info | Get GPIO line information. |
| | GPIO_line_open | Open a single GPIO line. |
| | GPIO_line_read | Read the state of a single GPIO line. |
| | GPIO_line_write | Write the state of a single GPIO line. |
| | GPIO_open | Open a Linux GPIO pin device. |
| | GPIO_read | Read a Linux GPIO pin. |
| | GPIO_write | Write a Linux GPIO pin. |
| | HIDRAW_close | Close a Linux raw HID. |
| | HIDRAW_get_info | Get Linux raw HID bus type, vendor ID, and product ID. |
| | | |

| | | |
|---|---|---|
| | HIDRAW_get_name | Get Linux raw HID name string. |
| S | HIDRAW_open1 | Open a Linux raw HID device by device node name. |
| S | HIDRAW_open2 | Open a Linux raw HID device by vendor ID and product ID. |
| S | HIDRAW_open3 | Open a Linux raw HID device by vendor ID and product ID and serial number. |
| S | HIDRAW_receive | Get a 64-byte report from a Linux HID. |
| S | HIDRAW_send | Send a 64-byte report to a Linux HID. |
| S | I2C_close | Close a Linux $I^2C$ bus controller device. |
| S | I2C_open | Open a Linux $I^2C$ bus controller device. |
| S | I2C_transaction | Send bytes to and/or receive bytes from an $I^2C$ slave device. |
| S | IPV4_ntoa | Convert an IPv4 address to a dotted notation string (*e.g.* 1.2.3.4). |

| | | |
|---|---|---|
| | IPV4_resolve | Resolve a domain name to an IPv4 host address. |
| | LINUX_closelog | Close the connection to the `syslog` service. |
| | LINUX_command | Execute a shell command string. |
| | LINUX_detach | Detach the process and run it in the background. |
| | LINUX_drop_privileges | Drop process privileges to those of the specified user.<br>**Remarks**<br>Only a process running at superuser privilege is allowed to drop privileges. |
| | LINUX_errno | Fetch the value of `errno`. |
| | LINUX_openlog | Open a connection to the `syslog` service. |
| | LINUX_poll | Wait for an event on one or more files. |
| | LINUX_strerror | Retrieve the error message for a particular `errno` error code. |
| | | |

| | | |
|---|---|---|
| | LINUX_syslog | Send a message to the `syslog` service. |
| S | LINUX_usleep | Sleep for the specified number of microseconds. |
| S | PWM_close | Close a Linux PWM output device. |
| S | PWM_configure | Configure a Linux PWM output device. |
| S | PWM_open | Open a Linux PWM output device. |
| S | PWM_write | Set a Linux PWM output device duty cycle. |
| S | SERIAL_close | Close a Linux serial port device. |
| S | SERIAL_open | Open a Linux serial port device. |
| S | SERIAL_receive | Receive data from a Linux serial port device. |
| S | SERIAL_send | Send data to a Linux serial port device. |
| S | SPI_close | Close a Linux SPI device. |
| S | SPI_open | Open a Linux SPI device. |
| S | SPI_transaction | Send bytes to and/or |

| | | |
|---|---|---|
| | | receive bytes from a Linux SPI device. |
| S | STREAM_decode_frame | Decode a frame. |
| S | STREAM_encode_frame | Encode a frame. |
| S | STREAM_receive_frame | Receive an encoded frame. |
| S | STREAM_send_frame | Send an encoded frame. |
| S | TCP4_accept | Start TCP server and wait for a single connection. |
| S | TCP4_close | Close a TCP connection. |
| S | TCP4_connect | Connect to a TCP server. |
| S | TCP4_receive | Receive bytes from TCP peer. |
| S | TCP4_send | Send bytes to TCP peer. |
| S | TCP4_server | Start a TCP server and fork for each connection. |
| S | UDP4_close | Close a UDP socket. |
| S | UDP4_open | Open a UDP socket. |
| S | UDP4_receive | Receive a UDP datagram. |
| S | UDP4_send | Send a UDP datagram. |
| S | WATCHDOG_close | Close a Linux watchdog |

| | | |
|---|---|---|
| | | timer device. |
| S | WATCHDOG_get_timeout | Query a Linux watchdog timer device. |
| S | WATCHDOG_kick | Reset the watchdog timer. |
| S | WATCHDOG_open | Open a Linux watchdog timer device. |
| S | WATCHDOG_set_timeout | Change the watchdog timer period. |

Top

# Fields

| | Name | Description |
|---|---|---|
| S | GPIO_DIRECTION_INPUT | Input data direction. |
| S | GPIO_DIRECTION_OUTPUT | Out data direction. |
| S | GPIO_DRIVER_OPENDRAIN | Open drain (sink only) output driver. |
| S | GPIO_DRIVER_OPENSOURCE | Open source (source only) output driver |
| S | GPIO_DRIVER_PUSHPULL | Push-pull (source and sink) output driver. |
| S | GPIO_EDGE_BOTH | Interrupt on both edges. |
| S | GPIO_EDGE_FALLING | Interrupt on falling |

| | | |
|---|---|---|
| | | edge. |
| S | GPIO_EDGE_NONE | Interrupts are disabled. |
| S | GPIO_EDGE_RISING | Interrupt on rising edge. |
| S | GPIO_EVENT_REQUEST_BOTH | Enable GPIO input interrupt on both edges. |
| S | GPIO_EVENT_REQUEST_FALLING | Enable GPIO input interrupt on falling edge. |
| S | GPIO_EVENT_REQUEST_NONE | Disable GPIO input interrupt. |
| S | GPIO_EVENT_REQUEST_RISING | Enable GPIO input interrupt on rising edge. |
| S | GPIO_LINE_INFO_ACTIVE_LOW | GPIO line is configured as active low (inverted). |
| S | GPIO_LINE_INFO_KERNEL | GPIO line is being used by the kernel. |
| S | GPIO_LINE_INFO_OPEN_DRAIN | GPIO line is configured as open drain (current sink only). |
| S | GPIO_LINE_INFO_OPEN_SOURCE | GPIO line is |

| | | |
|---|---|---|
| | | configured as open source (current source only). |
| S | GPIO_LINE_INFO_OUTPUT | GPIO line is configured as an output. |
| S | GPIO_LINE_REQUEST_ACTIVE_HIGH | Select GPIO line polarity active high (normal). |
| S | GPIO_LINE_REQUEST_ACTIVE_LOW | Select GPIO line polarity active low (inverted). |
| S | GPIO_LINE_REQUEST_INPUT | Select GPIO line direction input. |
| S | GPIO_LINE_REQUEST_OPEN_DRAIN | Select GPIO line driver open drain (current sink only). |
| S | GPIO_LINE_REQUEST_OPEN_SOURCE | Select GPIO line driver open source (current source only). |
| S | GPIO_LINE_REQUEST_OUTPUT | Select GPIO line direction output. |
| S | GPIO_LINE_REQUEST_PUSH_PULL | Select GPIO line driver push-pull (current source and sink). |
| S | GPIO_POLARITY_ACTIVEHIGH | Active high (normal) |

| | | |
|---|---|---|
| | | polarity. |
| S | GPIO_POLARITY_ACTIVELOW | Active low (inverted) polarity. |
| S | INADDR_ANY | IPv4 address for binding to all network interfaces. |
| S | INADDR_BROADCAST | IPv4 broadcast address. |
| S | INADDR_LOOPBACK | IPv4 address for binding to the loopback interface (aka `localhost`). |
| S | LOG_ALERT | Action must be taken immediately. |
| S | LOG_AUTH | Security/authorization messages. |
| S | LOG_AUTHPRIV | Securit/authorization messages. |
| S | LOG_CONS | Write directly to the system console if there is an error while sending to the system logger. |
| S | LOG_CRIT | Critical condition. |
| S | LOG_CRON | `cron` daemon messages. |

| | | |
|---|---|---|
| S | LOG_DAEMON | System daemons. |
| S | LOG_DEBUG | Debug message. |
| S | LOG_EMERG | System is unusable. |
| S | LOG_ERR | Error condition. |
| S | LOG_FTP | `FTP` daemon messages. |
| S | LOG_INFO | Informational message. |
| S | LOG_KERN | Kernel messages. |
| S | LOG_LOCAL0 | Reserved for local use. |
| S | LOG_LOCAL1 | Reserved for local use. |
| S | LOG_LOCAL2 | Reserved for local use. |
| S | LOG_LOCAL3 | Reserved for local use. |
| S | LOG_LOCAL4 | Reserved for local use. |
| S | LOG_LOCAL5 | Reserved for local use. |
| S | LOG_LOCAL6 | Reserved for local use. |

| | | |
|---|---|---|
| S | LOG_LOCAL7 | Reserved for local use. |
| S | LOG_LPR | Line printer subsystem |
| S | LOG_MAIL | Mail system. |
| S | LOG_NDELAY | Open the connection immediately. Do not wait until `syslog()` is called for the first time. |
| S | LOG_NEWS | Network news subsystem |
| S | LOG_NOTICE | Normal but significant condition. |
| S | LOG_NOWAIT | Don't wait for child processes that may have been created while logging the message. (Not applicable to `glibc`.) |
| S | LOG_ODELAY | Do not open the connection immediately. Wait until `syslog()` is called for the first time. |
| S | LOG_PERROR | Also log the message |

| | | |
|---|---|---|
| | | to `stderr`. |
| | LOG_PID | Include the caller's PID (process ID) with each message. |
| | LOG_PROGNAME | Use the program name for the identity string. |
| | LOG_SYSLOG | Messages generated internally by syslogd |
| | LOG_USER | Random user-level messages. |
| | LOG_UUCP | UUCP subsystem |
| | LOG_WARNING | Warning condition. |
| | MSG_DONTROUTE | Don't use a gateway to send out the packet, send to hosts only on directly connected networks. |
| | MSG_DONTWAIT | Enables nonblocking operation; if the operation would block, `EAGAIN` or `EWOULDBLOCK` is returned. |
| | MSG_MORE | The caller has more data to send. This flag informs the kernel to |

| | | |
|---|---|---|
| | | package all of the data sent in calls with this flag set into a single datagram which is transmitted only when a call is performed that does not specify this flag. |
| S | POLLERR | An error occurred. |
| S | POLLHUP | Peer closed connection. |
| S | POLLIN | There is data to read. |
| S | POLLNVAL | File descriptor is invalid. |
| S | POLLOUT | Writing is now possible. |
| S | POLLPRI | There is urgent data to read. |
| S | PWM_POLARITY_ACTIVEHIGH | Configure the PWM output as active high (normal). |
| S | PWM_POLARITY_ACTIVELOW | Configure the PWM output as active low (inverted). |
| S | SERIAL_PARITY_EVEN | Request even parity checking. |

| | | |
|---|---|---|
| | SERIAL_PARITY_NONE | Disable parity checking. |
| | SERIAL_PARITY_ODD | Request odd parity checking. |
| | SPI_AUTO_CS | Use hardware slave select. |

Top

## See Also

**Reference**

IO.Bindings Namespace

# libsimpleio Methods

The libsimpleio type exposes the following members.

## Methods

| | Name | Description |
|---|---|---|
| | ADC_close | Close a Linux IIO A/D converter input device. |
| | ADC_get_name | Get the subsystem name for the specified Linux IIO A/D converter device. |
| | ADC_open | Open a Linux IIO A/D converter input device. |
| | ADC_read | Read a Linux IIO A/D converter input device. |
| | DAC_close | Close a Linux IIO D/A converter output device. |
| | DAC_get_name | Get the subsystem name for the specified Linux IIO D/A converter device. |
| | DAC_open | Open a Linux IIO D/A |

| | | |
|---|---|---|
| | | converter output device. |
| S | DAC_write | Write to a Linux IIO D/A converter output device. |
| S | EVENT_close | Close an `epoll` event dispatcher. |
| S | EVENT_modify_fd | Modify a file registration. |
| S | EVENT_open | Open an `epoll` event dispatcher. |
| S | EVENT_register_fd | Register a file descriptor with an `epoll` event dispatcher. |
| S | EVENT_unregister_fd | Unregister a file from an `epoll` dispatcher. |
| S | EVENT_wait | Wait for events from an `epoll` dispatcher. |
| S | GPIO_chip_info | Get GPIO chip information. |
| S | GPIO_close | Close a Linux GPIO pin device. |
| S | GPIO_configure | Configure a Linux GPIO pin. |
| S | GPIO_line_close | Close a single GPIO line. |

| | | |
|---|---|---|
| S | GPIO_line_event | Read an edge trigger event from single GPIO line. |
| S | GPIO_line_info | Get GPIO line information. |
| S | GPIO_line_open | Open a single GPIO line. |
| S | GPIO_line_read | Read the state of a single GPIO line. |
| S | GPIO_line_write | Write the state of a single GPIO line. |
| S | GPIO_open | Open a Linux GPIO pin device. |
| S | GPIO_read | Read a Linux GPIO pin. |
| S | GPIO_write | Write a Linux GPIO pin. |
| S | HIDRAW_close | Close a Linux raw HID. |
| S | HIDRAW_get_info | Get Linux raw HID bus type, vendor ID, and product ID. |
| S | HIDRAW_get_name | Get Linux raw HID name string. |
| S | HIDRAW_open1 | Open a Linux raw HID device by device node name. |
| S | | |

| | | |
|---|---|---|
| | HIDRAW_open2 | Open a Linux raw HID device by vendor ID and product ID. |
| S | HIDRAW_open3 | Open a Linux raw HID device by vendor ID and product ID and serial number. |
| S | HIDRAW_receive | Get a 64-byte report from a Linux HID. |
| S | HIDRAW_send | Send a 64-byte report to a Linux HID. |
| S | I2C_close | Close a Linux $I^2C$ bus controller device. |
| S | I2C_open | Open a Linux $I^2C$ bus controller device. |
| S | I2C_transaction | Send bytes to and/or receive bytes from an $I^2C$ slave device. |
| S | IPV4_ntoa | Convert an IPv4 address to a dotted notation string (*e.g.* 1.2.3.4). |
| S | IPV4_resolve | Resolve a domain name to an IPv4 host address. |
| S | LINUX_closelog | Close the connection to the `syslog` service. |
| S | | |

| | | |
|---|---|---|
| | LINUX_command | Execute a shell command string. |
| S | LINUX_detach | Detach the process and run it in the background. |
| S | LINUX_drop_privileges | Drop process privileges to those of the specified user.<br>**Remarks**<br>Only a process running at superuser privilege is allowed to drop privileges. |
| S | LINUX_errno | Fetch the value of `errno`. |
| S | LINUX_openlog | Open a connection to the `syslog` service. |
| S | LINUX_poll | Wait for an event on one or more files. |
| S | LINUX_strerror | Retrieve the error message for a particular `errno` error code. |
| S | LINUX_syslog | Send a message to the `syslog` service. |
| S | LINUX_usleep | Sleep for the specified |

| | | |
|---|---|---|
| | | number of microseconds. |
| S | PWM_close | Close a Linux PWM output device. |
| S | PWM_configure | Configure a Linux PWM output device. |
| S | PWM_open | Open a Linux PWM output device. |
| S | PWM_write | Set a Linux PWM output device duty cycle. |
| S | SERIAL_close | Close a Linux serial port device. |
| S | SERIAL_open | Open a Linux serial port device. |
| S | SERIAL_receive | Receive data from a Linux serial port device. |
| S | SERIAL_send | Send data to a Linux serial port device. |
| S | SPI_close | Close a Linux SPI device. |
| S | SPI_open | Open a Linux SPI device. |
| S | SPI_transaction | Send bytes to and/or receive bytes from a Linux SPI device. |
| S | | |

| | | |
|---|---|---|
| | STREAM_decode_frame | Decode a frame. |
| S | STREAM_encode_frame | Encode a frame. |
| S | STREAM_receive_frame | Receive an encoded frame. |
| S | STREAM_send_frame | Send an encoded frame. |
| S | TCP4_accept | Start TCP server and wait for a single connection. |
| S | TCP4_close | Close a TCP connection. |
| S | TCP4_connect | Connect to a TCP server. |
| S | TCP4_receive | Receive bytes from TCP peer. |
| S | TCP4_send | Send bytes to TCP peer. |
| S | TCP4_server | Start a TCP server and fork for each connection. |
| S | UDP4_close | Close a UDP socket. |
| S | UDP4_open | Open a UDP socket. |
| S | UDP4_receive | Receive a UDP datagram. |
| S | UDP4_send | Send a UDP datagram. |

| | | |
|---|---|---|
| | WATCHDOG_close | Close a Linux watchdog timer device. |
| | WATCHDOG_get_timeout | Query a Linux watchdog timer device. |
| | WATCHDOG_kick | Reset the watchdog timer. |
| | WATCHDOG_open | Open a Linux watchdog timer device. |
| | WATCHDOG_set_timeout | Change the watchdog timer period. |

Top

## See Also

**Reference**

libsimpleio Class

IO.Bindings Namespace

# libsimpleioADC_close Method

Close a Linux IIO A/D converter input device.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static void ADC_close(
        int fd,
        out int error
)
```

### Parameters

*fd*
Type: SystemInt32
File descriptor.

*error*
Type: SystemInt32
Error code. Zero upon success or an `errno` value upon failure.

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioADC_get_name Method

Get the subsystem name for the specified Linux IIO A/D converter device.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static void ADC_get_name(
        int chip,
        StringBuilder name,
        int size,
        out int error
)
```

### Parameters

*chip*
Type: SystemInt32
Linux IIO device number.

*name*
Type: System.TextStringBuilder
Destination buffer.

*size*
Type: SystemInt32
Size of destination buffer.

*error*

Type: SystemInt32
Error code. Zero upon success or an `errno` value upon failure.

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioADC_open Method

Open a Linux IIO A/D converter input device.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static void ADC_open(
        int chip,
        int channel,
        out int fd,
        out int error
)
```

### Parameters

*chip*
Type: SystemInt32
Linux IIO device number.

*channel*
Type: SystemInt32
Input channel number.

*fd*
Type: SystemInt32
File descriptor.

*error*
Type: SystemInt32
Error code. Zero upon success or an `errno` value upon failure.

## See Also

### Reference

libsimpleio Class

IO.Bindings Namespace

# libsimpleioADC_read Method

Read a Linux IIO A/D converter input device.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static void ADC_read(
        int fd,
        out int sample,
        out int error
)
```

### Parameters

*fd*
Type: SystemInt32
File descriptor.

*sample*
Type: SystemInt32
Analog sample data.

*error*
Type: SystemInt32
Error code. Zero upon success or an `errno` value upon failure.

## See Also

### Reference

libsimpleio Class

## IO.Bindings Namespace

# libsimpleioDAC_close Method

Close a Linux IIO D/A converter output device.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static void DAC_close(
        int fd,
        out int error
)
```

### Parameters

*fd*
Type: SystemInt32
File descriptor.

*error*
Type: SystemInt32
Error code. Zero upon success or an `errno` value upon failure.

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioDAC_get_name Method

Get the subsystem name for the specified Linux IIO D/A converter device.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static void DAC_get_name(
        int chip,
        StringBuilder name,
        int size,
        out int error
)
```

### Parameters

*chip*
Type: SystemInt32
Linux IIO device number.

*name*
Type: System.TextStringBuilder
Destination buffer.

*size*
Type: SystemInt32
Size of destination buffer.

*error*

Type: SystemInt32
Error code. Zero upon success or an `errno` value upon failure.

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioDAC_open Method

Open a Linux IIO D/A converter output device.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static void DAC_open(
        int chip,
        int channel,
        out int fd,
        out int error
)
```

### Parameters

*chip*
Type: SystemInt32
Linux IIO device number.

*channel*
Type: SystemInt32
Output channel number.

*fd*
Type: SystemInt32
File descriptor.

*error*
Type: SystemInt32
Error code. Zero upon success or an `errno` value upon failure.

## See Also

### Reference

libsimpleio Class

IO.Bindings Namespace

# libsimpleioDAC_write Method

Write to a Linux IIO D/A converter output device.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static void DAC_write(
        int fd,
        int sample,
        out int error
)
```

### Parameters

*fd*
Type: SystemInt32
File descriptor.

*sample*
Type: SystemInt32
Analog sample data.

*error*
Type: SystemInt32
Error code. Zero upon success or an `errno` value upon failure.

## See Also

### Reference

libsimpleio Class

# IO.Bindings Namespace

# libsimpleioEVENT_close Method

Close an `epoll` event dispatcher.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static void EVENT_close(
        int epfd,
        out int error
)
```

### Parameters

*epfd*
Type: SystemInt32
File descriptor.

*error*
Type: SystemInt32
Error code. Zero upon success or an `errno` value upon failure.

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioEVENT_modify_fd Method

Modify a file registration.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static void EVENT_modify_fd(
        int epfd,
        int fd,
        int events,
        int handle,
        out int error
)
```

### Parameters

*epfd*
Type: SystemInt32
File descriptor for the dispatcher.

*fd*
Type: SystemInt32
File descriptor to register for events.

*events*
Type: SystemInt32
Events to register for. May be a sum of the individual event flags.

*handle*

Type: SystemInt32
Event handle.

*error*

Type: SystemInt32
Error code. Zero upon success or an `errno` value upon failure.

## See Also

**Reference**

libsimpleio Class
IO.Bindings Namespace

# libsimpleioEVENT_open Method

Open an `epoll` event dispatcher.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static void EVENT_open(
        out int epfd,
        out int error
)
```

### Parameters

*epfd*
Type: SystemInt32
File descriptor.

*error*
Type: SystemInt32
Error code. Zero upon success or an `errno` value upon failure.

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioEVENT_register_fd Method

Register a file descriptor with an `epoll` event dispatcher.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static void EVENT_register_fd(
        int epfd,
        int fd,
        int events,
        int handle,
        out int error
)
```

### Parameters

*epfd*
Type: SystemInt32
File descriptor for the dispatcher.

*fd*
Type: SystemInt32
File descriptor to register for events.

*events*
Type: SystemInt32
Events to register for. May be a sum of the individual event flags.

*handle*

Type: SystemInt32
Event handle.

*error*

Type: SystemInt32
Error code. Zero upon success or an `errno` value upon failure.

## See Also

**Reference**

libsimpleio Class
IO.Bindings Namespace

# libsimpleioEVENT_unregister_fd Method

Unregister a file from an `epoll` dispatcher.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static void EVENT_unregister_fd(
        int epfd,
        int fd,
        out int error
)
```

### Parameters

*epfd*
Type: SystemInt32
File descriptor for the dispatcher.>

*fd*
Type: SystemInt32
File descriptor.

*error*
Type: SystemInt32
Error code. Zero upon success or an `errno` value upon failure.

## See Also

**Reference**

libsimpleio Class

IO.Bindings Namespace

# libsimpleioEVENT_wait Method

Wait for events from an `epoll` dispatcher.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static void EVENT_wait(
        int epfd,
        out int fd,
        out int events,
        out int handle,
        int timeoutms,
        out int error
)
```

### Parameters

*epfd*
Type: SystemInt32
File descriptor for the displatcher.

*fd*
Type: SystemInt32
File descriptor the event is applicable to.

*events*
Type: SystemInt32
Events that occurred. May be a sum of the individual event flags.

*handle*

Type: SystemInt32
Event handle provided whent the file descriptor was registered.

*timeoutms*

Type: SystemInt32
Time in milliseconds to wait for an event.

*error*

Type: SystemInt32
Error code. Zero upon success or an `errno` value upon failure.

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioGPIO_chip_info Method

Get GPIO chip information.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static void GPIO_chip_info(
        int chip,
        StringBuilder name,
        int namesize,
        StringBuilder label,
        int labelsize,
        out int lines,
        out int error
)
```

### Parameters

*chip*
Type: SystemInt32
GPIO chip number.

*name*
Type: System.TextStringBuilder
GPIO chip name.

*namesize*

Type: SystemInt32
Maximum size of name.

*label*

Type: System.TextStringBuilder
GPIO chip label.

*labelsize*

Type: SystemInt32
Maximum size of label.

*lines*

Type: SystemInt32
Number of GPIO lines.

*error*

Type: SystemInt32
Error code. Zero upon success or an `errno` value upon failure.

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioGPIO_close Method

Close a Linux GPIO pin device.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static void GPIO_close(
        int fd,
        out int error
)
```

### Parameters

*fd*
Type: SystemInt32
File descriptor.

*error*
Type: SystemInt32
Error code. Zero upon success or an `errno` value upon failure.

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioGPIO_configure Method

Configure a Linux GPIO pin.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static void GPIO_configure(
        int pin,
        int direction,
        int state,
        int edge,
        int polarity,
        out int error
)
```

### Parameters

*pin*
Type: SystemInt32
Pin number.

*direction*
Type: SystemInt32
Data direction.

*state*
Type: SystemInt32
Initial GPIO output state.

*edge*

Type: SystemInt32

Interrupt edge for input pin.

*polarity*

Type: SystemInt32

Polarity

*error*

Type: SystemInt32

Error code. Zero upon success or an `errno` value upon failure.

## See Also

### Reference

libsimpleio Class

IO.Bindings Namespace

# libsimpleioGPIO_line_close Method

Close a single GPIO line.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static void GPIO_line_close(
        int fd,
        out int error
)
```

### Parameters

*fd*
Type: SystemInt32
Linux file descriptor.

*error*
Type: SystemInt32
Error code. Zero upon success or an `errno` value upon failure.

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioGPIO_line_event Method

Read an edge trigger event from single GPIO line.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static void GPIO_line_event(
        int fd,
        out int state,
        out int error
)
```

### Parameters

*fd*
Type: SystemInt32
Linux file descriptor.

*state*
Type: SystemInt32
State of the GPIO line after the edge trigger event.

*error*
Type: SystemInt32
Error code. Zero upon success or an `errno` value upon failure.

## See Also

**Reference**

libsimpleio Class

IO.Bindings Namespace

# libsimpleioGPIO_line_info Method

Get GPIO line information.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static void GPIO_line_info(
        int chip,
        int line,
        StringBuilder name,
        int namesize,
        StringBuilder label,
        int labelsize,
        out int error
)
```

### Parameters

*chip*
Type: SystemInt32
GPIO chip number.

*line*
Type: SystemInt32
GPIO line number.

*name*

Type: System.TextStringBuilder
GPIO line name.

*namesize*

Type: SystemInt32
Maximum size of name.

*label*

Type: System.TextStringBuilder
GPIO line label.

*labelsize*

Type: SystemInt32
Maximum size of label.

*error*

Type: SystemInt32
Error code. Zero upon success or an `errno` value upon failure.

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioGPIO_line_open Method

Open a single GPIO line.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static void GPIO_line_open(
        int chip,
        int line,
        int flags,
        int events,
        int state,
        out int fd,
        out int error
)
```

### Parameters

*chip*
Type: SystemInt32
GPIO chip number.

*line*
Type: SystemInt32
GPIO line number.

*flags*

Type: SystemInt32
GPIO line configuration flags.

*events*

Type: SystemInt32
GPIO line event flags.

*state*

Type: SystemInt32
Initial GPIO output state.

*fd*

Type: SystemInt32
Linux file descriptor.

*error*

Type: SystemInt32
Error code. Zero upon success or an `errno` value upon failure.

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioGPIO_line_read Method

Read the state of a single GPIO line.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static void GPIO_line_read(
        int fd,
        out int state,
        out int error
)
```

### Parameters

*fd*
Type: SystemInt32
Linux file descriptor.

*state*
Type: SystemInt32
State of the GPIO line.

*error*
Type: SystemInt32
Error code. Zero upon success or an `errno` value upon failure.

## See Also

**Reference**

libsimpleio Class

IO.Bindings Namespace

# libsimpleioGPIO_line_write Method

Write the state of a single GPIO line.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static void GPIO_line_write(
        int fd,
        int state,
        out int error
)
```

### Parameters

*fd*
Type: SystemInt32
Linux file descriptor.

*state*
Type: SystemInt32
State of the GPIO line.

*error*
Type: SystemInt32
Error code. Zero upon success or an `errno` value upon failure.

## See Also

**Reference**

libsimpleio Class

IO.Bindings Namespace

# libsimpleioGPIO_open Method

Open a Linux GPIO pin device.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static void GPIO_open(
        int pin,
        out int fd,
        out int error
)
```

### Parameters

*pin*
Type: SystemInt32
Pin number.

*fd*
Type: SystemInt32
File descriptor.

*error*
Type: SystemInt32
Error code. Zero upon success or an `errno` value upon failure.

## See Also

### Reference

libsimpleio Class

## IO.Bindings Namespace

# libsimpleioGPIO_read Method

Read a Linux GPIO pin.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static void GPIO_read(
        int fd,
        out int state,
        out int error
)
```

### Parameters

*fd*
Type: SystemInt32
File descriptor.

*state*
Type: SystemInt32
Pin state.

*error*
Type: SystemInt32
Error code. Zero upon success or an `errno` value upon failure.

## See Also

### Reference

libsimpleio Class

# IO.Bindings Namespace

# libsimpleioGPIO_write Method

Write a Linux GPIO pin.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static void GPIO_write(
        int fd,
        int state,
        out int error
)
```

### Parameters

*fd*
Type: SystemInt32
File descriptor.

*state*
Type: SystemInt32
Pin state.

*error*
Type: SystemInt32
Error code. Zero upon success or an `errno` value upon failure.

## See Also

### Reference

libsimpleio Class

# IO.Bindings Namespace

# libsimpleioHIDRAW_close Method

Close a Linux raw HID.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static void HIDRAW_close(
        int fd,
        out int error
)
```

### Parameters

*fd*
Type: SystemInt32
File descriptor.

*error*
Type: SystemInt32
Error code. Zero upon success or an `errno` value upon failure.

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioHIDRAW_get_info Method

Get Linux raw HID bus type, vendor ID, and product ID.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static void HIDRAW_get_info(
        int fd,
        out int bustype,
        out int vendor,
        out int product,
        out int error
)
```

### Parameters

*fd*
Type: SystemInt32
File descriptor.

*bustype*
Type: SystemInt32
Bus type.

*vendor*
Type: SystemInt32
Vendor ID.

*product*

Type: SystemInt32
Product ID.

*error*

Type: SystemInt32
Error code. Zero upon success or an `errno` value upon failure.

## See Also

**Reference**

libsimpleio Class
IO.Bindings Namespace

# libsimpleioHIDRAW_get_name Method

Get Linux raw HID name string.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static void HIDRAW_get_name(
        int fd,
        StringBuilder name,
        int size,
        out int error
)
```

### Parameters

*fd*
Type: SystemInt32
File descriptor.

*name*
Type: System.TextStringBuilder
Destination buffer.

*size*
Type: SystemInt32
Size of destination buffer.

*error*

Type: SystemInt32
Error code. Zero upon success or an `errno` value upon failure.

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioHIDRAW_open1 Method

Open a Linux raw HID device by device node name.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static void HIDRAW_open1(
        string devname,
        out int fd,
        out int error
)
```

### Parameters

*devname*
Type: SystemString
Device node name.

*fd*
Type: SystemInt32
Device node name.

*error*
Type: SystemInt32
Error code. Zero upon success or an `errno` value upon failure.

## See Also

**Reference**

libsimpleio Class

IO.Bindings Namespace

# libsimpleioHIDRAW_open2 Method

Open a Linux raw HID device by vendor ID and product ID.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static void HIDRAW_open2(
        int VID,
        int PID,
        out int fd,
        out int error
)
```

### Parameters

*VID*
Type: SystemInt32
Vendor ID.

*PID*
Type: SystemInt32
Product ID.

*fd*
Type: SystemInt32
File descriptor.

*error*

Type: SystemInt32
Error code. Zero upon success or an `errno` value upon failure.

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioHIDRAW_open3 Method

Open a Linux raw HID device by vendor ID and product ID and serial number.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static void HIDRAW_open3(
        int VID,
        int PID,
        string serial,
        out int fd,
        out int error
)
```

### Parameters

*VID*
Type: SystemInt32
Vendor ID.

*PID*
Type: SystemInt32
Product ID.

*serial*
Type: SystemString
Serial number.

*fd*

Type: SystemInt32

File descriptor.

*error*

Type: SystemInt32

Error code. Zero upon success or an `errno` value upon failure.

## See Also

**Reference**

libsimpleio Class

IO.Bindings Namespace

# libsimpleioHIDRAW_receive Method

Get a 64-byte report from a Linux HID.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static void HIDRAW_receive(
        int fd,
        byte[] buf,
        int bufsize,
        out int count,
        out int error
)
```

### Parameters

*fd*
Type: SystemInt32
File descriptor.

*buf*
Type: SystemByte
Destination buffer.

*bufsize*
Type: SystemInt32
Destination buffer size.

*count*

Type: SystemInt32
Number of bytes actually received.

*error*

Type: SystemInt32
Error code. Zero upon success or an `errno` value upon failure.

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioHIDRAW_send Method

Send a 64-byte report to a Linux HID.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static void HIDRAW_send(
        int fd,
        byte[] buf,
        int bufsize,
        out int count,
        out int error
)
```

### Parameters

*fd*
Type: SystemInt32
File descriptor.

*buf*
Type: SystemByte
Source buffer.

*bufsize*
Type: SystemInt32
Source buffer size.

*count*
Type: SystemInt32
Number of bytes actually sent.

*error*

Type: SystemInt32

Error code. Zero upon success or an `errno` value upon failure.

## See Also

### Reference

libsimpleio Class

IO.Bindings Namespace

# libsimpleioI2C_close Method

Close a Linux I²C bus controller device.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static void I2C_close(
        int fd,
        out int error
)
```

### Parameters

*fd*
Type: SystemInt32
File descriptor.

*error*
Type: SystemInt32
Error code. Zero upon success or an `errno` value upon failure.

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioI2C_open Method

Open a Linux I²C bus controller device.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static void I2C_open(
        string devname,
        out int fd,
        out int error
)
```

### Parameters

*devname*
Type: SystemString
Device node name.

*fd*
Type: SystemInt32
File descriptor.

*error*
Type: SystemInt32
Error code. Zero upon success or an `errno` value upon failure.

## See Also

### Reference

libsimpleio Class

# IO.Bindings Namespace

# libsimpleioI2C_transaction Method

Send bytes to and/or receive bytes from an $I^2C$ slave device.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static void I2C_transaction(
        int fd,
        int slaveaddr,
        byte[] cmd,
        int cmdlen,
        byte[] resp,
        int resplen,
        out int error
)
```

### Parameters

*fd*
Type: SystemInt32
File descriptor.

*slaveaddr*
Type: SystemInt32
Slave address.

*cmd*

Type: SystemByte
Source buffer.

*cmdlen*

Type: SystemInt32
Source buffer size.

*resp*

Type: SystemByte
Response buffer.

*resplen*

Type: SystemInt32
Response buffer size.

*error*

Type: SystemInt32
Error code. Zero upon success or an `errno` value upon failure.

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioIPV4_ntoa Method

Convert an IPv4 address to a dotted notation string (*e.g.* 1.2.3.4).

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static void IPV4_ntoa(
        int host,
        StringBuilder buf,
        int bufsize,
        out int error
)
```

### Parameters

*host*
Type: SystemInt32
IPv4 host address

*buf*
Type: System.TextStringBuilder
Destination buffer.

*bufsize*
Type: SystemInt32
Destination buffer size.

*error*
Type: SystemInt32
Error code. Zero upon success or an `errno` value upon failure.

## See Also

### Reference

libsimpleio Class

IO.Bindings Namespace

# libsimpleioIPV4_resolve Method

Resolve a domain name to an IPv4 host address.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static void IPV4_resolve(
        string hostname,
        out int host,
        out int error
)
```

### Parameters

*hostname*
Type: SystemString
Host name to resolve.

*host*
Type: SystemInt32
IPv4 host address.

*error*
Type: SystemInt32
Error code. Zero upon success or an `errno` value upon failure.

## See Also

### Reference

libsimpleio Class

# IO.Bindings Namespace

# libsimpleioLINUX_closelog Method

Close the connection to the `syslog` service.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static void LINUX_closelog(
        out int error
)
```

### Parameters

*error*
Type: SystemInt32
Error code. Zero upon success or an `errno` value upon failure.

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioLINUX_command Method

Execute a shell command string.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static void LINUX_command(
        string cmd,
        out int error
)
```

### Parameters

*cmd*
Type: SystemString
Command string.

*error*
Type: SystemInt32
Error code.

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioLINUX_detach Method

Detach the process and run it in the background.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static void LINUX_detach(
        out int error
)
```

### Parameters

*error*
Type: SystemInt32
Error code. Zero upon success or an `errno` value upon failure.

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioLINUX_drop_privileges Method

Drop process privileges to those of the specified user.

## Remarks

Only a process running at superuser privilege is allowed to drop privileges.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static void LINUX_drop_privileges(
        string username,
        out int error
)
```

### Parameters

*username*
Type: SystemString
User privileges to assume.

*error*
Type: SystemInt32
Error code. Zero upon success or an `errno` value upon failure.

## See Also

**Reference**

libsimpleio Class

IO.Bindings Namespace

# libsimpleioLINUX_errno Method

Fetch the value of `errno`.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static int LINUX_errno()
```

### Return Value

Type: Int32
Current value of `errno`.

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioLINUX_openlog Method

Open a connection to the `syslog` service.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static void LINUX_openlog(
        string id,
        int options,
        int facility,
        out int error
)
```

### Parameters

*id*
Type: SystemString
Program identifier.

*options*
Type: SystemInt32
Logging options.

*facility*
Type: SystemInt32
Logging facility identifier.

*error*

Type: SystemInt32
Error code. Zero upon success or an `errno` value upon failure.

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioLINUX_poll Method

Wait for an event on one or more files.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static void LINUX_poll(
        int numfiles,
        int[] files,
        int[] events,
        int[] results,
        int timeoutms,
        out int error
)
```

### Parameters

*numfiles*
Type: SystemInt32
Number elements in each of the following arrays.

*files*
Type: SystemInt32
File descriptors.

*events*
Type: SystemInt32
Events to wait for on each file descriptor.

*results*

Type: SystemInt32
Events that occurred on each file descriptor.

*timeoutms*

Type: SystemInt32
Milliseconds to wait for an event to occur. A value of -1 means wait forever and a value of 0 means do not wait at all.

*error*

Type: SystemInt32
Error code.

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioLINUX_strerror Method

Retrieve the error message for a particular `errno` error code.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static void LINUX_strerror(
        int error,
        StringBuilder buf,
        int bufsize
)
```

### Parameters

*error*
Type: SystemInt32
Error code.

*buf*
Type: System.TextStringBuilder
Destination buffer.

*bufsize*
Type: SystemInt32
Destination buffer size.

## See Also

**Reference**

libsimpleio Class

IO.Bindings Namespace

# libsimpleioLINUX_syslog Method

Send a message to the `syslog` service.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static void LINUX_syslog(
        int priority,
        string msg,
        out int error
)
```

### Parameters

*priority*
Type: SystemInt32
Message priority

*msg*
Type: SystemString
Message to send.

*error*
Type: SystemInt32
Error code. Zero upon success or an `errno` value upon failure.

## See Also

### Reference

libsimpleio Class

# IO.Bindings Namespace

# libsimpleioLINUX_usleep Method

Sleep for the specified number of microseconds.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static void LINUX_usleep(
        int microsecs,
        out int error
)
```

### Parameters

*microsecs*
Type: SystemInt32
Number of microseconds to sleep.

*error*
Type: SystemInt32
Error code.

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioPWM_close Method

Close a Linux PWM output device.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static void PWM_close(
        int fd,
        out int error
)
```

### Parameters

*fd*
Type: SystemInt32
File descriptor.

*error*
Type: SystemInt32
Error code. Zero upon success or an `errno` value upon failure.

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioPWM_configure Method

Configure a Linux PWM output device.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static void PWM_configure(
        int chip,
        int channel,
        int period,
        int ontime,
        int polarity,
        out int error
)
```

### Parameters

*chip*
Type: SystemInt32
Chip number.

*channel*
Type: SystemInt32
Channel number.

*period*
Type: SystemInt32
Pulse period in microseconds.

*ontime*

Type: SystemInt32

Initial on time in microseconds.

*polarity*

Type: SystemInt32

PWM output polarity (0 for active low/inverted or 1 for active high/normal).

*error*

Type: SystemInt32

Error code. Zero upon success or an `errno` value upon failure.

## Remarks

On many platforms two more more PWM outputs may share the same clock generator, so configuring different PWM pulse periods may not be possible.

## Remarks

Not all platforms support active low (inverted) PWM outputs.

## See Also

**Reference**

libsimpleio Class

IO.Bindings Namespace

# libsimpleioPWM_open Method

Open a Linux PWM output device.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static void PWM_open(
        int chip,
        int channel,
        out int fd,
        out int error
)
```

### Parameters

*chip*
Type: SystemInt32
Chip number.

*channel*
Type: SystemInt32
Channel number.

*fd*
Type: SystemInt32
File descriptor.

*error*
Type: SystemInt32
Error code. Zero upon success or an `errno` value upon failure.

## See Also

### Reference

libsimpleio Class

IO.Bindings Namespace

# libsimpleioPWM_write Method

Set a Linux PWM output device duty cycle.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static void PWM_write(
        int fd,
        int ontime,
        out int error
)
```

### Parameters

*fd*
Type: SystemInt32
File descriptor.

*ontime*
Type: SystemInt32
On time in microseconds.

*error*
Type: SystemInt32
Error code. Zero upon success or an `errno` value upon failure.

## See Also

### Reference

libsimpleio Class

# IO.Bindings Namespace

# libsimpleioSERIAL_close Method

Close a Linux serial port device.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static void SERIAL_close(
        int fd,
        out int error
)
```

### Parameters

*fd*
Type: SystemInt32
File descriptor.

*error*
Type: SystemInt32
Error code. Zero upon success or an `errno` value upon failure.

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioSERIAL_open Method

Open a Linux serial port device.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static void SERIAL_open(
        string devname,
        int baudrate,
        int parity,
        int databits,
        int stopbits,
        out int fd,
        out int error
)
```

### Parameters

*devname*
Type: SystemString
Device node name.

*baudrate*
Type: SystemInt32
Baud rate.

*parity*
Type: SystemInt32
Parity setting (0 to 2).

*databits*

Type: SystemInt32
Word size setting (5 to 8).

*stopbits*

Type: SystemInt32
Number of stop bits (1 or 2).

*fd*

Type: SystemInt32
File descriptor.

*error*

Type: SystemInt32
Error code. Zero upon success or an `errno` value upon failure.

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioSERIAL_receive Method

Receive data from a Linux serial port device.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static void SERIAL_receive(
        int fd,
        byte[] buf,
        int bufsize,
        out int count,
        out int error
)
```

### Parameters

*fd*
Type: SystemInt32
File descriptor.

*buf*
Type: SystemByte
Destination buffer.

*bufsize*
Type: SystemInt32
Destination buffer size.

*count*

Type: SystemInt32
Number of bytes actually received.

*error*

Type: SystemInt32
Error code. Zero upon success or an `errno` value upon failure.

## See Also

**Reference**

libsimpleio Class
IO.Bindings Namespace

# libsimpleioSERIAL_send Method

Send data to a Linux serial port device.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static void SERIAL_send(
        int fd,
        byte[] buf,
        int bufsize,
        out int count,
        out int error
)
```

### Parameters

*fd*
Type: SystemInt32
File descriptor.

*buf*
Type: SystemByte
Source buffer.

*bufsize*
Type: SystemInt32
Source buffer size.

*count*
Type: SystemInt32
Number of bytes actually sent.

*error*

Type: SystemInt32

Error code. Zero upon success or an `errno` value upon failure.

## See Also

### Reference

libsimpleio Class

IO.Bindings Namespace

# libsimpleioSPI_close Method

Close a Linux SPI device.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static void SPI_close(
        int fd,
        out int error
)
```

### Parameters

*fd*
Type: SystemInt32
File descriptor.

*error*
Type: SystemInt32
Error code. Zero upon success or an `errno` value upon failure.

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioSPI_open Method

Open a Linux SPI device.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static void SPI_open(
        string devname,
        int mode,
        int wordsize,
        int speed,
        out int fd,
        out int error
)
```

### Parameters

*devname*
Type: SystemString
Device node name.

*mode*
Type: SystemInt32
SPI transfer mode (0 .. 3)

*wordsize*
Type: SystemInt32
SPI transfer word size (8, 16, or 32).

*speed*

Type: SystemInt32
SPI transfer speed in Hz.

*fd*

Type: SystemInt32
File descriptor.

*error*

Type: SystemInt32
Error code. Zero upon success or an `errno` value upon failure.

## Remarks

The Linux kernel create a device nodes for each SPI slave device, of the form `/dev/spidevX.Y` where `X` is the SPI bus controller number and `Y` is the SPI slave select number.

## See Also

**Reference**

libsimpleio Class
IO.Bindings Namespace

# libsimpleioSPI_transaction Method

Send bytes to and/or receive bytes from a Linux SPI device.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static void SPI_transaction(
        int fd,
        int csfd,
        byte[] cmd,
        int cmdlen,
        int delayus,
        byte[] resp,
        int resplen,
        out int error
)
```

### Parameters

*fd*
Type: SystemInt32
File descriptor.

*csfd*
Type: SystemInt32
Chip select file descriptor.

*cmd*

Type: SystemByte
Source buffer.

*cmdlen*

Type: SystemInt32
Source buffer size.

*delayus*

Type: SystemInt32
Delay in microseconds between the write and read operations.

*resp*

Type: SystemByte
Destination buffer.

*resplen*

Type: SystemInt32
Destination buffer size.

*error*

Type: SystemInt32
Error code. Zero upon success or an `errno` value upon failure.

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioSTREAM_decode_frame Method

Decode a frame.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static void STREAM_decode_frame(
        byte[] src,
        int srclen,
        byte[] dst,
        int dstsize,
        out int dstlen,
        out int error
)
```

### Parameters

*src*
Type: SystemByte
Source buffer.

*srclen*
Type: SystemInt32
Source buffer size.

*dst*
Type: SystemByte
Destination buffer.

*dstsize*

Type: SystemInt32

Destination buffer size.

*dstlen*

Type: SystemInt32

Size of decoded frame.

*error*

Type: SystemInt32

Error code. Zero upon success or an `errno` value upon failure.

## See Also

### Reference

libsimpleio Class

IO.Bindings Namespace

# libsimpleioSTREAM_encode_frame Method

Encode a frame.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static void STREAM_encode_frame(
        byte[] src,
        int srclen,
        byte[] dst,
        int dstsize,
        out int dstlen,
        out int error
)
```

### Parameters

*src*
Type: SystemByte
Source buffer.

*srclen*
Type: SystemInt32
Source buffer size.

*dst*
Type: SystemByte
Destination buffer.

*dstsize*

Type: SystemInt32

Destination buffer size.

*dstlen*

Type: SystemInt32

Size of encoded frame.

*error*

Type: SystemInt32

Error code. Zero upon success or an `errno` value upon failure.

## See Also

### Reference

libsimpleio Class

IO.Bindings Namespace

# libsimpleioSTREAM_receive_frame Method

Receive an encoded frame.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static void STREAM_receive_frame(
        int fd,
        byte[] buf,
        int bufsize,
        out int count,
        out int error
)
```

### Parameters

*fd*
Type: SystemInt32
File descriptior.

*buf*
Type: SystemByte
Destination buffer.

*bufsize*
Type: SystemInt32
Destination buffer size.

*count*

Type: SystemInt32
Number of bytes actually received.

*error*

Type: SystemInt32
Error code. Zero upon success or an `errno` value upon failure.

## See Also

**Reference**

libsimpleio Class
IO.Bindings Namespace

# libsimpleioSTREAM_send_frame Method

Send an encoded frame.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static void STREAM_send_frame(
        int fd,
        byte[] buf,
        int bufsize,
        out int count,
        out int error
)
```

### Parameters

*fd*
Type: SystemInt32
File descriptor.

*buf*
Type: SystemByte
Source buffer.

*bufsize*
Type: SystemInt32
Source buffer size.

*count*

Type: SystemInt32
Number of bytes actually sent.

*error*

Type: SystemInt32
Error code. Zero upon success or an `errno` value upon failure.

## See Also

**Reference**

libsimpleio Class
IO.Bindings Namespace

# libsimpleioTCP4_accept Method

Start TCP server and wait for a single connection.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static void TCP4_accept(
        int host,
        int port,
        out int fd,
        out int error
)
```

### Parameters

*host*
Type: SystemInt32
IPv4 address, of the interface to listen on. Use 0.0.0.0 to listen on all interfaces.

*port*
Type: SystemInt32
TCP port number.

*fd*
Type: SystemInt32
File descriptor.

*error*

Type: SystemInt32
Error code. Zero upon success or an `errno` value upon failure.

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioTCP4_close Method

Close a TCP connection.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static void TCP4_close(
        int fd,
        out int error
)
```

### Parameters

*fd*
Type: SystemInt32
File descriptor.

*error*
Type: SystemInt32
Error code. Zero upon success or an `errno` value upon failure.

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioTCP4_connect Method

Connect to a TCP server.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static void TCP4_connect(
        int host,
        int port,
        out int fd,
        out int error
)
```

### Parameters

*host*
Type: SystemInt32
IPv4 host address.

*port*
Type: SystemInt32
TCP port number.

*fd*
Type: SystemInt32
File descriptor.

*error*
Type: SystemInt32
Error code. Zero upon success or an `errno` value upon failure.

## See Also

### Reference

libsimpleio Class

IO.Bindings Namespace

# libsimpleioTCP4_receive Method

Receive bytes from TCP peer.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static void TCP4_receive(
        int fd,
        byte[] buf,
        int bufsize,
        out int count,
        out int error
)
```

### Parameters

*fd*
Type: SystemInt32
File descriptor.

*buf*
Type: SystemByte
Destination buffer.

*bufsize*
Type: SystemInt32
Destination buffer size.

*count*
Type: SystemInt32
Number of bytes actually received.

*error*

Type: SystemInt32

Error code. Zero upon success or an `errno` value upon failure.

## See Also

### Reference

libsimpleio Class

IO.Bindings Namespace

# libsimpleioTCP4_send Method

Send bytes to TCP peer.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static void TCP4_send(
        int fd,
        byte[] buf,
        int bufsize,
        out int count,
        out int error
)
```

### Parameters

*fd*
Type: SystemInt32
File descriptor.

*buf*
Type: SystemByte
Source buffer.

*bufsize*
Type: SystemInt32
Source buffer size.

*count*
Type: SystemInt32
Number of bytes actually sent.

*error*

Type: SystemInt32

Error code. Zero upon success or an `errno` value upon failure.

## See Also

### Reference

libsimpleio Class

IO.Bindings Namespace

# libsimpleioTCP4_server Method

Start a TCP server and fork for each connection.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static void TCP4_server(
        int host,
        int port,
        out int fd,
        out int error
)
```

### Parameters

*host*
Type: SystemInt32
IPv4 address, of the interface to listen on. Use 0.0.0.0 to listen on all interfaces.

*port*
Type: SystemInt32
TCP port number.

*fd*
Type: SystemInt32
File descriptor.

*error*

Type: SystemInt32
Error code. Zero upon success or an `errno` value upon failure.

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioUDP4_close Method

Close a UDP socket.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static void UDP4_close(
        int fd,
        out int error
)
```

### Parameters

*fd*
Type: SystemInt32
File descriptor.

*error*
Type: SystemInt32
Error code. Zero upon success or an `errno` value upon failure.

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioUDP4_open Method

Open a UDP socket.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static void UDP4_open(
        int host,
        int port,
        out int fd,
        out int error
)
```

### Parameters

*host*
Type: SystemInt32
IPv4 host address.

*port*
Type: SystemInt32
UDP port number.

*fd*
Type: SystemInt32
File descriptor.

*error*
Type: SystemInt32
Error code. Zero upon success or an `errno` value upon failure.

## See Also

### Reference

libsimpleio Class

IO.Bindings Namespace

# libsimpleioUDP4_receive Method

Receive a UDP datagram.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static void UDP4_receive(
        int fd,
        out int host,
        out int port,
        byte[] buf,
        int bufsize,
        int flags,
        out int count,
        out int error
)
```

### Parameters

*fd*
Type: SystemInt32
File descriptor.

*host*
Type: SystemInt32
Source IPv4 host address.

*port*
Type: SystemInt32
Source UDP port number.

*buf*

Type: SystemByte

Destination buffer.

*bufsize*

Type: SystemInt32

Destination buffer size.

*flags*

Type: SystemInt32

Flags for the Linux `recvfrom()` system call.

*count*

Type: SystemInt32

Number of bytes actually received.

*error*

Type: SystemInt32

Error code. Zero upon success or an `errno` value upon failure.

## See Also

### Reference

libsimpleio Class

IO.Bindings Namespace

# libsimpleioUDP4_send Method

Send a UDP datagram.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static void UDP4_send(
        int fd,
        int host,
        int port,
        byte[] buf,
        int bufsize,
        int flags,
        out int count,
        out int error
)
```

### Parameters

*fd*
Type: SystemInt32
File descriptor.

*host*
Type: SystemInt32
Destination IPv4 host address.

*port*
Type: SystemInt32
Destination UDP port number.

*buf*

Type: SystemByte

Source buffer.

*bufsize*

Type: SystemInt32

Source buffer size.

*flags*

Type: SystemInt32

Flags for the Linux `sendto()` system call.

*count*

Type: SystemInt32

Number of bytes actually sent.

*error*

Type: SystemInt32

Error code. Zero upon success or an `errno` value upon failure.

## See Also

### Reference

libsimpleio Class

IO.Bindings Namespace

# libsimpleioWATCHDOG_close Method

Close a Linux watchdog timer device.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static void WATCHDOG_close(
        int fd,
        out int error
)
```

### Parameters

*fd*
Type: SystemInt32
File descriptor.

*error*
Type: SystemInt32
Error code. Zero upon success or an `errno` value upon failure.

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioWATCHDOG_get_timeout Method

Query a Linux watchdog timer device.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static void WATCHDOG_get_timeout(
        int fd,
        out int timeout,
        out int error
)
```

### Parameters

*fd*
Type: SystemInt32
File descriptor.

*timeout*
Type: SystemInt32
Timeout period in seconds.

*error*
Type: SystemInt32
Error code. Zero upon success or an `errno` value upon failure.

## See Also

**Reference**

libsimpleio Class

IO.Bindings Namespace

# libsimpleioWATCHDOG_kick Method

Reset the watchdog timer.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static void WATCHDOG_kick(
        int fd,
        out int error
)
```

### Parameters

*fd*
Type: SystemInt32
File descriptor.

*error*
Type: SystemInt32
Error code. Zero upon success or an `errno` value upon failure.

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioWATCHDOG_open Method

Open a Linux watchdog timer device.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static void WATCHDOG_open(
        string devname,
        out int fd,
        out int error
)
```

### Parameters

*devname*
Type: SystemString
Device node name.

*fd*
Type: SystemInt32
File descriptor.

*error*
Type: SystemInt32
Error code. Zero upon success or an `errno` value upon failure.

## See Also

**Reference**

libsimpleio Class

IO.Bindings Namespace

# libsimpleioWATCHDOG_set_timeout Method

Change the watchdog timer period.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static void WATCHDOG_set_timeout(
        int fd,
        int newtimeout,
        out int timeout,
        out int error
)
```

### Parameters

*fd*
Type: SystemInt32
File descriptor.

*newtimeout*
Type: SystemInt32
Requested timeout period in seconds.

*timeout*
Type: SystemInt32
Actual timeout period in seconds.

*error*

Type: SystemInt32
Error code. Zero upon success or an `errno` value upon failure.

## Remarks

Not all platforms allow changing the timeout period. Some platforms may not allow *increasing* the period.

## See Also

**Reference**
libsimpleio Class
IO.Bindings Namespace

# libsimpleio Fields

The libsimpleio type exposes the following members.

## Fields

| | Name | Description |
|---|---|---|
| S | GPIO_DIRECTION_INPUT | Input data direction. |
| S | GPIO_DIRECTION_OUTPUT | Out data direction. |
| S | GPIO_DRIVER_OPENDRAIN | Open drain (sink only) output driver. |
| S | GPIO_DRIVER_OPENSOURCE | Open source (source only) output driver |
| S | GPIO_DRIVER_PUSHPULL | Push-pull (source and sink) output driver. |
| S | GPIO_EDGE_BOTH | Interrupt on both edges. |
| S | GPIO_EDGE_FALLING | Interrupt on falling edge. |
| S | GPIO_EDGE_NONE | Interrupts are disabled. |
| S | GPIO_EDGE_RISING | Interrupt on rising edge. |

| | | |
|---|---|---|
| S | GPIO_EVENT_REQUEST_BOTH | Enable GPIO input interrupt on both edges. |
| S | GPIO_EVENT_REQUEST_FALLING | Enable GPIO input interrupt on falling edge. |
| S | GPIO_EVENT_REQUEST_NONE | Disable GPIO input interrupt. |
| S | GPIO_EVENT_REQUEST_RISING | Enable GPIO input interrupt on rising edge. |
| S | GPIO_LINE_INFO_ACTIVE_LOW | GPIO line is configured as active low (inverted). |
| S | GPIO_LINE_INFO_KERNEL | GPIO line is being used by the kernel. |
| S | GPIO_LINE_INFO_OPEN_DRAIN | GPIO line is configured as open drain (current sink only). |
| S | GPIO_LINE_INFO_OPEN_SOURCE | GPIO line is configured as open source (current source only). |
| S | GPIO_LINE_INFO_OUTPUT | GPIO line is configured as an output. |

| | | |
|---|---|---|
| S | GPIO_LINE_REQUEST_ACTIVE_HIGH | Select GPIO line polarity active high (normal). |
| S | GPIO_LINE_REQUEST_ACTIVE_LOW | Select GPIO line polarity active low (inverted). |
| S | GPIO_LINE_REQUEST_INPUT | Select GPIO line direction input. |
| S | GPIO_LINE_REQUEST_OPEN_DRAIN | Select GPIO line driver open drain (current sink only). |
| S | GPIO_LINE_REQUEST_OPEN_SOURCE | Select GPIO line driver open source (current source only). |
| S | GPIO_LINE_REQUEST_OUTPUT | Select GPIO line direction output. |
| S | GPIO_LINE_REQUEST_PUSH_PULL | Select GPIO line driver push-pull (current source and sink). |
| S | GPIO_POLARITY_ACTIVEHIGH | Active high (normal) polarity. |
| S | GPIO_POLARITY_ACTIVELOW | Active low (inverted) polarity. |
| S | INADDR_ANY | IPv4 address for binding to all network interfaces. |

| | | |
|---|---|---|
| S | INADDR_BROADCAST | IPv4 broadcast address. |
| S | INADDR_LOOPBACK | IPv4 address for binding to the loopback interface (aka `localhost`). |
| S | LOG_ALERT | Action must be taken immediately. |
| S | LOG_AUTH | Security/authorization messages. |
| S | LOG_AUTHPRIV | Securit/authorization messages. |
| S | LOG_CONS | Write directly to the system console if there is an error while sending to the system logger. |
| S | LOG_CRIT | Critical condition. |
| S | LOG_CRON | `cron` daemon messages. |
| S | LOG_DAEMON | System daemons. |
| S | LOG_DEBUG | Debug message. |
| S | LOG_EMERG | System is unusable. |
| S | LOG_ERR | Error condition. |

| | | |
|---|---|---|
| s | LOG_FTP | FTP daemon messages. |
| s | LOG_INFO | Informational message. |
| s | LOG_KERN | Kernel messages. |
| s | LOG_LOCAL0 | Reserved for local use. |
| s | LOG_LOCAL1 | Reserved for local use. |
| s | LOG_LOCAL2 | Reserved for local use. |
| s | LOG_LOCAL3 | Reserved for local use. |
| s | LOG_LOCAL4 | Reserved for local use. |
| s | LOG_LOCAL5 | Reserved for local use. |
| s | LOG_LOCAL6 | Reserved for local use. |
| s | LOG_LOCAL7 | Reserved for local use. |
| s | LOG_LPR | Line printer subsystem |
| s | LOG_MAIL | Mail system. |

| | | |
|---|---|---|
| S | LOG_NDELAY | Open the connection immediately. Do not wait until `syslog()` is called for the first time. |
| S | LOG_NEWS | Network news subsystem |
| S | LOG_NOTICE | Normal but significant condition. |
| S | LOG_NOWAIT | Don't wait for child processes that may have been created while logging the message. (Not applicable to `glibc`.) |
| S | LOG_ODELAY | Do not open the connection immediately. Wait until `syslog()` is called for the first time. |
| S | LOG_PERROR | Also log the message to `stderr`. |
| S | LOG_PID | Include the caller's PID (process ID) with each message. |
| S | LOG_PROGNAME | Use the program name for the identity |

| | | |
|---|---|---|
| | | string. |
| | LOG_SYSLOG | Messages generated internally by syslogd |
| | LOG_USER | Random user-level messages. |
| | LOG_UUCP | UUCP subsystem |
| | LOG_WARNING | Warning condition. |
| | MSG_DONTROUTE | Don't use a gateway to send out the packet, send to hosts only on directly connected networks. |
| | MSG_DONTWAIT | Enables nonblocking operation; if the operation would block, `EAGAIN` or `EWOULDBLOCK` is returned. |
| | MSG_MORE | The caller has more data to send. This flag informs the kernel to package all of the data sent in calls with this flag set into a single datagram which is transmitted only when a call is |

| | | |
|---|---|---|
| | | performed that does not specify this flag. |
| S | POLLERR | An error occurred. |
| S | POLLHUP | Peer closed connection. |
| S | POLLIN | There is data to read. |
| S | POLLNVAL | File descriptor is invalid. |
| S | POLLOUT | Writing is now possible. |
| S | POLLPRI | There is urgent data to read. |
| S | PWM_POLARITY_ACTIVEHIGH | Configure the PWM output as active high (normal). |
| S | PWM_POLARITY_ACTIVELOW | Configure the PWM output as active low (inverted). |
| S | SERIAL_PARITY_EVEN | Request even parity checking. |
| S | SERIAL_PARITY_NONE | Disable parity checking. |
| S | SERIAL_PARITY_ODD | Request odd parity checking. |
| S | | |

| | |
|---|---|
| SPI_AUTO_CS | Use hardware slave select. |

Top

## See Also

**Reference**

libsimpleio Class

IO.Bindings Namespace

# libsimpleioGPIO_DIRECTION_INPUT Field

Input data direction.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO_DIRECTION_INPUT = 0
```

### Field Value

Type: Int32

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioGPIO_DIRECTION_OUTPUT Field

Out data direction.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO_DIRECTION_OUTPUT = 1
```

### Field Value

Type: Int32

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioGPIO_DRIVER_OPENDRAIN Field

Open drain (sink only) output driver.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO_DRIVER_OPENDRAIN = 1
```

### Field Value

Type: Int32

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioGPIO_DRIVER_OPENSOURCE Field

Open source (source only) output driver

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO_DRIVER_OPENSOURCE = 2
```

### Field Value

Type: Int32

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioGPIO_DRIVER_PUSHPULL Field

Push-pull (source and sink) output driver.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public const int GPIO_DRIVER_PUSHPULL = 0
```

### Field Value

Type: Int32

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioGPIO_EDGE_BOTH Field

Interrupt on both edges.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public const int GPIO_EDGE_BOTH = 3
```

### Field Value

Type: Int32

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioGPIO_EDGE_FALLING Field

Interrupt on falling edge.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public const int GPIO_EDGE_FALLING = 2
```

### Field Value

Type: Int32

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioGPIO_EDGE_NONE Field

Interrupts are disabled.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public const int GPIO_EDGE_NONE = 0
```

### Field Value

Type: Int32

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioGPIO_EDGE_RISING Field

Interrupt on rising edge.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public const int GPIO_EDGE_RISING = 1
```

### Field Value

Type: Int32

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioGPIO_EVENT_REQUEST_BOTH Field

Enable GPIO input interrupt on both edges.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO_EVENT_REQUEST_BOTH = 3
```

### Field Value

Type: Int32

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioGPIO_EVENT_REQUEST_FALLING Field

Enable GPIO input interrupt on falling edge.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public const int GPIO_EVENT_REQUEST_FALLING = 2
```

### Field Value

Type: Int32

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioGPIO_EVENT_REQUEST_NONE Field

Disable GPIO input interrupt.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO_EVENT_REQUEST_NONE = 0
```

### Field Value

Type: Int32

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioGPIO_EVENT_REQUEST_RISING Field

Enable GPIO input interrupt on rising edge.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public const int GPIO_EVENT_REQUEST_RISING = 1
```

### Field Value

Type: Int32

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioGPIO_LINE_INFO_ACTIVE_LOW Field

GPIO line is configured as active low (inverted).

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO_LINE_INFO_ACTIVE_LOW = 4
```

### Field Value

Type: Int32

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioGPIO_LINE_INFO_KERNEL Field

GPIO line is being used by the kernel.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public const int GPIO_LINE_INFO_KERNEL = 1
```

### Field Value

Type: Int32

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioGPIO_LINE_INFO_OPEN_DRAIN Field

GPIO line is configured as open drain (current sink only).

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public const int GPIO_LINE_INFO_OPEN_DRAIN = 8
```

### Field Value

Type: Int32

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioGPIO_LINE_INFO_OPEN_SOURCE Field

GPIO line is configured as open source (current source only).

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO_LINE_INFO_OPEN_SOURCE = 16
```

### Field Value

Type: Int32

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioGPIO_LINE_INFO_OUTPUT Field

GPIO line is configured as an output.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO_LINE_INFO_OUTPUT = 2
```

### Field Value

Type: Int32

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioGPIO_LINE_REQUEST_ACTIVE_HIGH Field

Select GPIO line polarity active high (normal).

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO_LINE_REQUEST_ACTIVE_HIGH = 0
```

### Field Value

Type: Int32

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioGPIO_LINE_REQUEST_ACTIVE_LOW Field

Select GPIO line polarity active low (inverted).

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO_LINE_REQUEST_ACTIVE_LOW = 4
```

### Field Value

Type: Int32

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioGPIO_LINE_REQUEST_INPUT Field

Select GPIO line direction input.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO_LINE_REQUEST_INPUT = 1
```

### Field Value

Type: Int32

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioGPIO_LINE_REQUEST_OPEN_DRAIN Field

Select GPIO line driver open drain (current sink only).

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO_LINE_REQUEST_OPEN_DRAIN = 8
```

### Field Value

Type: Int32

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioGPIO_LINE_REQUEST_OPEN_SOURCE Field

Select GPIO line driver open source (current source only).

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public const int GPIO_LINE_REQUEST_OPEN_SOURCE = 16
```

### Field Value

Type: Int32

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioGPIO_LINE_REQUEST_OUTPUT Field

Select GPIO line direction output.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public const int GPIO_LINE_REQUEST_OUTPUT = 2
```

### Field Value

Type: Int32

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioGPIO_LINE_REQUEST_PUSH_PULL Field

Select GPIO line driver push-pull (current source and sink).

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO_LINE_REQUEST_PUSH_PULL = 0
```

### Field Value

Type: Int32

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioGPIO_POLARITY_ACTIVEHIGH Field

Active high (normal) polarity.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO_POLARITY_ACTIVEHIGH = 1
```

### Field Value

Type: Int32

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioGPIO_POLARITY_ACTIVELOW Field

Active low (inverted) polarity.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO_POLARITY_ACTIVELOW = 0
```

### Field Value

Type: Int32

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioINADDR_ANY Field

IPv4 address for binding to all network interfaces.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int INADDR_ANY = 0
```

### Field Value

Type: Int32

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioINADDR_BROADCAST Field

IPv4 broadcast address.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public const int INADDR_BROADCAST = -1
```

### Field Value

Type: Int32

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioINADDR_LOOPBACK Field

IPv4 address for binding to the loopback interface (aka `localhost`).

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public const int INADDR_LOOPBACK = 2130706433
```

### Field Value

Type: Int32

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioLOG_ALERT Field

Action must be taken immediately.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int LOG_ALERT = 1
```

### Field Value

Type: Int32

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioLOG_AUTH Field

Security/authorization messages.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int LOG_AUTH = 32
```

### Field Value

Type: Int32

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioLOG_AUTHPRIV Field

Securit/authorization messages.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int LOG_AUTHPRIV = 80
```

### Field Value

Type: Int32

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioLOG_CONS Field

Write directly to the system console if there is an error while sending to the system logger.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int LOG_CONS = 2
```

### Field Value

Type: Int32

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioLOG_CRIT Field

Critical condition.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int LOG_CRIT = 2
```

### Field Value

Type: Int32

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioLOG_CRON Field

`cron` daemon messages.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int LOG_CRON = 72
```

### Field Value

Type: Int32

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioLOG_DAEMON Field

System daemons.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int LOG_DAEMON = 24
```

### Field Value

Type: Int32

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioLOG_DEBUG Field

Debug message.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int LOG_DEBUG = 7
```

### Field Value

Type: Int32

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioLOG_EMERG Field

System is unusable.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int LOG_EMERG = 0
```

### Field Value

Type: Int32

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioLOG_ERR Field

Error condition.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int LOG_ERR = 3
```

### Field Value

Type: Int32

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioLOG_FTP Field

`FTP` daemon messages.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int LOG_FTP = 88
```

### Field Value

Type: Int32

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioLOG_INFO Field

Informational message.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int LOG_INFO = 6
```

### Field Value

Type: Int32

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioLOG_KERN Field

Kernel messages.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int LOG_KERN = 0
```

### Field Value

Type: Int32

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioLOG_LOCAL0 Field

Reserved for local use.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int LOG_LOCAL0 = 128
```

### Field Value

Type: Int32

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioLOG_LOCAL1 Field

Reserved for local use.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int LOG_LOCAL1 = 136
```

### Field Value

Type: Int32

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioLOG_LOCAL2 Field

Reserved for local use.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int LOG_LOCAL2 = 144
```

### Field Value

Type: Int32

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioLOG_LOCAL3 Field

Reserved for local use.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int LOG_LOCAL3 = 152
```

### Field Value

Type: Int32

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioLOG_LOCAL4 Field

Reserved for local use.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int LOG_LOCAL4 = 160
```

### Field Value

Type: Int32

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioLOG_LOCAL5 Field

Reserved for local use.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int LOG_LOCAL5 = 168
```

### Field Value

Type: Int32

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioLOG_LOCAL6 Field

Reserved for local use.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int LOG_LOCAL6 = 176
```

### Field Value

Type: Int32

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioLOG_LOCAL7 Field

Reserved for local use.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int LOG_LOCAL7 = 184
```

### Field Value

Type: Int32

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioLOG_LPR Field

Line printer subsystem

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int LOG_LPR = 48
```

### Field Value

Type: Int32

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioLOG_MAIL Field

Mail system.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int LOG_MAIL = 16
```

### Field Value

Type: Int32

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioLOG_NDELAY Field

Open the connection immediately. Do not wait until `syslog()` is called for the first time.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int LOG_NDELAY = 8
```

### Field Value

Type: Int32

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioLOG_NEWS Field

Network news subsystem

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int LOG_NEWS = 56
```

### Field Value

Type: Int32

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioLOG_NOTICE Field

Normal but significant condition.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int LOG_NOTICE = 5
```

### Field Value

Type: Int32

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioLOG_NOWAIT Field

Don't wait for child processes that may have been created while logging the message. (Not applicable to `glibc`.)

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int LOG_NOWAIT = 16
```

### Field Value

Type: Int32

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioLOG_ODELAY Field

Do not open the connection immediately. Wait until `syslog()` is called for the first time.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int LOG_ODELAY = 4
```

### Field Value

Type: Int32

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioLOG_PERROR Field

Also log the message to `stderr`.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int LOG_PERROR = 32
```

### Field Value

Type: Int32

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioLOG_PID Field

Include the caller's PID (process ID) with each message.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int LOG_PID = 1
```

### Field Value

Type: Int32

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioLOG_PROGNAME Field

Use the program name for the identity string.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const string LOG_PROGNAME = ""
```

### Field Value

Type: String

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioLOG_SYSLOG Field

Messages generated internally by syslogd

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int LOG_SYSLOG = 40
```

### Field Value

Type: Int32

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioLOG_USER Field

Random user-level messages.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int LOG_USER = 8
```

### Field Value

Type: Int32

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioLOG_UUCP Field

UUCP subsystem

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int LOG_UUCP = 64
```

### Field Value

Type: Int32

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioLOG_WARNING Field

Warning condition.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int LOG_WARNING = 4
```

### Field Value

Type: Int32

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioMSG_DONTROUTE Field

Don't use a gateway to send out the packet, send to hosts only on directly connected networks.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public const int MSG_DONTROUTE = 4
```

### Field Value

Type: Int32

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioMSG_DONTWAIT Field

Enables nonblocking operation; if the operation would block, `EAGAIN` or `EWOULDBLOCK` is returned.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int MSG_DONTWAIT = 64
```

### Field Value

Type: Int32

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioMSG_MORE Field

The caller has more data to send. This flag informs the kernel to package all of the data sent in calls with this flag set into a single datagram which is transmitted only when a call is performed that does not specify this flag.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public const int MSG_MORE = 32768
```

### Field Value

Type: Int32

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioPOLLERR Field

An error occurred.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int POLLERR = 8
```

### Field Value

Type: Int32

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioPOLLHUP Field

Peer closed connection.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int POLLHUP = 16
```

### Field Value

Type: Int32

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioPOLLIN Field

There is data to read.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int POLLIN = 1
```

### Field Value

Type: Int32

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioPOLLNVAL Field

File descriptor is invalid.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public const int POLLNVAL = 32
```

### Field Value

Type: Int32

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioPOLLOUT Field

Writing is now possible.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public const int POLLOUT = 4
```

### Field Value

Type: Int32

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioPOLLPRI Field

There is urgent data to read.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int POLLPRI = 2
```

### Field Value

Type: Int32

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioPWM_POLARITY_ACTIVEHIGH Field

Configure the PWM output as active high (normal).

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public const int PWM_POLARITY_ACTIVEHIGH = 1
```

### Field Value

Type: Int32

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioPWM_POLARITY_ACTIVELOW Field

Configure the PWM output as active low (inverted).

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int PWM_POLARITY_ACTIVELOW = 0
```

### Field Value

Type: Int32

## Remarks

Not all platforms support active low (inverted) PWM outputs.

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioSERIAL_PARITY_EVEN Field

Request even parity checking.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public const int SERIAL_PARITY_EVEN = 1
```

### Field Value

Type: Int32

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioSERIAL_PARITY_NONE Field

Disable parity checking.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public const int SERIAL_PARITY_NONE = 0
```

### Field Value

Type: Int32

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioSERIAL_PARITY_ODD Field

Request odd parity checking.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public const int SERIAL_PARITY_ODD = 2
```

### Field Value

Type: Int32

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# libsimpleioSPI_AUTO_CS Field

Use hardware slave select.

**Namespace:** IO.Bindings
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int SPI_AUTO_CS = -1
```

### Field Value

Type: Int32

## See Also

### Reference

libsimpleio Class
IO.Bindings Namespace

# IO.Devices.A4988 Namespace

A4988 Stepper Motor Controller Services.

## Classes

| | Class | Description |
|---|---|---|
| | Device | Encapsulates the A4988 Stepper Motor Controller. |

# Device Class

Encapsulates the A4988 Stepper Motor Controller.

## Inheritance Hierarchy

SystemObject IO.Devices.A4988Device

**Namespace:** IO.Devices.A4988
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public class Device : Output
```

The Device type exposes the following members.

## Constructors

| | Name | Description |
|---|---|---|
| | Device | Constructor for a single A4988 device. |

Top

## Properties

| | Name | Description |
|---|---|---|
| | StepsPerRotation | Read-only property returning the number of steps a stepper |

motor has.

Top

## Methods

| | Name | Description |
|---|---|---|
| | Disable | Disable the A4988 device. |
| | Enable | Enable the A4988 device. |
| | Move | Move the stepper motor a specified number of steps at a specified rate. |
| | Reset | Reset the A4988 device. |
| | Sleep | Put the A4988 device to sleep. |
| | Spin | Spin (i.e. continuous rotation) the stepper moter at a specified rate. |
| | Wakeup | Wake up the A4988 device. |

Top

## See Also

**Reference**

IO.Devices.A4988 Namespace

# Device Constructor

Constructor for a single A4988 device.

**Namespace:** IO.Devices.A4988
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public Device(
        int StepsPerRotation,
        Pin Step,
        Pin Dir,
        Pin Enable = null,
        Pin Reset = null,
        Pin Sleep = null
)
```

### Parameters

*StepsPerRotation*
Type: SystemInt32
The number of steps per rotation. This is a physical characteristic of the particular stepper motor being driven.

*Step*
Type: IO.Interfaces.GPIOPin
GPIO pin object for the `STEP` signal.

*Dir*
Type: IO.Interfaces.GPIOPin
GPIO pin object for the `DIR` signal.

*Enable* **(Optional)**

Type: IO.Interfaces.GPIOPin
GPIO pin object for the `-ENABLE` signal.

*Reset* **(Optional)**
Type: IO.Interfaces.GPIOPin
GPIO pin object for the `-RESET` signal.

*Sleep* **(Optional)**
Type: IO.Interfaces.GPIOPin
GPIO pin object for the `-SLEEP` signal.

## See Also

### Reference

Device Class
IO.Devices.A4988 Namespace

# Device Properties

The Device type exposes the following members.

## Properties

| | Name | Description |
|---|---|---|
| | StepsPerRotation | Read-only property returning the number of steps a stepper motor has. |

Top

## See Also

**Reference**

Device Class

IO.Devices.A4988 Namespace

# DeviceStepsPerRotation Property

Read-only property returning the number of steps a stepper motor has.

**Namespace:** IO.Devices.A4988
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public int StepsPerRotation { get; }
```

### Property Value

Type: Int32

### Implements

OutputStepsPerRotation

## See Also

### Reference

Device Class
IO.Devices.A4988 Namespace

# Device Methods

The Device type exposes the following members.

## Methods

| | Name | Description |
|---|---|---|
| | Disable | Disable the A4988 device. |
| | Enable | Enable the A4988 device. |
| | Move | Move the stepper motor a specified number of steps at a specified rate. |
| | Reset | Reset the A4988 device. |
| | Sleep | Put the A4988 device to sleep. |
| | Spin | Spin (i.e. continuous rotation) the stepper moter at a specified rate. |
| | Wakeup | Wake up the A4988 device. |

Top

## See Also

**Reference**

Device Class

IO.Devices.A4988 Namespace

# DeviceDisable Method

Disable the A4988 device.

**Namespace:** IO.Devices.A4988
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public void Disable()
```

## See Also

### Reference

Device Class
IO.Devices.A4988 Namespace

# DeviceEnable Method

Enable the A4988 device.

**Namespace:** IO.Devices.A4988
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public void Enable()
```

## See Also

### Reference

Device Class
IO.Devices.A4988 Namespace

# DeviceMove Method

Move the stepper motor a specified number of steps at a specified rate.

**Namespace:** IO.Devices.A4988
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public void Move(
        int steps,
        float rate
)
```

### Parameters

*steps*
Type: SystemInt32
Number of steps to move. Negative values indicate reverse motion and positive values indicate forward motion. (The directions are nominal and depend on how the stepper motor coils are wired). Zero indicates the stepper motor should be stopped.

*rate*
Type: SystemSingle
The rate of motion, in steps per second. Negative values indicate reverse motion and positive values indicate forward motion. (The directions are nominal and depend on how the stepper motor coils are wired). Zero indicates the stepper motor should be stopped.

**Implements**
OutputMove(Int32, Single)

## Remarks

This implementation supports a maximum rate of 500 steps per second.

## See Also

**Reference**
Device Class
IO.Devices.A4988 Namespace

# DeviceReset Method

Reset the A4988 device.

**Namespace:** IO.Devices.A4988
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public void Reset()
```

## See Also

### Reference

Device Class
IO.Devices.A4988 Namespace

# DeviceSleep Method

Put the A4988 device to sleep.

**Namespace:** IO.Devices.A4988
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public void Sleep()
```

## See Also

### Reference

Device Class
IO.Devices.A4988 Namespace

# DeviceSpin Method

Spin (i.e. continuous rotation) the stepper moter at a specified rate.

**Namespace:** IO.Devices.A4988
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public void Spin(
        float rate
)
```

### Parameters

*rate*
Type: SystemSingle
The rate of motion, in steps per second. Negative values indicate reverse motion and positive values indicate forward motion. (The directions are nominal and depend on how the stepper motor coils are wired). Zero indicates the stepper motor should be stopped.

### Implements

OutputSpin(Single)

## Remarks

The A4988 stepper motor driver does not support continuous rotation and this method will always throw an exception.

## See Also

### Reference

Device Class

IO.Devices.A4988 Namespace

# DeviceWakeup Method

Wake up the A4988 device.

**Namespace:** IO.Devices.A4988
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public void Wakeup()
```

## See Also

### Reference

Device Class
IO.Devices.A4988 Namespace

# IO.Devices.AD5593R Namespace

AD5593 Analog/Digital I/O Device Services.

## Classes

| | Class | Description |
|---|---|---|
| | Device | Enscapsulates the AD5593R $I^2C$ Analog/Digital I/O device. |

## Enumerations

| | Enumeration | Description |
|---|---|---|
| | PinMode | AD5593R I/O Pin Modes. |
| | ReferenceMode | ADC5593R ADC and DAC reference settings. |

# Device Class

Enscapsulates the AD5593R I$^2$C Analog/Digital I/O device.

## Inheritance Hierarchy

SystemObject IO.Devices.AD5593RDevice

**Namespace:** IO.Devices.AD5593R
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public class Device
```

The Device type exposes the following members.

## Constructors

| | Name | Description |
|---|---|---|
| | Device | Constructor for a single AD5593R device. |

Top

## Properties

| | Name | Description |
|---|---|---|
| | ADC_Reference | Write-only property for setting the AD5593R ADC reference |

| | | |
|---|---|---|
| | | mode. |
| | DAC_Reference | Write-only property for setting the AD5593R DAC reference mode. |
| | GPIO_Inputs | GPIO input register state. Any I/O pin that is not configured as a GPIO input will read as zero. |
| | GPIO_Outputs | GPIO output register state. Any I/O pin that is not configured as a GPIO output will be written as zero. |

Top

# Methods

| | Name | Description |
|---|---|---|
| | ADC_Create | Create an AD5593R ADC input object. |
| | ConfigureChannel | Configure a single ADC5593R I/O pin. |
| | DAC_Create | Create an AD5593R DAC output object. |
| | GPIO_Create | Create an AD5593R GPIO pin object. |
| | Read_ADC | Read from an ADC channel. |
| | Write_DAC | Write to a DAC channel. |

Top

## Fields

| | Name | Description |
|---|---|---|
| S | ADC_Resolution | ADC resolution in bits. |
| S | DAC_Resolution | DAC resolution in bits. |
| S | MaxChannel | Maximum I/O channel number. |
| S | MinChannel | Minimum I/O channel number. |

Top

## See Also

### Reference

IO.Devices.AD5593R Namespace

# Device Constructor

Constructor for a single AD5593R device.

**Namespace:** IO.Devices.AD5593R
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public Device(
        Bus bus,
        int addr
)
```

### Parameters

*bus*
Type: IO.Interfaces.I2CBus
$I^2C$ bus controller object.

*addr*
Type: SystemInt32
$I^2C$ slave address.

## See Also

### Reference

Device Class
IO.Devices.AD5593R Namespace

# Device Properties

The Device type exposes the following members.

## Properties

| | Name | Description |
|---|---|---|
| | ADC_Reference | Write-only property for setting the AD5593R ADC reference mode. |
| | DAC_Reference | Write-only property for setting the AD5593R DAC reference mode. |
| | GPIO_Inputs | GPIO input register state. Any I/O pin that is not configured as a GPIO input will read as zero. |
| | GPIO_Outputs | GPIO output register state. Any I/O pin that is not configured as a GPIO output will be written as zero. |

Top

## See Also

**Reference**

Device Class

# IO.Devices.AD5593R Namespace

# DeviceADC_Reference Property

Write-only property for setting the AD5593R ADC reference mode.

**Namespace:** IO.Devices.AD5593R
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public ReferenceMode ADC_Reference { set; }
```

### Property Value

Type: ReferenceMode

## See Also

### Reference

Device Class
IO.Devices.AD5593R Namespace

# DeviceDAC_Reference Property

Write-only property for setting the AD5593R DAC reference mode.

**Namespace:** IO.Devices.AD5593R
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public ReferenceMode DAC_Reference { set; }
```

### Property Value

Type: ReferenceMode

## See Also

### Reference

Device Class
IO.Devices.AD5593R Namespace

# DeviceGPIO_Inputs Property

GPIO input register state. Any I/O pin that is not configured as a GPIO input will read as zero.

**Namespace:** IO.Devices.AD5593R
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public byte GPIO_Inputs { get; }
```

### Property Value

Type: Byte

## See Also

### Reference

Device Class
IO.Devices.AD5593R Namespace

# DeviceGPIO_Outputs Property

GPIO output register state. Any I/O pin that is not configured as a GPIO output will be written as zero.

**Namespace:** IO.Devices.AD5593R
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public byte GPIO_Outputs { get; set; }
```

### Property Value

Type: Byte

## See Also

### Reference

Device Class
IO.Devices.AD5593R Namespace

# Device Methods

The Device type exposes the following members.

## Methods

| | Name | Description |
|---|---|---|
| | ADC_Create | Create an AD5593R ADC input object. |
| | ConfigureChannel | Configure a single ADC5593R I/O pin. |
| | DAC_Create | Create an AD5593R DAC output object. |
| | GPIO_Create | Create an AD5593R GPIO pin object. |
| | Read_ADC | Read from an ADC channel. |
| | Write_DAC | Write to a DAC channel. |

Top

## See Also

**Reference**

Device Class

IO.Devices.AD5593R Namespace

# DeviceADC_Create Method

Create an AD5593R ADC input object.

**Namespace:** IO.Devices.AD5593R
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public Sample ADC_Create(
        int channel
)
```

### Parameters

*channel*
Type: SystemInt32
AD5593R ADC channel number (0 to 7).

### Return Value

Type: Sample
ADC input object.

## See Also

### Reference

Device Class
IO.Devices.AD5593R Namespace

# DeviceConfigureChannel Method

Configure a single ADC5593R I/O pin.

**Namespace:** IO.Devices.AD5593R
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public void ConfigureChannel(
        int channel,
        PinMode mode
)
```

### Parameters

*channel*
Type: SystemInt32
ADC5593R I/O channel number (0 to 7).

*mode*
Type: IO.Devices.AD5593RPinMode
ADC5593R I/O pin mode.

## See Also

### Reference

Device Class
IO.Devices.AD5593R Namespace

# DeviceDAC_Create Method

Create an AD5593R DAC output object.

**Namespace:** IO.Devices.AD5593R
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public Sample DAC_Create(
        int channel,
        int sample = 0
)
```

### Parameters

*channel*
Type: SystemInt32
AD5593R DAC channel number (0 to 7).

*sample* **(Optional)**
Type: SystemInt32
Initial DAC output sample.

### Return Value

Type: Sample
DAC output object.

## See Also

### Reference

Device Class

# IO.Devices.AD5593R Namespace

# DeviceGPIO_Create Method

Create an AD5593R GPIO pin object.

**Namespace:** IO.Devices.AD5593R
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public Pin GPIO_Create(
        int channel,
        Direction dir,
        bool state = false
)
```

### Parameters

*channel*
Type: SystemInt32
AD5593R GPIO channel number (0 to 7).

*dir*
Type: IO.Interfaces.GPIODirection
GPIO pin data direction.

*state* **(Optional)**
Type: SystemBoolean
Initial GPIO output state.

### Return Value

Type: Pin
GPIO pin object.

## See Also

### Reference

Device Class

IO.Devices.AD5593R Namespace

# DeviceRead_ADC Method

Read from an ADC channel.

**Namespace:** IO.Devices.AD5593R
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public int Read_ADC(
        int channel
)
```

### Parameters

*channel*
Type: SystemInt32
ADC channel number (0 to 7).

### Return Value

Type: Int32
ADC input sample data (0 to 4095).

## See Also

### Reference

Device Class
IO.Devices.AD5593R Namespace

# DeviceWrite_DAC Method

Write to a DAC channel.

**Namespace:** IO.Devices.AD5593R
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public void Write_DAC(
        int channel,
        int data
)
```

### Parameters

*channel*
Type: SystemInt32
DAC channel number (0 to 7).

*data*
Type: SystemInt32
DAC output sample data (0 to 4095).

## See Also

### Reference

Device Class
IO.Devices.AD5593R Namespace

# Device Fields

The Device type exposes the following members.

## Fields

| | Name | Description |
|---|---|---|
| S | ADC_Resolution | ADC resolution in bits. |
| S | DAC_Resolution | DAC resolution in bits. |
| S | MaxChannel | Maximum I/O channel number. |
| S | MinChannel | Minimum I/O channel number. |

Top

## See Also

**Reference**

Device Class

IO.Devices.AD5593R Namespace

# DeviceADC_Resolution Field

ADC resolution in bits.

**Namespace:** IO.Devices.AD5593R
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int ADC_Resolution = 12
```

### Field Value

Type: Int32

## See Also

### Reference

Device Class
IO.Devices.AD5593R Namespace

# DeviceDAC_Resolution Field

DAC resolution in bits.

**Namespace:** IO.Devices.AD5593R
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int DAC_Resolution = 12
```

### Field Value

Type: Int32

## See Also

### Reference

Device Class
IO.Devices.AD5593R Namespace

# DeviceMaxChannel Field

Maximum I/O channel number.

**Namespace:** IO.Devices.AD5593R
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int MaxChannel = 7
```

### Field Value

Type: Int32

## See Also

### Reference

Device Class
IO.Devices.AD5593R Namespace

# DeviceMinChannel Field

Minimum I/O channel number.

**Namespace:** IO.Devices.AD5593R
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int MinChannel = 0
```

### Field Value

Type: Int32

## See Also

### Reference

Device Class
IO.Devices.AD5593R Namespace

# PinMode Enumeration

AD5593R I/O Pin Modes.

**Namespace:** IO.Devices.AD5593R
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public enum PinMode
```

## Members

| Member name | Value | Description |
|---|---|---|
| ADC_Input | 0 | Analog input. |
| DAC_Output | 1 | Analog output. |
| GPIO_Input | 2 | GPIO input. |
| GPIO_Output | 3 | GPIO output. |
| GPIO_Output_OpenDrain | 4 | GPIO open drain output. |

## See Also

### Reference

# IO.Devices.AD5593R Namespace

# ReferenceMode Enumeration

ADC5593R ADC and DAC reference settings.

**Namespace:** IO.Devices.AD5593R
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public enum ReferenceMode
```

## Members

| Member name | Value | Description |
|---|---|---|
| Internalx1 | 0 | The reference voltage is 2.5V using the internal reference. |
| Internalx2 | 1 | The reference voltage is 5.0V using the internal reference. |
| Externalx1 | 2 | The reference voltage is 1.0*Vref, using an external reference. |
| Externalx2 | 3 | The reference voltage is 2.0*Vref, using an external reference. |

## See Also

### Reference

IO.Devices.AD5593R Namespace

# IO.Devices.AD5593R.ADC Namespace

AD5593 Analog/Digital I/O Device ADC Input Services.

## Classes

| | Class | Description |
|---|---|---|
| | Sample | Encapsulates AD5593R ADC inputs. |

# Sample Class

Encapsulates AD5593R ADC inputs.

## Inheritance Hierarchy

SystemObject IO.Devices.AD5593R.ADCSample

**Namespace:** IO.Devices.AD5593R.ADC
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public class Sample : Sample
```

The Sample type exposes the following members.

## Constructors

| | Name | Description |
|---|---|---|
| | Sample | Create an AD5593R ADC input. |

Top

## Properties

| | Name | Description |
|---|---|---|
| | resolution | Read-only property returning the number of bits of resolution. |

| | | |
|---|---|---|
| | sample | Read-only property returning an integer analog sample value. |

Top

## See Also

### Reference

IO.Devices.AD5593R.ADC Namespace

# Sample Constructor

Create an AD5593R ADC input.

**Namespace:** IO.Devices.AD5593R.ADC
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public Sample(
        Device dev,
        int channel
)
```

### Parameters

*dev*
Type: IO.Devices.AD5593RDevice
AD5593R device object.

*channel*
Type: SystemInt32
AD5593R I/O channel number (0 to 7).

## See Also

### Reference

Sample Class
IO.Devices.AD5593R.ADC Namespace

# Sample Properties

The Sample type exposes the following members.

## Properties

| | Name | Description |
|---|---|---|
| | resolution | Read-only property returning the number of bits of resolution. |
| | sample | Read-only property returning an integer analog sample value. |

Top

## See Also

**Reference**

Sample Class

IO.Devices.AD5593R.ADC Namespace

# Sampleresolution Property

Read-only property returning the number of bits of resolution.

**Namespace:** IO.Devices.AD5593R.ADC
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public int resolution { get; }
```

### Property Value

Type: Int32

### Implements

Sampleresolution

## See Also

### Reference

Sample Class
IO.Devices.AD5593R.ADC Namespace

# Samplesample Property

Read-only property returning an integer analog sample value.

**Namespace:** IO.Devices.AD5593R.ADC
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public int sample { get; }
```

### Property Value

Type: Int32

### Implements

Samplesample

## See Also

### Reference

Sample Class
IO.Devices.AD5593R.ADC Namespace

# IO.Devices.AD5593R.DAC Namespace

AD5593 Analog/Digital I/O Device DAC Output Services.

## Classes

| | Class | Description |
|---|---|---|
| | Sample | Encapsulates AD5593R DAC outputs. |

# Sample Class

Encapsulates AD5593R DAC outputs.

## Inheritance Hierarchy

SystemObject IO.Devices.AD5593R.DACSample

**Namespace:** IO.Devices.AD5593R.DAC
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public class Sample : Sample
```

The Sample type exposes the following members.

## Constructors

| | Name | Description |
|---|---|---|
| | Sample | Create an AD5593R DAC output. |

Top

## Properties

| | Name | Description |
|---|---|---|
| | resolution | Read-only property returning the number of bits of resolution. |

| | | |
|---|---|---|
| | sample | Write-only property for writing an integer analog sample to a DAC output. |

Top

## See Also

### Reference

IO.Devices.AD5593R.DAC Namespace

# Sample Constructor

Create an AD5593R DAC output.

**Namespace:** IO.Devices.AD5593R.DAC
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public Sample(
        Device dev,
        int channel,
        int sample = 0
)
```

### Parameters

*dev*
Type: IO.Devices.AD5593RDevice
AD5593R device object.

*channel*
Type: SystemInt32
AD5593R I/O channel number (0 to 7).

*sample* **(Optional)**
Type: SystemInt32
Initial DAC output sample.

## See Also

### Reference

Sample Class

# IO.Devices.AD5593R.DAC Namespace

# Sample Properties

The Sample type exposes the following members.

## Properties

| | Name | Description |
|---|---|---|
| | resolution | Read-only property returning the number of bits of resolution. |
| | sample | Write-only property for writing an integer analog sample to a DAC output. |

Top

## See Also

**Reference**

Sample Class

IO.Devices.AD5593R.DAC Namespace

# Sampleresolution Property

Read-only property returning the number of bits of resolution.

**Namespace:** IO.Devices.AD5593R.DAC
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public int resolution { get; }
```

### Property Value

Type: Int32

### Implements

Sampleresolution

## See Also

### Reference

Sample Class
IO.Devices.AD5593R.DAC Namespace

# Samplesample Property

Write-only property for writing an integer analog sample to a DAC output.

**Namespace:** IO.Devices.AD5593R.DAC
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public int sample { set; }
```

### Property Value

Type: Int32

### Implements

Samplesample

## See Also

### Reference

Sample Class
IO.Devices.AD5593R.DAC Namespace

# IO.Devices.AD5593R.GPIO Namespace

AD5593 Analog/Digital I/O Device GPIO Pin Services.

## Classes

| | Class | Description |
|---|---|---|
| | Pin | Encapsulates AD5593R GPIO pins. |

# Pin Class

Encapsulates AD5593R GPIO pins.

## Inheritance Hierarchy

SystemObject IO.Devices.AD5593R.GPIOPin

**Namespace:** IO.Devices.AD5593R.GPIO
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public class Pin : Pin
```

The Pin type exposes the following members.

## Constructors

| | Name | Description |
|---|---|---|
| | Pin | Create an AD5593R GPIO pin. |

Top

## Properties

| | Name | Description |
|---|---|---|
| | state | Read/Write GPIO state property. |

Top

## See Also

### Reference

IO.Devices.AD5593R.GPIO Namespace

# Pin Constructor

Create an AD5593R GPIO pin.

**Namespace:** IO.Devices.AD5593R.GPIO
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public Pin(
        Device dev,
        int channel,
        Direction dir,
        bool state = false
)
```

### Parameters

*dev*
Type: IO.Devices.AD5593RDevice
AD5593R device object.

*channel*
Type: SystemInt32
AD5593R I/O channel number (0 to 7).

*dir*
Type: IO.Interfaces.GPIODirection
GPIO pin data direction.

*state* **(Optional)**
Type: SystemBoolean
Initial GPIO output state.

## See Also

### Reference

Pin Class

IO.Devices.AD5593R.GPIO Namespace

# Pin Properties

The Pin type exposes the following members.

## Properties

| | Name | Description |
|---|---|---|
| | state | Read/Write GPIO state property. |

Top

## See Also

### Reference

Pin Class
IO.Devices.AD5593R.GPIO Namespace

# Pinstate Property

Read/Write GPIO state property.

**Namespace:** IO.Devices.AD5593R.GPIO
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public bool state { get; set; }
```

### Property Value

Type: Boolean

### Implements

Pinstate

## See Also

### Reference

Pin Class
IO.Devices.AD5593R.GPIO Namespace

# IO.Devices.ADC121C021 Namespace

ADC121C021 $I^2C$ A/D Converter Services

## Classes

| | Class | Description |
|---|---|---|
| | Sample | Encapsulates the ADC121C021 $I^2C$ A/D converter. |

# Sample Class

Encapsulates the ADC121C021 I$^2$C A/D converter.

## Inheritance Hierarchy

SystemObject IO.Devices.ADC121C021Sample

**Namespace:** IO.Devices.ADC121C021
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public class Sample : Sample
```

The Sample type exposes the following members.

## Constructors

| | Name | Description |
|---|---|---|
| | Sample | Constructor for an ADC121C021 analog input. |

Top

## Properties

| | Name | Description |
|---|---|---|
| | resolution | Return the number of bits of A/D |

| | | |
|---|---|---|
| | | resolution. |
| | sample | Returns a single 12-bit analog sample. |

Top

## See Also

### Reference

IO.Devices.ADC121C021 Namespace

# Sample Constructor

Constructor for an ADC121C021 analog input.

**Namespace:** IO.Devices.ADC121C021
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

**C#** **VB** **F#** Copy

```
public Sample(
        Bus bus,
        byte addr
)
```

### Parameters

*bus*
Type: IO.Interfaces.I2CBus
$I^2C$ bus controller.

*addr*
Type: SystemByte
$I^2C$ slave address.

## See Also

### Reference

Sample Class
IO.Devices.ADC121C021 Namespace

# Sample Properties

The Sample type exposes the following members.

## Properties

| | Name | Description |
|---|---|---|
| | resolution | Return the number of bits of A/D resolution. |
| | sample | Returns a single 12-bit analog sample. |

Top

## See Also

**Reference**

Sample Class

IO.Devices.ADC121C021 Namespace

# Sampleresolution Property

Return the number of bits of A/D resolution.

**Namespace:** IO.Devices.ADC121C021
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public int resolution { get; }
```

### Property Value

Type: Int32

### Implements

Sampleresolution

## See Also

### Reference

Sample Class
IO.Devices.ADC121C021 Namespace

# Samplesample Property

Returns a single 12-bit analog sample.

**Namespace:** IO.Devices.ADC121C021
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public int sample { get; }
```

### Property Value

Type: Int32

### Implements

Samplesample

## See Also

### Reference

Sample Class
IO.Devices.ADC121C021 Namespace

# IO.Devices.ClickBoards.RemoteIO.ADAC Namespace

Mikroelektronika ADAC Click MIKROE-2690 Services

## Classes

| | Class | Description |
|---|---|---|
| | Board | Encapsulates the Mikroelektronika ADAC Click Board. MIKROE-2690. |

# Board Class

Encapsulates the Mikroelektronika ADAC Click Board. MIKROE-2690.

## Inheritance Hierarchy

SystemObject IO.Devices.ClickBoards.RemoteIO.ADACBoard

**Namespace:** IO.Devices.ClickBoards.RemoteIO.ADAC
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public class Board
```

The Board type exposes the following members.

## Constructors

| | Name | Description |
|---|---|---|
| | Board | Constructor for a single ADAC click. |

Top

## Properties

| | Name | Description |
|---|---|---|
| | device | Returns the underlying AD5593R device object. |

Top

## Methods

| | Name | Description |
|---|---|---|
| | ADC | Factory function for creating ADC inputs. |
| | DAC | Factory function for creating DAC outputs. |
| | GPIO | Factory function for creating GPIO pins. |
| | Reset | Issue hardware reset to the AD5593R. |

Top

## Fields

| | Name | Description |
|---|---|---|
| S | DefaultAddress | Default $I^2C$ slave address. |

Top

## See Also

**Reference**

IO.Devices.ClickBoards.RemoteIO.ADAC Namespace

# Board Constructor

Constructor for a single ADAC click.

**Namespace:** IO.Devices.ClickBoards.RemoteIO.ADAC
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public Board(
        int socknum,
        int addr = 16,
        Device remdev = null
)
```

### Parameters

*socknum*
Type: SystemInt32
mikroBUS socket number.

*addr* **(Optional)**
Type: SystemInt32
$I^2C$ slave address.

*remdev* **(Optional)**
Type: IO.Objects.RemoteIODevice
Remote I/O server device object.

## See Also

### Reference

Board Class

# IO.Devices.ClickBoards.RemoteIO.ADAC Namespace

# Board Properties

The Board type exposes the following members.

## Properties

| | Name | Description |
|---|---|---|
| | device | Returns the underlying AD5593R device object. |

Top

## See Also

**Reference**

Board Class

IO.Devices.ClickBoards.RemoteIO.ADAC Namespace

# Boarddevice Property

Returns the underlying AD5593R device object.

**Namespace:** IO.Devices.ClickBoards.RemoteIO.ADAC
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public Device device { get; }
```

### Property Value

Type: Device

## See Also

### Reference

Board Class
IO.Devices.ClickBoards.RemoteIO.ADAC Namespace

# Board Methods

The Board type exposes the following members.

## Methods

| | Name | Description |
|---|---|---|
| | ADC | Factory function for creating ADC inputs. |
| | DAC | Factory function for creating DAC outputs. |
| | GPIO | Factory function for creating GPIO pins. |
| | Reset | Issue hardware reset to the AD5593R. |

Top

## See Also

**Reference**

Board Class

IO.Devices.ClickBoards.RemoteIO.ADAC Namespace

# BoardADC Method

Factory function for creating ADC inputs.

**Namespace:** IO.Devices.ClickBoards.RemoteIO.ADAC
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public Sample ADC(
        int channel
)
```

### Parameters

*channel*
Type: SystemInt32
AD5593R I/O channel number (0 to 7).

### Return Value

Type: Sample
ADC input object.

## See Also

### Reference

Board Class
IO.Devices.ClickBoards.RemoteIO.ADAC Namespace

# BoardDAC Method

Factory function for creating DAC outputs.

**Namespace:** IO.Devices.ClickBoards.RemoteIO.ADAC
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public Sample DAC(
        int channel,
        int sample = 0
)
```

### Parameters

*channel*
Type: SystemInt32
AD5593R I/O channel number (0 to 7).

*sample* **(Optional)**
Type: SystemInt32
Initial DAC output sample.

### Return Value

Type: Sample
DAC output object.

## See Also

### Reference

Board Class

# IO.Devices.ClickBoards.RemoteIO.ADAC Namespace

# BoardGPIO Method

Factory function for creating GPIO pins.

**Namespace:** IO.Devices.ClickBoards.RemoteIO.ADAC
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public Pin GPIO(
        int channel,
        Direction dir,
        bool state = false
)
```

### Parameters

*channel*
Type: SystemInt32
AD5593R I/O channel number (0 to 7).

*dir*
Type: IO.Interfaces.GPIODirection
GPIO pin direction.

*state* **(Optional)**
Type: SystemBoolean
Initial GPIO output state.

### Return Value

Type: Pin
GPIO pin object.

## See Also

### Reference

Board Class

IO.Devices.ClickBoards.RemoteIO.ADAC Namespace

# BoardReset Method

Issue hardware reset to the AD5593R.

**Namespace:** IO.Devices.ClickBoards.RemoteIO.ADAC
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public void Reset()
```

## See Also

### Reference

Board Class
IO.Devices.ClickBoards.RemoteIO.ADAC Namespace

# Board Fields

The Board type exposes the following members.

## Fields

| | Name | Description |
|---|---|---|
| | DefaultAddress | Default $I^2C$ slave address. |

Top

## See Also

**Reference**

Board Class

IO.Devices.ClickBoards.RemoteIO.ADAC Namespace

# BoardDefaultAddress Field

Default I$^2$C slave address.

**Namespace:** IO.Devices.ClickBoards.RemoteIO.ADAC
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const byte DefaultAddress = 16
```

### Field Value

Type: Byte

## See Also

### Reference

Board Class
IO.Devices.ClickBoards.RemoteIO.ADAC Namespace

# IO.Devices.ClickBoards.RemoteIO.Expand Namespace

Mikroelektronika Expand Click MIKROE-951 Services

## Classes

| | Class | Description |
|---|---|---|
| | Board | Encapsulates the Mikroelektronika Expand Click Board. |

# Board Class

Encapsulates the Mikroelektronika Expand Click Board.

## Inheritance Hierarchy

SystemObject IO.Devices.ClickBoards.RemoteIO.ExpandBoard

**Namespace:** IO.Devices.ClickBoards.RemoteIO.Expand
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public class Board
```

The Board type exposes the following members.

## Constructors

| | Name | Description |
|---|---|---|
| | Board | Constructor for a single Expand click. |

Top

## Properties

| | Name | Description |
|---|---|---|
| | device | Returns the underlying MCP23S17 device object. |

Top

## Methods

| | Name | Description |
|---|---|---|
| | GPIO | Factory function for creating GPIO pins. |
| | Reset | Issue hardware reset to the MCP23S17. |

Top

## See Also

**Reference**

IO.Devices.ClickBoards.RemoteIO.Expand Namespace

# Board Constructor

Constructor for a single Expand click.

**Namespace:** IO.Devices.ClickBoards.RemoteIO.Expand
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public Board(
        int socknum,
        Device remdev = null
)
```

### Parameters

*socknum*
Type: SystemInt32
mikroBUS socket number.

*remdev* **(Optional)**
Type: IO.Objects.RemoteIODevice
Remote I/O server device object.

## See Also

### Reference

Board Class
IO.Devices.ClickBoards.RemoteIO.Expand Namespace

# Board Properties

The Board type exposes the following members.

## Properties

| | Name | Description |
|---|---|---|
| | device | Returns the underlying MCP23S17 device object. |

Top

## See Also

**Reference**

Board Class

IO.Devices.ClickBoards.RemoteIO.Expand Namespace

# Boarddevice Property

Returns the underlying MCP23S17 device object.

**Namespace:** IO.Devices.ClickBoards.RemoteIO.Expand
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public Device device { get; }
```

### Property Value

Type: Device

## See Also

### Reference

Board Class
IO.Devices.ClickBoards.RemoteIO.Expand Namespace

# Board Methods

The Board type exposes the following members.

## Methods

| | Name | Description |
|---|---|---|
| | GPIO | Factory function for creating GPIO pins. |
| | Reset | Issue hardware reset to the MCP23S17. |

Top

## See Also

**Reference**

Board Class

IO.Devices.ClickBoards.RemoteIO.Expand Namespace

# BoardGPIO Method

Factory function for creating GPIO pins.

**Namespace:** IO.Devices.ClickBoards.RemoteIO.Expand
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public Pin GPIO(
        int channel,
        Direction dir,
        bool state = false
)
```

### Parameters

*channel*
Type: SystemInt32
MCP23S17 channel number (0 to 15).

*dir*
Type: IO.Interfaces.GPIODirection
GPIO pin direction.

*state* **(Optional)**
Type: SystemBoolean
Initial GPIO output state.

### Return Value

Type: Pin
GPIO pin object.

## See Also

### Reference

Board Class

IO.Devices.ClickBoards.RemoteIO.Expand Namespace

# BoardReset Method

Issue hardware reset to the MCP23S17.

**Namespace:** IO.Devices.ClickBoards.RemoteIO.Expand
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public void Reset()
```

## See Also

### Reference

Board Class
IO.Devices.ClickBoards.RemoteIO.Expand Namespace

# IO.Devices.ClickBoards.RemoteIO.Expand2 Namespace

Mikroelektronika Expand 2 Click MIKROE-1838 Services

## Classes

| | Class | Description |
|---|---|---|
| | Board | Encapsulates the Mikroelektronika Expand 2 Click Board. |

# Board Class

Encapsulates the Mikroelektronika Expand 2 Click Board.

## Inheritance Hierarchy

SystemObject IO.Devices.ClickBoards.RemoteIO.Expand2Board

**Namespace:** IO.Devices.ClickBoards.RemoteIO.Expand2
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public class Board
```

The Board type exposes the following members.

## Constructors

| | Name | Description |
|---|---|---|
| | Board | Constructor for a single Expand 2 click. |

Top

## Properties

| | Name | Description |
|---|---|---|
| | device | Returns the underlying MCP23017 device object. |

Top

## Methods

| | Name | Description |
|---|---|---|
| | GPIO | Factory function for creating GPIO pins. |
| | Reset | Issue hardware reset to the MCP23017. |

Top

## Fields

| | Name | Description |
|---|---|---|
| S | DefaultAddress | Default $I^2C$ slave address. |

Top

## See Also

**Reference**

IO.Devices.ClickBoards.RemoteIO.Expand2 Namespace

# Board Constructor

Constructor for a single Expand 2 click.

**Namespace:** IO.Devices.ClickBoards.RemoteIO.Expand2
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public Board(
        int socknum,
        int addr = 32,
        Device remdev = null
)
```

### Parameters

*socknum*
Type: SystemInt32
mikroBUS socket number.

*addr* **(Optional)**
Type: SystemInt32
$I^2C$ slave address.

*remdev* **(Optional)**
Type: IO.Objects.RemoteIODevice
Remote I/O server device object.

## See Also

### Reference

Board Class

# IO.Devices.ClickBoards.RemoteIO.Expand2 Namespace

# Board Properties

The Board type exposes the following members.

## Properties

| | Name | Description |
|---|---|---|
| | device | Returns the underlying MCP23017 device object. |

Top

## See Also

**Reference**

Board Class

IO.Devices.ClickBoards.RemoteIO.Expand2 Namespace

# Boarddevice Property

Returns the underlying MCP23017 device object.

**Namespace:** IO.Devices.ClickBoards.RemoteIO.Expand2
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public Device device { get; }
```

### Property Value

Type: Device

## See Also

### Reference

Board Class
IO.Devices.ClickBoards.RemoteIO.Expand2 Namespace

# Board Methods

The Board type exposes the following members.

## Methods

| | Name | Description |
|---|---|---|
| | GPIO | Factory function for creating GPIO pins. |
| | Reset | Issue hardware reset to the MCP23017. |

Top

## See Also

**Reference**

Board Class

IO.Devices.ClickBoards.RemoteIO.Expand2 Namespace

# BoardGPIO Method

Factory function for creating GPIO pins.

**Namespace:** IO.Devices.ClickBoards.RemoteIO.Expand2
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public Pin GPIO(
        int channel,
        Direction dir,
        bool state = false
)
```

### Parameters

*channel*
Type: SystemInt32
MCP23017 channel number (0 to 15).

*dir*
Type: IO.Interfaces.GPIODirection
GPIO pin direction.

*state* **(Optional)**
Type: SystemBoolean
Initial GPIO output state.

### Return Value

Type: Pin
GPIO pin object.

## See Also

### Reference

Board Class

IO.Devices.ClickBoards.RemoteIO.Expand2 Namespace

# BoardReset Method

Issue hardware reset to the MCP23017.

**Namespace:** IO.Devices.ClickBoards.RemoteIO.Expand2
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public void Reset()
```

## See Also

### Reference

Board Class
IO.Devices.ClickBoards.RemoteIO.Expand2 Namespace

# Board Fields

The Board type exposes the following members.

## Fields

| | Name | Description |
|---|---|---|
| | DefaultAddress | Default $I^2C$ slave address. |

Top

## See Also

**Reference**

Board Class

IO.Devices.ClickBoards.RemoteIO.Expand2 Namespace

# BoardDefaultAddress Field

Default I$^2$C slave address.

**Namespace:** IO.Devices.ClickBoards.RemoteIO.Expand2
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int DefaultAddress = 32
```

### Field Value

Type: Int32

## See Also

### Reference

Board Class
IO.Devices.ClickBoards.RemoteIO.Expand2 Namespace

# IO.Devices.ClickBoards.RemoteIO.PWM Namespace

Mikroelektronika PWM Click MIKROE-1898 Services

## Classes

| | Class | Description |
|---|---|---|
| | Board | Encapsulates the Mikroelektronika PWM Click Board. MIKROE-1898. |

# Board Class

Encapsulates the Mikroelektronika PWM Click Board. MIKROE-1898.

## Inheritance Hierarchy

SystemObject IO.Devices.ClickBoards.RemoteIO.PWMBoard

**Namespace:** IO.Devices.ClickBoards.RemoteIO.PWM
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public class Board
```

The Board type exposes the following members.

## Constructors

| | Name | Description |
|---|---|---|
| | Board | Constructor for a single PWM click. |

Top

## Properties

| | Name | Description |
|---|---|---|
| | dev | Returns the underlying PCA9685 device object. |

Top

## Methods

| | Name | Description |
|---|---|---|
| | GPIO | Factory function for creating GPIO output pins. |
| | PWM | Factory function for creating PWM outputs. |
| | Servo | Factory function for creating servo outputs. |

Top

## Fields

| | Name | Description |
|---|---|---|
| S | DefaultAddress | Default $I^2C$ slave address. |

Top

## See Also

**Reference**

IO.Devices.ClickBoards.RemoteIO.PWM Namespace

# Board Constructor

Constructor for a single PWM click.

**Namespace:** IO.Devices.ClickBoards.RemoteIO.PWM
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public Board(
        int socknum,
        int freq,
        int addr = 64,
        Device remdev = null
)
```

### Parameters

*socknum*
Type: SystemInt32
mikroBUS socket number.

*freq*
Type: SystemInt32
PWM pulse frequency in Hz.

*addr* **(Optional)**
Type: SystemInt32
$I^2C$ slave address.

*remdev* **(Optional)**
Type: IO.Objects.RemoteIODevice
Remote I/O device object.

## See Also

### Reference

Board Class

IO.Devices.ClickBoards.RemoteIO.PWM Namespace

# Board Properties

The Board type exposes the following members.

## Properties

| | Name | Description |
|---|---|---|
| | dev | Returns the underlying PCA9685 device object. |

Top

## See Also

**Reference**

Board Class

IO.Devices.ClickBoards.RemoteIO.PWM Namespace

# Boarddev Property

Returns the underlying PCA9685 device object.

**Namespace:** IO.Devices.ClickBoards.RemoteIO.PWM
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public Device dev { get; }
```

### Property Value

Type: Device

## See Also

### Reference

Board Class
IO.Devices.ClickBoards.RemoteIO.PWM Namespace

# Board Methods

The Board type exposes the following members.

## Methods

| | Name | Description |
|---|---|---|
| | GPIO | Factory function for creating GPIO output pins. |
| | PWM | Factory function for creating PWM outputs. |
| | Servo | Factory function for creating servo outputs. |

Top

## See Also

**Reference**

Board Class

IO.Devices.ClickBoards.RemoteIO.PWM Namespace

# BoardGPIO Method

Factory function for creating GPIO output pins.

**Namespace:** IO.Devices.ClickBoards.RemoteIO.PWM
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public Pin GPIO(
        int channel,
        bool state = false
)
```

### Parameters

*channel*
Type: SystemInt32
PCA9685 output channel number.

*state* **(Optional)**
Type: SystemBoolean
Initial GPIO output state.

### Return Value

Type: Pin
GPIO output pin object.

## See Also

### Reference

Board Class

## IO.Devices.ClickBoards.RemoteIO.PWM Namespace

# BoardPWM Method

Factory function for creating PWM outputs.

**Namespace:** IO.Devices.ClickBoards.RemoteIO.PWM
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public Output PWM(
        int channel,
        double dutycycle = 0
)
```

### Parameters

*channel*
Type: SystemInt32
PCA9685 output channel number.

*dutycycle* **(Optional)**
Type: SystemDouble
Initial PWM output duty cycle.

### Return Value

Type: Output
PWM output object.

## See Also

### Reference

Board Class

# IO.Devices.ClickBoards.RemoteIO.PWM Namespace

# BoardServo Method

Factory function for creating servo outputs.

**Namespace:** IO.Devices.ClickBoards.RemoteIO.PWM
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public Output Servo(
        int channel,
        double position = 0
)
```

### Parameters

*channel*
Type: SystemInt32
PCA9685 output channel number.

*position* **(Optional)**
Type: SystemDouble
Initial servo position.>

### Return Value

Type: Output
Servo output object.

## See Also

### Reference

Board Class

# IO.Devices.ClickBoards.RemoteIO.PWM Namespace

# Board Fields

The Board type exposes the following members.

## Fields

| | Name | Description |
|---|---|---|
| | DefaultAddress | Default $I^2C$ slave address. |

Top

## See Also

**Reference**

Board Class

IO.Devices.ClickBoards.RemoteIO.PWM Namespace

# BoardDefaultAddress Field

Default I$^2$C slave address.

**Namespace:** IO.Devices.ClickBoards.RemoteIO.PWM
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const byte DefaultAddress = 64
```

### Field Value

Type: Byte

## See Also

### Reference

Board Class
IO.Devices.ClickBoards.RemoteIO.PWM Namespace

# IO.Devices.ClickBoards.RemoteIO.SevenSegment Namespace

Mikroelektronika 7Seg Click MIKROE-1201 Services

## Classes

| | Class | Description |
|---|---|---|
| | Board | Encapsulates the Mikroelektronika 7Seg Click Board. MIKROE-1201. |

## Enumerations

| | Enumeration | Description |
|---|---|---|
| | BoardBase | Numeral systems. |
| | BoardZeroBlanking | Zero blanking modes. |

# Board Class

Encapsulates the Mikroelektronika 7Seg Click Board. MIKROE-1201.

## Inheritance Hierarchy

SystemObject IO.Devices.ClickBoards.RemoteIO.SevenSegmentBoard

**Namespace:** IO.Devices.ClickBoards.RemoteIO.SevenSegment
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public class Board
```

The Board type exposes the following members.

## Constructors

| | Name | Description |
|---|---|---|
| | Board | Constructor for a single 7seg click. |

Top

## Properties

| | Name | Description |
|---|---|---|
| | blanking | Zero blanking mode. Allowed values are `None`, `Leading`, and `Full`. |

| | | |
|---|---|---|
| | brightness | Write-only property for setting the brightness of the display. Allowed values are 0.0 to 100.0 percent. |
| | leftdp | Write-only property for setting the left digit decimal point. |
| | radix | Numerical base or radix. Allowed values are `Decimal` and `Hexadecimal`. |
| | rightdp | Write-only property for setting the right digit decimal point. |
| | state | Write-only property for setting the state of the display. Allowed values are 0 to 99 for decimal mode and 0 to 255 for hexadecimal mode. |

Top

# Methods

| | Name | Description |
|---|---|---|
| | Clear | Clear the display. |
| | Reset | Issue hardware reset to the 74HC595 shift register chain. |

Top

# Remarks

The `MISO`*aka*`SDI` pin should be removed from the 7seg click, because it is not tri-state and will interfere with other devices on the same SPI

bus.

## See Also

### Reference

IO.Devices.ClickBoards.RemoteIO.SevenSegment Namespace

# Board Constructor

Constructor for a single 7seg click.

**Namespace:** IO.Devices.ClickBoards.RemoteIO.SevenSegment
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public Board(
        int socket,
        BoardBase radix = BoardBase.Decimal,
        BoardZeroBlanking blanking = BoardZeroBlanking
        int pwmfreq = 100,
        Device remdev = null
)
```

### Parameters

*socket*
Type: SystemInt32
mikroBUS socket number.

*radix* **(Optional)**
Type: IO.Devices.ClickBoards.RemoteIO.SevenSegmentBoardBase
Numerical base or radix. Allowed values are `Decimal` and `Hexadecimal`.

*blanking* **(Optional)**
Type: IO.Devices.ClickBoards.RemoteIO.SevenSegmentBoardZeroBlanking
Zero blanking. Allowed values are `None`, `Leading`, and `Full`.

*pwmfreq* **(Optional)**

Type: SystemInt32

PWM frequency. Set to zero to use GPIO instead of PWM.

*remdev* **(Optional)**

Type: IO.Objects.RemoteIODevice

Remote I/O server device object.

## See Also

### Reference

Board Class

IO.Devices.ClickBoards.RemoteIO.SevenSegment Namespace

# Board Properties

The Board type exposes the following members.

## Properties

| | Name | Description |
|---|---|---|
| | blanking | Zero blanking mode. Allowed values are `None`, `Leading`, and `Full`. |
| | brightness | Write-only property for setting the brightness of the display. Allowed values are 0.0 to 100.0 percent. |
| | leftdp | Write-only property for setting the left digit decimal point. |
| | radix | Numerical base or radix. Allowed values are `Decimal` and `Hexadecimal`. |
| | rightdp | Write-only property for setting the right digit decimal point. |
| | state | Write-only property for setting the state of the display. Allowed values are 0 to 99 for decimal mode and 0 to 255 for hexadecimal mode. |

Top

## See Also

### Reference

Board Class

IO.Devices.ClickBoards.RemoteIO.SevenSegment Namespace

# Boardblanking Property

Zero blanking mode. Allowed values are `None`, `Leading`, and `Full`.

**Namespace:** IO.Devices.ClickBoards.RemoteIO.SevenSegment
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public BoardZeroBlanking blanking { get; set; }
```

### Property Value

Type: BoardZeroBlanking

## See Also

### Reference

Board Class
IO.Devices.ClickBoards.RemoteIO.SevenSegment Namespace

# Boardbrightness Property

Write-only property for setting the brightness of the display. Allowed values are 0.0 to 100.0 percent.

**Namespace:** IO.Devices.ClickBoards.RemoteIO.SevenSegment
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public double brightness { set; }
```

### Property Value

Type: Double

## See Also

### Reference

Board Class
IO.Devices.ClickBoards.RemoteIO.SevenSegment Namespace

# Boardleftdp Property

Write-only property for setting the left digit decimal point.

**Namespace:** IO.Devices.ClickBoards.RemoteIO.SevenSegment
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public bool leftdp { set; }
```

### Property Value

Type: Boolean

## See Also

### Reference

Board Class
IO.Devices.ClickBoards.RemoteIO.SevenSegment Namespace

# Boardradix Property

Numerical base or radix. Allowed values are `Decimal` and `Hexadecimal`.

**Namespace:** IO.Devices.ClickBoards.RemoteIO.SevenSegment
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public BoardBase radix { get; set; }
```

### Property Value

Type: BoardBase

## See Also

### Reference

Board Class
IO.Devices.ClickBoards.RemoteIO.SevenSegment Namespace

# Boardrightdp Property

Write-only property for setting the right digit decimal point.

**Namespace:** IO.Devices.ClickBoards.RemoteIO.SevenSegment
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public bool rightdp { set; }
```

### Property Value

Type: Boolean

## See Also

### Reference

Board Class
IO.Devices.ClickBoards.RemoteIO.SevenSegment Namespace

# Boardstate Property

Write-only property for setting the state of the display. Allowed values are 0 to 99 for decimal mode and 0 to 255 for hexadecimal mode.

**Namespace:** IO.Devices.ClickBoards.RemoteIO.SevenSegment
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public int state { set; }
```

### Property Value

Type: Int32

## See Also

### Reference

Board Class
IO.Devices.ClickBoards.RemoteIO.SevenSegment Namespace

# Board Methods

The Board type exposes the following members.

## Methods

| | Name | Description |
|---|---|---|
| | Clear | Clear the display. |
| | Reset | Issue hardware reset to the 74HC595 shift register chain. |

Top

## See Also

**Reference**

Board Class

IO.Devices.ClickBoards.RemoteIO.SevenSegment Namespace

# BoardClear Method

Clear the display.

**Namespace:** IO.Devices.ClickBoards.RemoteIO.SevenSegment
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public void Clear()
```

## See Also

### Reference

Board Class
IO.Devices.ClickBoards.RemoteIO.SevenSegment Namespace

# BoardReset Method

Issue hardware reset to the 74HC595 shift register chain.

**Namespace:** IO.Devices.ClickBoards.RemoteIO.SevenSegment
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public void Reset()
```

## See Also

### Reference

Board Class
IO.Devices.ClickBoards.RemoteIO.SevenSegment Namespace

# BoardBase Enumeration

Numeral systems.

**Namespace:** IO.Devices.ClickBoards.RemoteIO.SevenSegment
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public enum Base
```

## Members

| Member name | Value | Description |
|---|---|---|
| Decimal | 0 | Base 10. |
| Hexadecimal | 1 | Base 16. |

## See Also

### Reference

IO.Devices.ClickBoards.RemoteIO.SevenSegment Namespace

# BoardZeroBlanking Enumeration

Zero blanking modes.

**Namespace:** IO.Devices.ClickBoards.RemoteIO.SevenSegment
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public enum ZeroBlanking
```

## Members

| Member name | Value | Description |
|---|---|---|
| None | 0 | No zero blanking. |
| Leading | 1 | Leading zero blanking. |
| Full | 2 | Full zero blanking. |

## See Also

### Reference

IO.Devices.ClickBoards.RemoteIO.SevenSegment Namespace

# IO.Devices.ClickBoards.SimpleIO.ADAC Namespace

Mikroelektronika ADAC Click MIKROE-2690 Services

## Classes

| | Class | Description |
|---|---|---|
| | Board | Encapsulates the Mikroelektronika ADAC Click Board. MIKROE-2690. |

# Board Class

Encapsulates the Mikroelektronika ADAC Click Board. MIKROE-2690.

## Inheritance Hierarchy

SystemObject IO.Devices.ClickBoards.SimpleIO.ADACBoard

**Namespace:** IO.Devices.ClickBoards.SimpleIO.ADAC
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public class Board
```

The Board type exposes the following members.

## Constructors

| | Name | Description |
|---|---|---|
| | Board | Constructor for a single ADAC click. |

Top

## Properties

| | Name | Description |
|---|---|---|
| | device | Returns the underlying AD5593R device object. |

Top

## Methods

| | Name | Description |
|---|---|---|
| | ADC | Factory function for creating ADC inputs. |
| | DAC | Factory function for creating DAC outputs. |
| | GPIO | Factory function for creating GPIO pins. |
| | Reset | Issue hardware reset to the AD5593R. |

Top

## Fields

| | Name | Description |
|---|---|---|
| S | DefaultAddress | Default $I^2C$ slave address. |

Top

## See Also

**Reference**

IO.Devices.ClickBoards.SimpleIO.ADAC Namespace

# Board Constructor

Constructor for a single ADAC click.

**Namespace:** IO.Devices.ClickBoards.SimpleIO.ADAC
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public Board(
        int socknum,
        int addr = 16
)
```

### Parameters

*socknum*
Type: SystemInt32
mikroBUS socket number.

*addr* **(Optional)**
Type: SystemInt32
$I^2C$ slave address.

## See Also

### Reference

Board Class
IO.Devices.ClickBoards.SimpleIO.ADAC Namespace

# Board Properties

The Board type exposes the following members.

## Properties

| | Name | Description |
|---|---|---|
| | device | Returns the underlying AD5593R device object. |

Top

## See Also

**Reference**

Board Class

IO.Devices.ClickBoards.SimpleIO.ADAC Namespace

# Boarddevice Property

Returns the underlying AD5593R device object.

**Namespace:** IO.Devices.ClickBoards.SimpleIO.ADAC
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public Device device { get; }
```

### Property Value

Type: Device

## See Also

### Reference

Board Class
IO.Devices.ClickBoards.SimpleIO.ADAC Namespace

# Board Methods

The Board type exposes the following members.

## Methods

| | Name | Description |
|---|---|---|
| | ADC | Factory function for creating ADC inputs. |
| | DAC | Factory function for creating DAC outputs. |
| | GPIO | Factory function for creating GPIO pins. |
| | Reset | Issue hardware reset to the AD5593R. |

Top

## See Also

**Reference**

Board Class

IO.Devices.ClickBoards.SimpleIO.ADAC Namespace

# BoardADC Method

Factory function for creating ADC inputs.

**Namespace:** IO.Devices.ClickBoards.SimpleIO.ADAC
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

**C#** **VB** **F#** Copy

```
public Sample ADC(
        int channel
)
```

### Parameters

*channel*
Type: SystemInt32
AD5593R I/O channel number (0 to 7).

### Return Value

Type: Sample
ADC input object.

## See Also

### Reference

Board Class
IO.Devices.ClickBoards.SimpleIO.ADAC Namespace

# BoardDAC Method

Factory function for creating DAC outputs.

**Namespace:** IO.Devices.ClickBoards.SimpleIO.ADAC
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public Sample DAC(
        int channel,
        int sample = 0
)
```

### Parameters

*channel*
Type: SystemInt32
AD5593R I/O channel number (0 to 7).

*sample* **(Optional)**
Type: SystemInt32
Initial DAC output sample.

### Return Value

Type: Sample
DAC output object.

## See Also

### Reference

Board Class

# IO.Devices.ClickBoards.SimpleIO.ADAC Namespace

# BoardGPIO Method

Factory function for creating GPIO pins.

**Namespace:** IO.Devices.ClickBoards.SimpleIO.ADAC
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public Pin GPIO(
        int channel,
        Direction dir,
        bool state = false
)
```

### Parameters

*channel*
Type: SystemInt32
AD5593R I/O channel number (0 to 7).

*dir*
Type: IO.Interfaces.GPIODirection
GPIO pin direction.

*state* **(Optional)**
Type: SystemBoolean
Initial GPIO output state.

### Return Value

Type: Pin
GPIO pin object.

## See Also

### Reference

Board Class

IO.Devices.ClickBoards.SimpleIO.ADAC Namespace

# BoardReset Method

Issue hardware reset to the AD5593R.

**Namespace:** IO.Devices.ClickBoards.SimpleIO.ADAC
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public void Reset()
```

## See Also

### Reference

Board Class
IO.Devices.ClickBoards.SimpleIO.ADAC Namespace

# Board Fields

The Board type exposes the following members.

## Fields

| | Name | Description |
|---|---|---|
| | DefaultAddress | Default $I^2C$ slave address. |

Top

## See Also

**Reference**

Board Class

IO.Devices.ClickBoards.SimpleIO.ADAC Namespace

# BoardDefaultAddress Field

Default I$^2$C slave address.

**Namespace:** IO.Devices.ClickBoards.SimpleIO.ADAC
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const byte DefaultAddress = 16
```

### Field Value

Type: Byte

## See Also

### Reference

Board Class
IO.Devices.ClickBoards.SimpleIO.ADAC Namespace

# IO.Devices.ClickBoards.SimpleIO.Expand Namespace

Mikroelektronika Expand Click MIKROE-951 Services

## Classes

| | Class | Description |
|---|---|---|
| | Board | Encapsulates the Mikroelektronika Expand Click Board. |

# Board Class

Encapsulates the Mikroelektronika Expand Click Board.

## Inheritance Hierarchy

SystemObject IO.Devices.ClickBoards.SimpleIO.ExpandBoard

**Namespace:** IO.Devices.ClickBoards.SimpleIO.Expand
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public class Board
```

The Board type exposes the following members.

## Constructors

| | Name | Description |
|---|---|---|
| | Board | Constructor for a single Expand 2 click. |

Top

## Properties

| | Name | Description |
|---|---|---|
| | device | Returns the underlying MCP23S17 device object. |

Top

## Methods

| | Name | Description |
|---|---|---|
| | GPIO | Factory function for creating GPIO pins. |
| | Reset | Issue hardware reset to the MCP23S17. |

Top

## See Also

**Reference**

IO.Devices.ClickBoards.SimpleIO.Expand Namespace

# Board Constructor

Constructor for a single Expand 2 click.

**Namespace:** IO.Devices.ClickBoards.SimpleIO.Expand
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public Board(
        int socknum
)
```

### Parameters

*socknum*
Type: SystemInt32
mikroBUS socket number.

## See Also

### Reference

Board Class
IO.Devices.ClickBoards.SimpleIO.Expand Namespace

# Board Properties

The Board type exposes the following members.

## Properties

| | Name | Description |
|---|---|---|
| | device | Returns the underlying MCP23S17 device object. |

Top

## See Also

**Reference**

Board Class

IO.Devices.ClickBoards.SimpleIO.Expand Namespace

# Boarddevice Property

Returns the underlying MCP23S17 device object.

**Namespace:** IO.Devices.ClickBoards.SimpleIO.Expand
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public Device device { get; }
```

### Property Value

Type: Device

## See Also

### Reference

Board Class
IO.Devices.ClickBoards.SimpleIO.Expand Namespace

# Board Methods

The Board type exposes the following members.

## Methods

| | Name | Description |
|---|---|---|
| | GPIO | Factory function for creating GPIO pins. |
| | Reset | Issue hardware reset to the MCP23S17. |

Top

## See Also

**Reference**

Board Class

IO.Devices.ClickBoards.SimpleIO.Expand Namespace

# BoardGPIO Method

Factory function for creating GPIO pins.

**Namespace:** IO.Devices.ClickBoards.SimpleIO.Expand
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public Pin GPIO(
        int channel,
        Direction dir,
        bool state = false
)
```

### Parameters

*channel*
Type: SystemInt32
MCP23S17 channel number (0 to 15).

*dir*
Type: IO.Interfaces.GPIODirection
GPIO pin direction.

*state* **(Optional)**
Type: SystemBoolean
Initial GPIO output state.

### Return Value

Type: Pin
GPIO pin object.

## See Also

### Reference

Board Class

IO.Devices.ClickBoards.SimpleIO.Expand Namespace

# BoardReset Method

Issue hardware reset to the MCP23S17.

**Namespace:** IO.Devices.ClickBoards.SimpleIO.Expand
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public void Reset()
```

## See Also

### Reference

Board Class
IO.Devices.ClickBoards.SimpleIO.Expand Namespace

# IO.Devices.ClickBoards.SimpleIO.Expand2 Namespace

Mikroelektronika Expand 2 Click MIKROE-1838 Services

## Classes

| | Class | Description |
|---|---|---|
| | Board | Encapsulates the Mikroelektronika Expand 2 Click Board. |

# Board Class

Encapsulates the Mikroelektronika Expand 2 Click Board.

## Inheritance Hierarchy

SystemObject IO.Devices.ClickBoards.SimpleIO.Expand2Board

**Namespace:** IO.Devices.ClickBoards.SimpleIO.Expand2
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

**C#** **VB** **F#** Copy

```
public class Board
```

The Board type exposes the following members.

## Constructors

| | Name | Description |
|---|---|---|
| | Board | Constructor for a single Expand 2 click. |

Top

## Properties

| | Name | Description |
|---|---|---|
| | device | Returns the underlying MCP23017 device object. |

Top

## Methods

| | Name | Description |
|---|---|---|
| | GPIO | Factory function for creating GPIO pins. |
| | Reset | Issue hardware reset to the MCP23017. |

Top

## Fields

| | Name | Description |
|---|---|---|
| S | DefaultAddress | Default $I^2C$ slave address. |

Top

## See Also

**Reference**

IO.Devices.ClickBoards.SimpleIO.Expand2 Namespace

# Board Constructor

Constructor for a single Expand 2 click.

**Namespace:** IO.Devices.ClickBoards.SimpleIO.Expand2
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public Board(
        int socknum,
        int addr = 32
)
```

### Parameters

*socknum*
Type: SystemInt32
mikroBUS socket number.

*addr* **(Optional)**
Type: SystemInt32
$I^2C$ slave address.

## See Also

### Reference

Board Class
IO.Devices.ClickBoards.SimpleIO.Expand2 Namespace

# Board Properties

The Board type exposes the following members.

## Properties

| | Name | Description |
|---|---|---|
| | device | Returns the underlying MCP23017 device object. |

Top

## See Also

**Reference**

Board Class

IO.Devices.ClickBoards.SimpleIO.Expand2 Namespace

# Boarddevice Property

Returns the underlying MCP23017 device object.

**Namespace:** IO.Devices.ClickBoards.SimpleIO.Expand2
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public Device device { get; }
```

### Property Value

Type: Device

## See Also

### Reference

Board Class
IO.Devices.ClickBoards.SimpleIO.Expand2 Namespace

# Board Methods

The Board type exposes the following members.

## Methods

| | Name | Description |
|---|---|---|
| | GPIO | Factory function for creating GPIO pins. |
| | Reset | Issue hardware reset to the MCP23017. |

Top

## See Also

**Reference**

Board Class

IO.Devices.ClickBoards.SimpleIO.Expand2 Namespace

# BoardGPIO Method

Factory function for creating GPIO pins.

**Namespace:** IO.Devices.ClickBoards.SimpleIO.Expand2
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public Pin GPIO(
        int channel,
        Direction dir,
        bool state = false
)
```

### Parameters

*channel*
Type: SystemInt32
MCP23017 channel number (0 to 15).

*dir*
Type: IO.Interfaces.GPIODirection
GPIO pin direction.

*state* **(Optional)**
Type: SystemBoolean
Initial GPIO output state.

### Return Value

Type: Pin
GPIO pin object.

## See Also

### Reference

Board Class

IO.Devices.ClickBoards.SimpleIO.Expand2 Namespace

# BoardReset Method

Issue hardware reset to the MCP23017.

**Namespace:** IO.Devices.ClickBoards.SimpleIO.Expand2
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public void Reset()
```

## See Also

### Reference

Board Class
IO.Devices.ClickBoards.SimpleIO.Expand2 Namespace

# Board Fields

The Board type exposes the following members.

## Fields

| | Name | Description |
|---|---|---|
| | DefaultAddress | Default $I^2C$ slave address. |

Top

## See Also

**Reference**

Board Class

IO.Devices.ClickBoards.SimpleIO.Expand2 Namespace

# BoardDefaultAddress Field

Default I²C slave address.

**Namespace:** IO.Devices.ClickBoards.SimpleIO.Expand2
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int DefaultAddress = 32
```

### Field Value

Type: Int32

## See Also

### Reference

Board Class
IO.Devices.ClickBoards.SimpleIO.Expand2 Namespace

# IO.Devices.ClickBoards.SimpleIO.PWM Namespace

Mikroelektronika PWM Click MIKROE-1898 Services

## Classes

| | Class | Description |
|---|---|---|
| | Board | Encapsulates the Mikroelektronika PWM Click Board. MIKROE-1898. |

# Board Class

Encapsulates the Mikroelektronika PWM Click Board. MIKROE-1898.

## Inheritance Hierarchy

SystemObject IO.Devices.ClickBoards.SimpleIO.PWMBoard

**Namespace:** IO.Devices.ClickBoards.SimpleIO.PWM
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public class Board
```

The Board type exposes the following members.

## Constructors

| | Name | Description |
|---|---|---|
| | Board | Constructor for a single PWM click. |

Top

## Properties

| | Name | Description |
|---|---|---|
| | dev | Returns the underlying PCA9685 device object. |

Top

## Methods

| | Name | Description |
|---|---|---|
| | GPIO | Factory function for creating GPIO output pins. |
| | PWM | Factory function for creating PWM outputs. |
| | Servo | Factory function for creating servo outputs. |

Top

## Fields

| | Name | Description |
|---|---|---|
| S | DefaultAddress | Default $I^2C$ slave address. |

Top

## See Also

**Reference**

IO.Devices.ClickBoards.SimpleIO.PWM Namespace

# Board Constructor

Constructor for a single PWM click.

**Namespace:** IO.Devices.ClickBoards.SimpleIO.PWM
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public Board(
        int socknum,
        int freq,
        int addr = 64
)
```

### Parameters

*socknum*
Type: SystemInt32
mikroBUS socket number.

*freq*
Type: SystemInt32
PWM pulse frequency in Hz.

*addr* **(Optional)**
Type: SystemInt32
$I^2C$ slave address.

## See Also

### Reference

Board Class

# IO.Devices.ClickBoards.SimpleIO.PWM Namespace

# Board Properties

The Board type exposes the following members.

## Properties

| | Name | Description |
|---|---|---|
| | dev | Returns the underlying PCA9685 device object. |

Top

## See Also

**Reference**

Board Class

IO.Devices.ClickBoards.SimpleIO.PWM Namespace

# Boarddev Property

Returns the underlying PCA9685 device object.

**Namespace:** IO.Devices.ClickBoards.SimpleIO.PWM
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public Device dev { get; }
```

### Property Value

Type: Device

## See Also

### Reference

Board Class
IO.Devices.ClickBoards.SimpleIO.PWM Namespace

# Board Methods

The Board type exposes the following members.

## Methods

| | Name | Description |
|---|---|---|
| | GPIO | Factory function for creating GPIO output pins. |
| | PWM | Factory function for creating PWM outputs. |
| | Servo | Factory function for creating servo outputs. |

Top

## See Also

**Reference**

Board Class

IO.Devices.ClickBoards.SimpleIO.PWM Namespace

# BoardGPIO Method

Factory function for creating GPIO output pins.

**Namespace:** IO.Devices.ClickBoards.SimpleIO.PWM
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public Pin GPIO(
        int channel,
        bool state = false
)
```

### Parameters

*channel*
Type: SystemInt32
PCA9685 output channel number.

*state* **(Optional)**
Type: SystemBoolean
Initial GPIO output state.

### Return Value

Type: Pin
GPIO output pin object.

## See Also

### Reference

Board Class

# IO.Devices.ClickBoards.SimpleIO.PWM Namespace

# BoardPWM Method

Factory function for creating PWM outputs.

**Namespace:** IO.Devices.ClickBoards.SimpleIO.PWM
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

**C#** **VB** **F#** Copy

```
public Output PWM(
        int channel,
        double dutycycle = 0
)
```

### Parameters

*channel*
Type: SystemInt32
PCA9685 output channel number.

*dutycycle* **(Optional)**
Type: SystemDouble
Initial PWM output duty cycle.

### Return Value

Type: Output
PWM output object.

## See Also

### Reference

Board Class

# IO.Devices.ClickBoards.SimpleIO.PWM Namespace

# BoardServo Method

Factory function for creating servo outputs.

**Namespace:** IO.Devices.ClickBoards.SimpleIO.PWM
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public Output Servo(
        int channel,
        double position = 0
)
```

### Parameters

*channel*
Type: SystemInt32
PCA9685 output channel number.

*position* **(Optional)**
Type: SystemDouble
Initial servo position.>

### Return Value

Type: Output
Servo output object.

## See Also

### Reference

Board Class

# IO.Devices.ClickBoards.SimpleIO.PWM Namespace

# Board Fields

The Board type exposes the following members.

## Fields

| | Name | Description |
|---|---|---|
| | DefaultAddress | Default $I^2C$ slave address. |

Top

## See Also

**Reference**

Board Class

IO.Devices.ClickBoards.SimpleIO.PWM Namespace

# BoardDefaultAddress Field

Default I$^2$C slave address.

**Namespace:** IO.Devices.ClickBoards.SimpleIO.PWM
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public const byte DefaultAddress = 64
```

### Field Value

Type: Byte

## See Also

### Reference

Board Class
IO.Devices.ClickBoards.SimpleIO.PWM Namespace

# IO.Devices.ClickBoards.SimpleIO.SevenSegment Namespace

Mikroelektronika 7Seg Click MIKROE-1201 Services

## Classes

| | Class | Description |
|---|---|---|
| | Board | Encapsulates the Mikroelektronika 7Seg Click Board. MIKROE-1201. |

## Enumerations

| | Enumeration | Description |
|---|---|---|
| | BoardBase | Numeral systems. |
| | BoardZeroBlanking | Zero blanking modes. |

# Board Class

Encapsulates the Mikroelektronika 7Seg Click Board. MIKROE-1201.

## Inheritance Hierarchy

SystemObject IO.Devices.ClickBoards.SimpleIO.SevenSegmentBoard

**Namespace:** IO.Devices.ClickBoards.SimpleIO.SevenSegment
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

**C#** **VB** **F#** Copy

```
public class Board
```

The Board type exposes the following members.

## Constructors

| | Name | Description |
|---|---|---|
| | Board | Constructor for a single 7seg click. |

Top

## Properties

| | Name | Description |
|---|---|---|
| | blanking | Zero blanking mode. Allowed values are `None`, `Leading`, and `Full`. |

| | | |
|---|---|---|
| | brightness | Write-only property for setting the brightness of the display. Allowed values are 0.0 to 100.0 percent. |
| | leftdp | Write-only property for setting the left digit decimal point. |
| | radix | Numerical base or radix. Allowed values are `Decimal` and `Hexadecimal`. |
| | rightdp | Write-only property for setting the right digit decimal point. |
| | state | Write-only property for setting the state of the display. Allowed values are 0 to 99 for decimal mode and 0 to 255 for hexadecimal mode. |

Top

# Methods

| | Name | Description |
|---|---|---|
| | Clear | Clear the display. |
| | Reset | Issue hardware reset to the 74HC595 shift register chain. |

Top

# Remarks

The `MISO`*aka*`SDI` pin should be removed from the 7seg click, because it is not tri-state and will interfere with other devices on the same SPI

bus.

## See Also

### Reference

IO.Devices.ClickBoards.SimpleIO.SevenSegment Namespace

# Board Constructor

Constructor for a single 7seg click.

**Namespace:** IO.Devices.ClickBoards.SimpleIO.SevenSegment
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public Board(
        int socket,
        BoardBase radix = BoardBase.Decimal,
        BoardZeroBlanking blanking = BoardZeroBlanking
        int pwmfreq = 100
)
```

### Parameters

*socket*
Type: SystemInt32
mikroBUS socket number.

*radix* **(Optional)**
Type: IO.Devices.ClickBoards.SimpleIO.SevenSegmentBoardBase
Numerical base or radix. Allowed values are `Decimal` and `Hexadecimal`.

*blanking* **(Optional)**
Type: IO.Devices.ClickBoards.SimpleIO.SevenSegmentBoardZeroBlanking
Zero blanking. Allowed values are `None`, `Leading`, and `Full`.

*pwmfreq* **(Optional)**

Type: SystemInt32
PWM frequency. Set to zero to use GPIO instead of PWM.

## See Also

### Reference

Board Class
IO.Devices.ClickBoards.SimpleIO.SevenSegment Namespace

# Board Properties

The Board type exposes the following members.

## Properties

| | Name | Description |
|---|---|---|
| | blanking | Zero blanking mode. Allowed values are `None`, `Leading`, and `Full`. |
| | brightness | Write-only property for setting the brightness of the display. Allowed values are 0.0 to 100.0 percent. |
| | leftdp | Write-only property for setting the left digit decimal point. |
| | radix | Numerical base or radix. Allowed values are `Decimal` and `Hexadecimal`. |
| | rightdp | Write-only property for setting the right digit decimal point. |
| | state | Write-only property for setting the state of the display. Allowed values are 0 to 99 for decimal mode and 0 to 255 for hexadecimal mode. |

Top

## See Also

### Reference

Board Class

IO.Devices.ClickBoards.SimpleIO.SevenSegment Namespace

# Boardblanking Property

Zero blanking mode. Allowed values are `None`, `Leading`, and `Full`.

**Namespace:** IO.Devices.ClickBoards.SimpleIO.SevenSegment
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public BoardZeroBlanking blanking { get; set; }
```

### Property Value

Type: BoardZeroBlanking

## See Also

### Reference

Board Class
IO.Devices.ClickBoards.SimpleIO.SevenSegment Namespace

# Boardbrightness Property

Write-only property for setting the brightness of the display. Allowed values are 0.0 to 100.0 percent.

**Namespace:** IO.Devices.ClickBoards.SimpleIO.SevenSegment
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public double brightness { set; }
```

### Property Value

Type: Double

## See Also

### Reference

Board Class
IO.Devices.ClickBoards.SimpleIO.SevenSegment Namespace

# Boardleftdp Property

Write-only property for setting the left digit decimal point.

**Namespace:** IO.Devices.ClickBoards.SimpleIO.SevenSegment
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public bool leftdp { set; }
```

### Property Value

Type: Boolean

## See Also

### Reference

Board Class
IO.Devices.ClickBoards.SimpleIO.SevenSegment Namespace

# Boardradix Property

Numerical base or radix. Allowed values are `Decimal` and `Hexadecimal`.

**Namespace:** IO.Devices.ClickBoards.SimpleIO.SevenSegment
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public BoardBase radix { get; set; }
```

### Property Value

Type: BoardBase

## See Also

### Reference

Board Class
IO.Devices.ClickBoards.SimpleIO.SevenSegment Namespace

# Boardrightdp Property

Write-only property for setting the right digit decimal point.

**Namespace:** IO.Devices.ClickBoards.SimpleIO.SevenSegment
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public bool rightdp { set; }
```

### Property Value

Type: Boolean

## See Also

### Reference

Board Class
IO.Devices.ClickBoards.SimpleIO.SevenSegment Namespace

# Boardstate Property

Write-only property for setting the state of the display. Allowed values are 0 to 99 for decimal mode and 0 to 255 for hexadecimal mode.

**Namespace:** IO.Devices.ClickBoards.SimpleIO.SevenSegment
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public int state { set; }
```

### Property Value

Type: Int32

## See Also

### Reference

Board Class
IO.Devices.ClickBoards.SimpleIO.SevenSegment Namespace

# Board Methods

The Board type exposes the following members.

## Methods

| | Name | Description |
|---|---|---|
| | Clear | Clear the display. |
| | Reset | Issue hardware reset to the 74HC595 shift register chain. |

Top

## See Also

**Reference**

Board Class

IO.Devices.ClickBoards.SimpleIO.SevenSegment Namespace

# BoardClear Method

Clear the display.

**Namespace:** IO.Devices.ClickBoards.SimpleIO.SevenSegment
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public void Clear()
```

## See Also

### Reference

Board Class
IO.Devices.ClickBoards.SimpleIO.SevenSegment Namespace

# BoardReset Method

Issue hardware reset to the 74HC595 shift register chain.

**Namespace:** IO.Devices.ClickBoards.SimpleIO.SevenSegment
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public void Reset()
```

## See Also

### Reference

Board Class
IO.Devices.ClickBoards.SimpleIO.SevenSegment Namespace

# BoardBase Enumeration

Numeral systems.

**Namespace:** IO.Devices.ClickBoards.SimpleIO.SevenSegment
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public enum Base
```

## Members

| Member name | Value | Description |
|---|---|---|
| Decimal | 0 | Base 10. |
| Hexadecimal | 1 | Base 16. |

## See Also

### Reference

IO.Devices.ClickBoards.SimpleIO.SevenSegment Namespace

# BoardZeroBlanking Enumeration

Zero blanking modes.

**Namespace:** IO.Devices.ClickBoards.SimpleIO.SevenSegment
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public enum ZeroBlanking
```

## Members

| Member name | Value | Description |
|---|---|---|
| None | 0 | No zero blanking. |
| Leading | 1 | Leading zero blanking. |
| Full | 2 | Full zero blanking. |

## See Also

### Reference

IO.Devices.ClickBoards.SimpleIO.SevenSegment Namespace

# IO.Devices.Grove.ADC Namespace

Seeed Studio Grove I$^2$C ADC (ADC121C021) Services

## Classes

| | Class | Description |
|---|---|---|
| | Device | Encapsulates the Seeed Studio Grove I$^2$C ADC (ADC121C021). |

# Device Class

Encapsulates the Seeed Studio Grove I$^2$C ADC (ADC121C021).

## Inheritance Hierarchy

SystemObject IO.Devices.Grove.ADCDevice

**Namespace:** IO.Devices.Grove.ADC
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public class Device
```

The Device type exposes the following members.

## Constructors

| | Name | Description |
|---|---|---|
| | Device | Constructor for a Seeed Studio Grove I$^2$C ADC (ADC121C021). |

Top

## Properties

| | Name | Description |
|---|---|---|
| | voltage | Read-only property returning an analog |

input voltage measurement.

Top

## See Also

### Reference

IO.Devices.Grove.ADC Namespace

# Device Constructor

Constructor for a Seeed Studio Grove I$^2$C ADC (ADC121C021).

**Namespace:** IO.Devices.Grove.ADC
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public Device(
        Bus bus,
        byte addr = 80
)
```

### Parameters

*bus*
Type: IO.Interfaces.I2CBus
I$^2$C bus object.

*addr* **(Optional)**
Type: SystemByte
I$^2$C device address.

## See Also

### Reference

Device Class
IO.Devices.Grove.ADC Namespace

# Device Properties

The Device type exposes the following members.

## Properties

| | Name | Description |
|---|---|---|
| | voltage | Read-only property returning an analog input voltage measurement. |

Top

## See Also

**Reference**

Device Class

IO.Devices.Grove.ADC Namespace

# Devicevoltage Property

Read-only property returning an analog input voltage measurement.

**Namespace:** IO.Devices.Grove.ADC
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public double voltage { get; }
```

### Property Value

Type: Double

## See Also

### Reference

Device Class
IO.Devices.Grove.ADC Namespace

# IO.Devices.Grove.Temperature Namespace

Seeed Studio Grove Temperature Sensor (thermistor) Services

## Classes

| | Class | Description |
|---|---|---|
| | Device | Encapsulates the Seeed Studio Grove Temperature Sensor (thermistor). |

# Device Class

Encapsulates the Seeed Studio Grove Temperature Sensor (thermistor).

## Inheritance Hierarchy

SystemObject IO.Devices.Grove.TemperatureDevice

**Namespace:** IO.Devices.Grove.Temperature
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public class Device : Sensor
```

The Device type exposes the following members.

## Constructors

| | Name | Description |
|---|---|---|
| | Device | Constructor for a Seeed Studio Grove Temperature Sensor (thermistor). |

Top

## Properties

| | Name | Description |
|---|---|---|
| | Celsius | Read-only property returning the |

| | | |
|---|---|---|
| | | temperature in degrees Celsius. |
| | Fahrenheit | Read-only property returning the temperature in degrees Fahrenheit. |
| | Kelvins | Read-only property returning the temperature in Kelvins. |

Top

## See Also

### Reference

IO.Devices.Grove.Temperature Namespace

# Device Constructor

Constructor for a Seeed Studio Grove Temperature Sensor (thermistor).

**Namespace:** IO.Devices.Grove.Temperature
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

**C#** **VB** **F#** Copy

```
public Device(
        Voltage Vin,
        double Vcc = 3.3
)
```

### Parameters

*Vin*
Type: IO.Interfaces.ADCVoltage
Voltage input object.

*Vcc* **(Optional)**
Type: SystemDouble
Reference voltage.

## See Also

### Reference

Device Class
IO.Devices.Grove.Temperature Namespace

# Device Properties

The Device type exposes the following members.

## Properties

| | Name | Description |
|---|---|---|
| | Celsius | Read-only property returning the temperature in degrees Celsius. |
| | Fahrenheit | Read-only property returning the temperature in degrees Fahrenheit. |
| | Kelvins | Read-only property returning the temperature in Kelvins. |

Top

## See Also

**Reference**

Device Class

IO.Devices.Grove.Temperature Namespace

# DeviceCelsius Property

Read-only property returning the temperature in degrees Celsius.

**Namespace:** IO.Devices.Grove.Temperature
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public double Celsius { get; }
```

### Property Value

Type: Double

### Implements

SensorCelsius

## See Also

### Reference

Device Class
IO.Devices.Grove.Temperature Namespace

# DeviceFahrenheit Property

Read-only property returning the temperature in degrees Fahrenheit.

**Namespace:** IO.Devices.Grove.Temperature
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public double Fahrenheit { get; }
```

### Property Value

Type: Double

### Implements

SensorFahrenheit

## See Also

### Reference

Device Class
IO.Devices.Grove.Temperature Namespace

# DeviceKelvins Property

Read-only property returning the temperature in Kelvins.

**Namespace:** IO.Devices.Grove.Temperature
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public double Kelvins { get; }
```

### Property Value

Type: Double

### Implements

SensorKelvins

## See Also

### Reference

Device Class
IO.Devices.Grove.Temperature Namespace

# IO.Devices.Grove.Temperature_Humidity Namespace

Seeed Studio Grove Temperature and Humdity Sensor (TH02) Services.

## Classes

| | Class | Description |
|---|---|---|
| | Device | Encapsulate the Seeed Studio Grove Temperature and Humidity Sensor (TH02). |

# Device Class

Encapsulate the Seeed Studio Grove Temperature and Humidity Sensor (TH02).

## Inheritance Hierarchy

SystemObject IO.Devices.TH02Device
IO.Devices.Grove.Temperature_HumidityDevice

**Namespace:** IO.Devices.Grove.Temperature_Humidity
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

**C#** **VB** **F#** Copy

```
public class Device : Device
```

The Device type exposes the following members.

## Constructors

| | Name | Description |
|---|---|---|
| | Device | Constructor for a Seeed Studio Grove Temperature and Humidity Sensor (TH02). |

Top

## Properties

| | Name | Description |
|---|---|---|
| | Celsius | Read-only property returning the temperature in degrees Celsius. (Inherited from Device.) |
| | DeviceID | Read-only property returning the device ID. (Inherited from Device.) |
| | Fahrenheit | Read-only property returning the temperature in degrees Fahrenheit. (Inherited from Device.) |
| | Humidity | Read-only property returning the percentage relative humidity. (Inherited from Device.) |
| | Kelvins | Read-only property returning the temperature in Kelvins. (Inherited from Device.) |

Top

# See Also

**Reference**

IO.Devices.Grove.Temperature_Humidity Namespace

# Device Constructor

Constructor for a Seeed Studio Grove Temperature and Humidity Sensor (TH02).

**Namespace:** IO.Devices.Grove.Temperature_Humidity
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

**C#** **VB** **F#** Copy

```
public Device(
        Bus bus
)
```

### Parameters

*bus*
Type: IO.Interfaces.I2CBus
$I^2$ bus object.

## See Also

### Reference

Device Class
IO.Devices.Grove.Temperature_Humidity Namespace

# Device Properties

The Device type exposes the following members.

## Properties

| | Name | Description |
|---|---|---|
| | Celsius | Read-only property returning the temperature in degrees Celsius. (Inherited from Device.) |
| | DeviceID | Read-only property returning the device ID. (Inherited from Device.) |
| | Fahrenheit | Read-only property returning the temperature in degrees Fahrenheit. (Inherited from Device.) |
| | Humidity | Read-only property returning the percentage relative humidity. (Inherited from Device.) |
| | Kelvins | Read-only property returning the temperature in Kelvins. (Inherited from Device.) |

Top

## See Also

**Reference**

Device Class

IO.Devices.Grove.Temperature_Humidity Namespace

# IO.Devices.HDC1080 Namespace

HDC1080 $I^2C$ Temperature/Humidity Sensor Services

## Classes

| | Class | Description |
|---|---|---|
| | Device | Encapsulates the HDC1080 temperature and humidity sensor. |

# Device Class

Encapsulates the HDC1080 temperature and humidity sensor.

## Inheritance Hierarchy

SystemObject IO.Devices.HDC1080Device
IO.Devices.Pmod.HYGRODevice

**Namespace:** IO.Devices.HDC1080
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public class Device : Sensor, Sensor
```

The Device type exposes the following members.

## Constructors

| | Name | Description |
|---|---|---|
| | Device | Constructor for an HDC1080 temperature and humidity sensor object. |

Top

## Properties

| | Name | Description |
|---|---|---|

| | | |
|---|---|---|
| | Celsius | Read-only property returning the temperature in degrees Celsius. |
| | DeviceID | Read-only property returning the device ID. |
| | Fahrenheit | Read-only property returning the temperature in degrees Fahrenheit. |
| | Humidity | Read-only property returning the percentage relative humidity. |
| | Kelvins | Read-only property returning the temperature in Kelvins. |
| | ManufacturerID | Read-only property returning the manufacturer ID. |

Top

# Methods

| | Name | Description |
|---|---|---|
| | Read | Read from an HDC1080 device register. |
| | Write | Write to an HDC1080 device register. |

Top

# Fields

| | Name | Description |
|---|---|---|
| S | RegConfiguration | Configuration Register |

| | | |
|---|---|---|
| | | address. |
| S | RegDeviceID | Device ID Register address. |
| S | RegHumidity | Humidity Register address. |
| S | RegManufacturerID | Manufacturer ID Register address. |
| S | RegSerialNumberFirst | Serial Number First Bits Register address. |
| S | RegSerialNumberLast | Serial Number Last Bits Register address. |
| S | RegSerialNumberMid | Serial Number Middle Bits Register address. |
| S | RegTemperature | Temperature Register address. |

Top

## See Also

**Reference**

IO.Devices.HDC1080 Namespace

# Device Constructor

Constructor for an HDC1080 temperature and humidity sensor object.

**Namespace:** IO.Devices.HDC1080
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public Device(
        Bus bus
)
```

### Parameters

*bus*
Type: IO.Interfaces.I2CBus
$I^2C$ bus controller.

## See Also

### Reference

Device Class
IO.Devices.HDC1080 Namespace

# Device Properties

The Device type exposes the following members.

## Properties

| | Name | Description |
|---|---|---|
| | Celsius | Read-only property returning the temperature in degrees Celsius. |
| | DeviceID | Read-only property returning the device ID. |
| | Fahrenheit | Read-only property returning the temperature in degrees Fahrenheit. |
| | Humidity | Read-only property returning the percentage relative humidity. |
| | Kelvins | Read-only property returning the temperature in Kelvins. |
| | ManufacturerID | Read-only property returning the manufacturer ID. |

Top

## See Also

### Reference

Device Class
IO.Devices.HDC1080 Namespace

# DeviceCelsius Property

Read-only property returning the temperature in degrees Celsius.

**Namespace:** IO.Devices.HDC1080
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public double Celsius { get; }
```

### Property Value

Type: Double

### Implements

SensorCelsius

## See Also

### Reference

Device Class
IO.Devices.HDC1080 Namespace

# DeviceDeviceID Property

Read-only property returning the device ID.

**Namespace:** IO.Devices.HDC1080
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public ushort DeviceID { get; }
```

### Property Value

Type: UInt16

## See Also

### Reference

Device Class
IO.Devices.HDC1080 Namespace

# DeviceFahrenheit Property

Read-only property returning the temperature in degrees Fahrenheit.

**Namespace:** IO.Devices.HDC1080
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public double Fahrenheit { get; }
```

### Property Value

Type: Double

### Implements

SensorFahrenheit

## See Also

### Reference

Device Class
IO.Devices.HDC1080 Namespace

# DeviceHumidity Property

Read-only property returning the percentage relative humidity.

**Namespace:** IO.Devices.HDC1080
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public double Humidity { get; }
```

### Property Value

Type: Double

### Implements

SensorHumidity

## See Also

### Reference

Device Class
IO.Devices.HDC1080 Namespace

# DeviceKelvins Property

Read-only property returning the temperature in Kelvins.

**Namespace:** IO.Devices.HDC1080
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public double Kelvins { get; }
```

### Property Value

Type: Double

### Implements

SensorKelvins

## See Also

### Reference

Device Class
IO.Devices.HDC1080 Namespace

# DeviceManufacturerID Property

Read-only property returning the manufacturer ID.

**Namespace:** IO.Devices.HDC1080
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public ushort ManufacturerID { get; }
```

### Property Value

Type: UInt16

## See Also

### Reference

Device Class
IO.Devices.HDC1080 Namespace

# Device Methods

The Device type exposes the following members.

## Methods

| | Name | Description |
|---|---|---|
| | Read | Read from an HDC1080 device register. |
| | Write | Write to an HDC1080 device register. |

Top

## See Also

**Reference**

Device Class

IO.Devices.HDC1080 Namespace

# DeviceRead Method

Read from an HDC1080 device register.

**Namespace:** IO.Devices.HDC1080
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

**C#** **VB** **F#** Copy

```
public ushort Read(
        byte reg
)
```

### Parameters

*reg*
Type: SystemByte
8-bit register address.

### Return Value

Type: UInt16
16-bit register data

## See Also

### Reference

Device Class
IO.Devices.HDC1080 Namespace

# DeviceWrite Method

Write to an HDC1080 device register.

**Namespace:** IO.Devices.HDC1080
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public void Write(
        byte reg,
        ushort data
)
```

### Parameters

*reg*
Type: SystemByte
8-bit register address.

*data*
Type: SystemUInt16
16-bit register data.

## See Also

### Reference

Device Class
IO.Devices.HDC1080 Namespace

# Device Fields

The Device type exposes the following members.

## Fields

| | Name | Description |
|---|---|---|
| | RegConfiguration | Configuration Register address. |
| | RegDeviceID | Device ID Register address. |
| | RegHumidity | Humidity Register address. |
| | RegManufacturerID | Manufacturer ID Register address. |
| | RegSerialNumberFirst | Serial Number First Bits Register address. |
| | RegSerialNumberLast | Serial Number Last Bits Register address. |
| | RegSerialNumberMid | Serial Number Middle Bits Register address. |
| | RegTemperature | Temperature Register address. |

Top

## See Also

### Reference

Device Class

IO.Devices.HDC1080 Namespace

# DeviceRegConfiguration Field

Configuration Register address.

**Namespace:** IO.Devices.HDC1080
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const byte RegConfiguration = 2
```

### Field Value

Type: Byte

## See Also

### Reference

Device Class
IO.Devices.HDC1080 Namespace

# DeviceRegDeviceID Field

Device ID Register address.

**Namespace:** IO.Devices.HDC1080
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const byte RegDeviceID = 255
```

### Field Value

Type: Byte

## See Also

### Reference

Device Class
IO.Devices.HDC1080 Namespace

# DeviceRegHumidity Field

Humidity Register address.

**Namespace:** IO.Devices.HDC1080
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const byte RegHumidity = 1
```

### Field Value

Type: Byte

## See Also

### Reference

Device Class
IO.Devices.HDC1080 Namespace

# DeviceRegManufacturerID Field

Manufacturer ID Register address.

**Namespace:** IO.Devices.HDC1080
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const byte RegManufacturerID = 254
```

### Field Value

Type: Byte

## See Also

### Reference

Device Class
IO.Devices.HDC1080 Namespace

# DeviceRegSerialNumberFirst Field

Serial Number First Bits Register address.

**Namespace:** IO.Devices.HDC1080
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const byte RegSerialNumberFirst = 251
```

### Field Value

Type: Byte

## See Also

### Reference

Device Class
IO.Devices.HDC1080 Namespace

# DeviceRegSerialNumberLast Field

Serial Number Last Bits Register address.

**Namespace:** IO.Devices.HDC1080
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const byte RegSerialNumberLast = 253
```

### Field Value

Type: Byte

## See Also

### Reference

Device Class
IO.Devices.HDC1080 Namespace

# DeviceRegSerialNumberMid Field

Serial Number Middle Bits Register address.

**Namespace:** IO.Devices.HDC1080
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const byte RegSerialNumberMid = 252
```

### Field Value

Type: Byte

## See Also

### Reference

Device Class
IO.Devices.HDC1080 Namespace

# DeviceRegTemperature Field

Temperature Register address.

**Namespace:** IO.Devices.HDC1080
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const byte RegTemperature = 0
```

### Field Value

Type: Byte

## See Also

### Reference

Device Class
IO.Devices.HDC1080 Namespace

# IO.Devices.MCP23017 Namespace

MCP23017 $I^2C$ GPIO Expander Device Services.

## Classes

| | Class | Description |
|---|---|---|
| | Device | Encapsulates the MCP23017 $I^2C$ I/O GPIO Expander. |

# Device Class

Encapsulates the MCP23017 I$^2$C I/O GPIO Expander.

## Inheritance Hierarchy

SystemObject IO.Devices.MCP23017Device

**Namespace:** IO.Devices.MCP23017
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public class Device
```

The Device type exposes the following members.

## Constructors

| | Name | Description |
|---|---|---|
| | Device | Constructor for a single MCP23017 device. |

Top

## Properties

| | Name | Description |
|---|---|---|
| | Direction | Data Direction Property (16 bits). Bits 0 |

| | | |
|---|---|---|
| | | to 7 correspond to PORT A and bits 8 to 15 correspond to PORT B. For each bit, 0=input and 1=output. |
| | DirectionA | Port A Data Direction Property (8 bits). For each bit, 0=input and 1=output. |
| | DirectionB | Port B Data Direction Property (8 bits). For each bit, 0=input and 1=output. |
| | Polarity | Data Polarity Property (16 bits). Bits 0 to 7 correspond to PORT A and bits 8 to 15 correspond to PORT B. For each bit, 0=input and 1=output. |
| | PolarityA | Port A Data Polarity Property (8 bits). For each bit, 0=input and 1=output. |
| | PolarityB | Port B Data Polarity Property (8 bits). For each bit, 0=input and 1=output. |
| | Port | Port Data Property (16 bites). Bits 0 to 7 correspond to PORT A and bits 8 to 15 correspond to PORT B. |
| | PortA | Port A Data Property (8 bits). |
| | PortB | Port B Data Property (8 bits). |
| | Pullups | Input Pullup Property (16 bits). Bits 0 to 7 correspond to PORT A and bits 8 to 15 correspond to PORT B. For each bit, 0=high impedance and 1=100k pullup. |
| | | |

| | | |
|---|---|---|
| | PullupsA | Port A Input Pullup Property (16 bits). For each bit, 0=high impedance and 1=100k pullup. |
| | PullupsB | Port B Input Pullup Property (16 bits). For each bit, 0=high impedance and 1=100k pullup. |

Top

## Methods

| | Name | Description |
|---|---|---|
| | GPIO_Create | Create an MCP23017 GPIO pin object. |

Top

## Fields

| | Name | Description |
|---|---|---|
| S | MaxChannel | Maximum I/O channel number. |
| S | MinChannel | Minimum I/O channel number. |

Top

## See Also

**Reference**

IO.Devices.MCP23017 Namespace

# Device Constructor

Constructor for a single MCP23017 device.

**Namespace:** IO.Devices.MCP23017
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

**C#** **VB** **F#** Copy

```
public Device(
        Bus bus,
        int addr
)
```

### Parameters

*bus*
Type: IO.Interfaces.I2CBus
$I^2C$ bus controller object.

*addr*
Type: SystemInt32
$I^2C$ slave address.

## See Also

### Reference

Device Class
IO.Devices.MCP23017 Namespace

# Device Properties

The Device type exposes the following members.

## Properties

| | Name | Description |
|---|---|---|
| | Direction | Data Direction Property (16 bits). Bits 0 to 7 correspond to PORT A and bits 8 to 15 correspond to PORT B. For each bit, 0=input and 1=output. |
| | DirectionA | Port A Data Direction Property (8 bits). For each bit, 0=input and 1=output. |
| | DirectionB | Port B Data Direction Property (8 bits). For each bit, 0=input and 1=output. |
| | Polarity | Data Polarity Property (16 bits). Bits 0 to 7 correspond to PORT A and bits 8 to 15 correspond to PORT B. For each bit, 0=input and 1=output. |
| | PolarityA | Port A Data Polarity Property (8 bits). For each bit, 0=input and 1=output. |
| | PolarityB | Port B Data Polarity Property (8 bits). For each bit, 0=input and 1=output. |
| | Port | Port Data Property (16 bites). Bits 0 to 7 correspond to PORT A and bits 8 to |

| | | |
|---|---|---|
| | | 15 correspond to PORT B. |
| | PortA | Port A Data Property (8 bits). |
| | PortB | Port B Data Property (8 bits). |
| | Pullups | Input Pullup Property (16 bits). Bits 0 to 7 correspond to PORT A and bits 8 to 15 correspond to PORT B. For each bit, 0=high impedance and 1=100k pullup. |
| | PullupsA | Port A Input Pullup Property (16 bits). For each bit, 0=high impedance and 1=100k pullup. |
| | PullupsB | Port B Input Pullup Property (16 bits). For each bit, 0=high impedance and 1=100k pullup. |

Top

## See Also

**Reference**

Device Class

IO.Devices.MCP23017 Namespace

# DeviceDirection Property

Data Direction Property (16 bits). Bits 0 to 7 correspond to PORT A and bits 8 to 15 correspond to PORT B. For each bit, 0=input and 1=output.

**Namespace:** IO.Devices.MCP23017
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public uint Direction { get; set; }
```

### Property Value

Type: UInt32

## Remarks

This property follows the industry standard convention for data direction bit polarity (1=output) rather than the MCP23017 `IODIR` register polarity (0=output).

## See Also

### Reference

Device Class
IO.Devices.MCP23017 Namespace

# DeviceDirectionA Property

Port A Data Direction Property (8 bits). For each bit, 0=input and 1=output.

**Namespace:** IO.Devices.MCP23017
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public byte DirectionA { get; set; }
```

### Property Value

Type: Byte

## Remarks

This property follows the industry standard convention for data direction bit polarity (1=output) rather than the MCP23017 `IODIRA` register polarity (0=output).

## See Also

### Reference

Device Class
IO.Devices.MCP23017 Namespace

# DeviceDirectionB Property

Port B Data Direction Property (8 bits). For each bit, 0=input and 1=output.

**Namespace:** IO.Devices.MCP23017
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public byte DirectionB { get; set; }
```

### Property Value

Type: Byte

## Remarks

This property follows the industry standard convention for data direction bit polarity (1=output) rather than the MCP23017 `IODIRA` register polarity (0=output).

## See Also

### Reference

Device Class
IO.Devices.MCP23017 Namespace

# DevicePolarity Property

Data Polarity Property (16 bits). Bits 0 to 7 correspond to PORT A and bits 8 to 15 correspond to PORT B. For each bit, 0=input and 1=output.

**Namespace:** IO.Devices.MCP23017
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public uint Polarity { get; set; }
```

### Property Value

Type: UInt32

## See Also

### Reference

Device Class
IO.Devices.MCP23017 Namespace

# DevicePolarityA Property

Port A Data Polarity Property (8 bits). For each bit, 0=input and 1=output.

**Namespace:** IO.Devices.MCP23017
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public byte PolarityA { get; set; }
```

### Property Value

Type: Byte

## See Also

### Reference

Device Class
IO.Devices.MCP23017 Namespace

# DevicePolarityB Property

Port B Data Polarity Property (8 bits). For each bit, 0=input and 1=output.

**Namespace:** IO.Devices.MCP23017
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public byte PolarityB { get; set; }
```

### Property Value

Type: Byte

## See Also

### Reference

Device Class
IO.Devices.MCP23017 Namespace

# DevicePort Property

Port Data Property (16 bites). Bits 0 to 7 correspond to PORT A and bits 8 to 15 correspond to PORT B.

**Namespace:** IO.Devices.MCP23017
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public uint Port { get; set; }
```

### Property Value

Type: UInt32

## See Also

### Reference

Device Class
IO.Devices.MCP23017 Namespace

# DevicePortA Property

Port A Data Property (8 bits).

**Namespace:** IO.Devices.MCP23017
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public byte PortA { get; set; }
```

### Property Value

Type: Byte

## See Also

### Reference

Device Class
IO.Devices.MCP23017 Namespace

# DevicePortB Property

Port B Data Property (8 bits).

**Namespace:** IO.Devices.MCP23017
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public byte PortB { get; set; }
```

### Property Value

Type: Byte

## See Also

### Reference

Device Class
IO.Devices.MCP23017 Namespace

# DevicePullups Property

Input Pullup Property (16 bits). Bits 0 to 7 correspond to PORT A and bits 8 to 15 correspond to PORT B. For each bit, 0=high impedance and 1=100k pullup.

**Namespace:** IO.Devices.MCP23017
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public uint Pullups { get; set; }
```

### Property Value

Type: UInt32

## See Also

### Reference

Device Class
IO.Devices.MCP23017 Namespace

# DevicePullupsA Property

Port A Input Pullup Property (16 bits). For each bit, 0=high impedance and 1=100k pullup.

**Namespace:** IO.Devices.MCP23017
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public byte PullupsA { get; set; }
```

### Property Value

Type: Byte

## See Also

### Reference

Device Class
IO.Devices.MCP23017 Namespace

# DevicePullupsB Property

Port B Input Pullup Property (16 bits). For each bit, 0=high impedance and 1=100k pullup.

**Namespace:** IO.Devices.MCP23017
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public byte PullupsB { get; set; }
```

### Property Value

Type: Byte

## See Also

### Reference

Device Class
IO.Devices.MCP23017 Namespace

# Device Methods

The Device type exposes the following members.

## Methods

| | Name | Description |
|---|---|---|
| | GPIO_Create | Create an MCP23017 GPIO pin object. |

Top

## See Also

**Reference**

Device Class

IO.Devices.MCP23017 Namespace

# DeviceGPIO_Create Method

Create an MCP23017 GPIO pin object.

**Namespace:** IO.Devices.MCP23017
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public Pin GPIO_Create(
        int channel,
        Direction dir,
        bool state = false
)
```

### Parameters

*channel*
Type: SystemInt32
MCP23017 channel number (0 to 15).

*dir*
Type: IO.Interfaces.GPIODirection
GPIO pin data direction.

*state* **(Optional)**
Type: SystemBoolean
Initial GPIO output state.

### Return Value

Type: Pin
GPIO pin object.

## See Also

### Reference

Device Class

IO.Devices.MCP23017 Namespace

# Device Fields

The Device type exposes the following members.

## Fields

| | Name | Description |
|---|---|---|
| | MaxChannel | Maximum I/O channel number. |
| | MinChannel | Minimum I/O channel number. |

Top

## See Also

**Reference**

Device Class

IO.Devices.MCP23017 Namespace

# DeviceMaxChannel Field

Maximum I/O channel number.

**Namespace:** IO.Devices.MCP23017
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int MaxChannel = 15
```

### Field Value

Type: Int32

## See Also

### Reference

Device Class
IO.Devices.MCP23017 Namespace

# DeviceMinChannel Field

Minimum I/O channel number.

**Namespace:** IO.Devices.MCP23017
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int MinChannel = 0
```

### Field Value

Type: Int32

## See Also

### Reference

Device Class
IO.Devices.MCP23017 Namespace

# IO.Devices.MCP23017.GPIO Namespace

MCP23017 I$^2$C GPIO Expander GPIO Pin Services.

## Classes

| | Class | Description |
|---|---|---|
| | Pin | Encapsulates MCP23017 GPIO pins. |

# Pin Class

Encapsulates MCP23017 GPIO pins.

## Inheritance Hierarchy

SystemObject IO.Devices.MCP23017.GPIOPin

**Namespace:** IO.Devices.MCP23017.GPIO
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public class Pin : Pin
```

The Pin type exposes the following members.

## Constructors

| | Name | Description |
|---|---|---|
| | Pin | Create a single MCP23017 GPIO pin. |

Top

## Properties

| | Name | Description |
|---|---|---|
| | state | Read/Write GPIO state property. |

Top

## See Also

### Reference

IO.Devices.MCP23017.GPIO Namespace

# Pin Constructor

Create a single MCP23017 GPIO pin.

**Namespace:** IO.Devices.MCP23017.GPIO
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public Pin(
        Device dev,
        int channel,
        Direction dir,
        bool state = false
)
```

### Parameters

*dev*
Type: IO.Devices.MCP23017Device
MCP23017 device object.

*channel*
Type: SystemInt32
MCP23017 I/O channel number.

*dir*
Type: IO.Interfaces.GPIODirection
GPIO pin data direction.

*state* **(Optional)**
Type: SystemBoolean
Initial GPIO output state.

## See Also

### Reference

Pin Class

IO.Devices.MCP23017.GPIO Namespace

# Pin Properties

The Pin type exposes the following members.

## Properties

| | Name | Description |
|---|---|---|
| | state | Read/Write GPIO state property. |

Top

## See Also

**Reference**

Pin Class

IO.Devices.MCP23017.GPIO Namespace

# Pinstate Property

Read/Write GPIO state property.

**Namespace:** IO.Devices.MCP23017.GPIO
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public bool state { get; set; }
```

### Property Value

Type: Boolean

### Implements

Pinstate

## See Also

### Reference

Pin Class
IO.Devices.MCP23017.GPIO Namespace

# IO.Devices.MCP23S17 Namespace

MCP23S17 SPI GPIO Expander Device Services.

## Classes

| | Class | Description |
|---|---|---|
| | Device | Encapsulates the MCP23S17 SPI I/O GPIO Expander. |

# Device Class

Encapsulates the MCP23S17 SPI I/O GPIO Expander.

## Inheritance Hierarchy

SystemObject IO.Devices.MCP23S17Device

**Namespace:** IO.Devices.MCP23S17
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public class Device
```

The Device type exposes the following members.

## Constructors

| | Name | Description |
|---|---|---|
| | Device | Constructor for a single MCP23S17 device. |

Top

## Properties

| | Name | Description |
|---|---|---|
| | Direction | Data Direction Property (16 bits). Bits 0 |

| | | |
|---|---|---|
| | | to 7 correspond to PORT A and bits 8 to 15 correspond to PORT B. For each bit, 0=input and 1=output. |
| | DirectionA | Port A Data Direction Property (8 bits). For each bit, 0=input and 1=output. |
| | DirectionB | Port B Data Direction Property (8 bits). For each bit, 0=input and 1=output. |
| | Polarity | Data Polarity Property (16 bits). Bits 0 to 7 correspond to PORT A and bits 8 to 15 correspond to PORT B. For each bit, 0=input and 1=output. |
| | PolarityA | Port A Data Polarity Property (8 bits). For each bit, 0=input and 1=output. |
| | PolarityB | Port B Data Polarity Property (8 bits). For each bit, 0=input and 1=output. |
| | Port | Port Data Property (16 bites). Bits 0 to 7 correspond to PORT A and bits 8 to 15 correspond to PORT B. |
| | PortA | Port A Data Property (8 bits). |
| | PortB | Port B Data Property (8 bits). |
| | Pullups | Input Pullup Property (16 bits). Bits 0 to 7 correspond to PORT A and bits 8 to 15 correspond to PORT B. For each bit, 0=high impedance and 1=100k pullup. |
| | | |

| | | |
|---|---|---|
| | PullupsA | Port A Input Pullup Property (16 bits). For each bit, 0=high impedance and 1=100k pullup. |
| | PullupsB | Port B Input Pullup Property (16 bits). For each bit, 0=high impedance and 1=100k pullup. |

Top

## Methods

| | Name | Description |
|---|---|---|
| | GPIO_Create | Create an MCP23S17 GPIO pin object. |

Top

## Fields

| | Name | Description |
|---|---|---|
| S | MaxChannel | Maximum I/O channel number. |
| S | MinChannel | Minimum I/O channel number. |
| S | SPI_Frequency | SPI maximum clock frequency in Hz. |
| S | SPI_Mode | SPI transfer mode. |
| S | SPI_WordSize | SPI transaction word size. |

Top

## See Also

### Reference

IO.Devices.MCP23S17 Namespace

# Device Constructor

Constructor for a single MCP23S17 device.

**Namespace:** IO.Devices.MCP23S17
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public Device(
        Device dev
)
```

### Parameters

*dev*
Type: IO.Interfaces.SPIDevice
SPI slave device object.

## See Also

### Reference

Device Class
IO.Devices.MCP23S17 Namespace

# Device Properties

The Device type exposes the following members.

## Properties

| | Name | Description |
|---|---|---|
| | Direction | Data Direction Property (16 bits). Bits 0 to 7 correspond to PORT A and bits 8 to 15 correspond to PORT B. For each bit, 0=input and 1=output. |
| | DirectionA | Port A Data Direction Property (8 bits). For each bit, 0=input and 1=output. |
| | DirectionB | Port B Data Direction Property (8 bits). For each bit, 0=input and 1=output. |
| | Polarity | Data Polarity Property (16 bits). Bits 0 to 7 correspond to PORT A and bits 8 to 15 correspond to PORT B. For each bit, 0=input and 1=output. |
| | PolarityA | Port A Data Polarity Property (8 bits). For each bit, 0=input and 1=output. |
| | PolarityB | Port B Data Polarity Property (8 bits). For each bit, 0=input and 1=output. |
| | Port | Port Data Property (16 bites). Bits 0 to 7 correspond to PORT A and bits 8 to |

| | | |
|---|---|---|
| | | 15 correspond to PORT B. |
| | PortA | Port A Data Property (8 bits). |
| | PortB | Port B Data Property (8 bits). |
| | Pullups | Input Pullup Property (16 bits). Bits 0 to 7 correspond to PORT A and bits 8 to 15 correspond to PORT B. For each bit, 0=high impedance and 1=100k pullup. |
| | PullupsA | Port A Input Pullup Property (16 bits). For each bit, 0=high impedance and 1=100k pullup. |
| | PullupsB | Port B Input Pullup Property (16 bits). For each bit, 0=high impedance and 1=100k pullup. |

Top

## See Also

**Reference**

Device Class

IO.Devices.MCP23S17 Namespace

# DeviceDirection Property

Data Direction Property (16 bits). Bits 0 to 7 correspond to PORT A and bits 8 to 15 correspond to PORT B. For each bit, 0=input and 1=output.

**Namespace:** IO.Devices.MCP23S17
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public uint Direction { get; set; }
```

### Property Value

Type: UInt32

## Remarks

This property follows the industry standard convention for data direction bit polarity (1=output) rather than the MCP23S17 `IODIR` register polarity (0=output).

## See Also

### Reference

Device Class
IO.Devices.MCP23S17 Namespace

# DeviceDirectionA Property

Port A Data Direction Property (8 bits). For each bit, 0=input and 1=output.

**Namespace:** IO.Devices.MCP23S17
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public byte DirectionA { get; set; }
```

### Property Value

Type: Byte

## Remarks

This property follows the industry standard convention for data direction bit polarity (1=output) rather than the MCP23S17 `IODIRA` register polarity (0=output).

## See Also

### Reference

Device Class
IO.Devices.MCP23S17 Namespace

# DeviceDirectionB Property

Port B Data Direction Property (8 bits). For each bit, 0=input and 1=output.

**Namespace:** IO.Devices.MCP23S17
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public byte DirectionB { get; set; }
```

### Property Value

Type: Byte

## Remarks

This property follows the industry standard convention for data direction bit polarity (1=output) rather than the MCP23S17 `IODIRA` register polarity (0=output).

## See Also

### Reference

Device Class
IO.Devices.MCP23S17 Namespace

# DevicePolarity Property

Data Polarity Property (16 bits). Bits 0 to 7 correspond to PORT A and bits 8 to 15 correspond to PORT B. For each bit, 0=input and 1=output.

**Namespace:** IO.Devices.MCP23S17
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public uint Polarity { get; set; }
```

### Property Value

Type: UInt32

## See Also

### Reference

Device Class
IO.Devices.MCP23S17 Namespace

# DevicePolarityA Property

Port A Data Polarity Property (8 bits). For each bit, 0=input and 1=output.

**Namespace:** IO.Devices.MCP23S17
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public byte PolarityA { get; set; }
```

### Property Value

Type: Byte

## See Also

### Reference

Device Class
IO.Devices.MCP23S17 Namespace

# DevicePolarityB Property

Port B Data Polarity Property (8 bits). For each bit, 0=input and 1=output.

**Namespace:** IO.Devices.MCP23S17
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public byte PolarityB { get; set; }
```

### Property Value

Type: Byte

## See Also

### Reference

Device Class
IO.Devices.MCP23S17 Namespace

# DevicePort Property

Port Data Property (16 bites). Bits 0 to 7 correspond to PORT A and bits 8 to 15 correspond to PORT B.

**Namespace:** IO.Devices.MCP23S17
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public uint Port { get; set; }
```

### Property Value

Type: UInt32

## See Also

### Reference

Device Class
IO.Devices.MCP23S17 Namespace

# DevicePortA Property

Port A Data Property (8 bits).

**Namespace:** IO.Devices.MCP23S17
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public byte PortA { get; set; }
```

### Property Value

Type: Byte

## See Also

### Reference

Device Class
IO.Devices.MCP23S17 Namespace

# DevicePortB Property

Port B Data Property (8 bits).

**Namespace:** IO.Devices.MCP23S17
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public byte PortB { get; set; }
```

### Property Value

Type: Byte

## See Also

### Reference

Device Class
IO.Devices.MCP23S17 Namespace

# DevicePullups Property

Input Pullup Property (16 bits). Bits 0 to 7 correspond to PORT A and bits 8 to 15 correspond to PORT B. For each bit, 0=high impedance and 1=100k pullup.

**Namespace:** IO.Devices.MCP23S17
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

**C#** **VB** **F#** Copy

```
public uint Pullups { get; set; }
```

### Property Value

Type: UInt32

## See Also

### Reference

Device Class
IO.Devices.MCP23S17 Namespace

# DevicePullupsA Property

Port A Input Pullup Property (16 bits). For each bit, 0=high impedance and 1=100k pullup.

**Namespace:** IO.Devices.MCP23S17
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public byte PullupsA { get; set; }
```

### Property Value

Type: Byte

## See Also

### Reference

Device Class
IO.Devices.MCP23S17 Namespace

# DevicePullupsB Property

Port B Input Pullup Property (16 bits). For each bit, 0=high impedance and 1=100k pullup.

**Namespace:** IO.Devices.MCP23S17
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public byte PullupsB { get; set; }
```

### Property Value

Type: Byte

## See Also

### Reference

Device Class
IO.Devices.MCP23S17 Namespace

# Device Methods

The Device type exposes the following members.

## Methods

| | Name | Description |
|---|---|---|
| | GPIO_Create | Create an MCP23S17 GPIO pin object. |

Top

## See Also

**Reference**

Device Class

IO.Devices.MCP23S17 Namespace

# DeviceGPIO_Create Method

Create an MCP23S17 GPIO pin object.

**Namespace:** IO.Devices.MCP23S17
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public Pin GPIO_Create(
        int channel,
        Direction dir,
        bool state = false
)
```

### Parameters

*channel*
Type: SystemInt32
MCP23S17 channel number (0 to 15).

*dir*
Type: IO.Interfaces.GPIODirection
GPIO pin data direction.

*state* **(Optional)**
Type: SystemBoolean
Initial GPIO output state.

### Return Value

Type: Pin
GPIO pin object.

## See Also

### Reference

Device Class

IO.Devices.MCP23S17 Namespace

# Device Fields

The Device type exposes the following members.

## Fields

| | Name | Description |
|---|---|---|
| S | MaxChannel | Maximum I/O channel number. |
| S | MinChannel | Minimum I/O channel number. |
| S | SPI_Frequency | SPI maximum clock frequency in Hz. |
| S | SPI_Mode | SPI transfer mode. |
| S | SPI_WordSize | SPI transaction word size. |

Top

## See Also

**Reference**

Device Class

IO.Devices.MCP23S17 Namespace

# DeviceMaxChannel Field

Maximum I/O channel number.

**Namespace:** IO.Devices.MCP23S17
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int MaxChannel = 15
```

### Field Value

Type: Int32

## See Also

### Reference

Device Class
IO.Devices.MCP23S17 Namespace

# DeviceMinChannel Field

Minimum I/O channel number.

**Namespace:** IO.Devices.MCP23S17
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int MinChannel = 0
```

### Field Value

Type: Int32

## See Also

### Reference

Device Class
IO.Devices.MCP23S17 Namespace

# DeviceSPI_Frequency Field

SPI maximum clock frequency in Hz.

**Namespace:** IO.Devices.MCP23S17
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int SPI_Frequency = 10000000
```

### Field Value

Type: Int32

## Remarks

Guaranteed only for 2.7V to 5.5V and -40°C to +85°C.

## See Also

### Reference

Device Class
IO.Devices.MCP23S17 Namespace

# DeviceSPI_Mode Field

SPI transfer mode.

**Namespace:** IO.Devices.MCP23S17
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int SPI_Mode = 0
```

### Field Value

Type: Int32

## See Also

### Reference

Device Class
IO.Devices.MCP23S17 Namespace

# DeviceSPI_WordSize Field

SPI transaction word size.

**Namespace:** IO.Devices.MCP23S17
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int SPI_WordSize = 8
```

### Field Value

Type: Int32

## See Also

### Reference

Device Class
IO.Devices.MCP23S17 Namespace

# IO.Devices.MCP23S17.GPIO Namespace

MCP23S17 SPI GPIO Expander GPIO Pin Services.

## Classes

| | Class | Description |
|---|---|---|
| | Pin | Encapsulates MCP23S17 GPIO pins. |

# Pin Class

Encapsulates MCP23S17 GPIO pins.

## Inheritance Hierarchy

SystemObject IO.Devices.MCP23S17.GPIOPin

**Namespace:** IO.Devices.MCP23S17.GPIO
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public class Pin : Pin
```

The Pin type exposes the following members.

## Constructors

| | Name | Description |
|---|---|---|
| | Pin | Create a single MCP23S17 GPIO pin. |

Top

## Properties

| | Name | Description |
|---|---|---|
| | state | Read/Write GPIO state property. |

Top

## See Also

### Reference

IO.Devices.MCP23S17.GPIO Namespace

# Pin Constructor

Create a single MCP23S17 GPIO pin.

**Namespace:** IO.Devices.MCP23S17.GPIO
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public Pin(
        Device dev,
        int channel,
        Direction dir,
        bool state = false
)
```

### Parameters

*dev*
Type: IO.Devices.MCP23S17Device
MCP23S17 device object.

*channel*
Type: SystemInt32
MCP23S17 I/O channel number.

*dir*
Type: IO.Interfaces.GPIODirection
GPIO pin data direction.

*state* **(Optional)**
Type: SystemBoolean
Initial GPIO output state.

## See Also

### Reference

Pin Class

IO.Devices.MCP23S17.GPIO Namespace

# Pin Properties

The Pin type exposes the following members.

## Properties

| | Name | Description |
|---|---|---|
| | state | Read/Write GPIO state property. |

Top

## See Also

**Reference**

Pin Class

IO.Devices.MCP23S17.GPIO Namespace

# Pinstate Property

Read/Write GPIO state property.

**Namespace:** IO.Devices.MCP23S17.GPIO
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public bool state { get; set; }
```

### Property Value

Type: Boolean

### Implements

Pinstate

## See Also

### Reference

Pin Class
IO.Devices.MCP23S17.GPIO Namespace

# IO.Devices.PCA8574 Namespace

PCA8574 (and similar) $I^2C$ GPIO Expander Device Services

## Classes

| | Class | Description |
|---|---|---|
| | Device | Encapsulates PCA8574 (and similar) $I^2C$ GPIO Expanders. |

# Device Class

Encapsulates PCA8574 (and similar) I$^2$C GPIO Expanders.

## Inheritance Hierarchy

SystemObject IO.Devices.PCA8574Device

**Namespace:** IO.Devices.PCA8574
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public class Device
```

The Device type exposes the following members.

## Constructors

| | Name | Description |
|---|---|---|
| | Device | Constructor for a PCA8574 (or similar) GPIO Expander. |

Top

## Properties

| | Name | Description |
|---|---|---|
| | Latch | This read-only property returns the last |

value written to the output latch.

Top

## Methods

| | Name | Description |
|---|---|---|
| | Read | Return actual state of the GPIO pins. |
| | Write | Write all GPIO pins. |

Top

## Fields

| | Name | Description |
|---|---|---|
| S | MAX_PINS | The number of available GPIO pins per chip. |

Top

## See Also

**Reference**

IO.Devices.PCA8574 Namespace

# Device Constructor

Constructor for a PCA8574 (or similar) GPIO Expander.

**Namespace:** IO.Devices.PCA8574
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public Device(
        Bus bus,
        int addr,
        byte states = 255
)
```

### Parameters

*bus*
Type: IO.Interfaces.I2CBus
$I^2C$ bus controller.

*addr*
Type: SystemInt32
$I^2C$ slave address.

*states* **(Optional)**
Type: SystemByte
Initial output states.

## See Also

### Reference

Device Class
IO.Devices.PCA8574 Namespace

# Device Properties

The Device type exposes the following members.

## Properties

| | Name | Description |
|---|---|---|
| | Latch | This read-only property returns the last value written to the output latch. |

Top

## See Also

**Reference**

Device Class

IO.Devices.PCA8574 Namespace

# DeviceLatch Property

This read-only property returns the last value written to the output latch.

**Namespace:** IO.Devices.PCA8574
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public byte Latch { get; }
```

### Property Value

Type: Byte

## See Also

### Reference

Device Class
IO.Devices.PCA8574 Namespace

# Device Methods

The Device type exposes the following members.

## Methods

| | Name | Description |
|---|---|---|
| | Read | Return actual state of the GPIO pins. |
| | Write | Write all GPIO pins. |

Top

## See Also

**Reference**

Device Class

IO.Devices.PCA8574 Namespace

# DeviceRead Method

Return actual state of the GPIO pins.

**Namespace:** IO.Devices.PCA8574
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public byte Read()
```

### Return Value

Type: Byte
Pin states (MSB = GPIO7).

## See Also

### Reference

Device Class
IO.Devices.PCA8574 Namespace

# DeviceWrite Method

Write all GPIO pins.

**Namespace:** IO.Devices.PCA8574
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

**C#** **VB** **F#** Copy

```
public void Write(
        byte data
)
```

### Parameters

*data*
Type: SystemByte
Data to write to pins (MSB = GPIO7).

## See Also

### Reference

Device Class
IO.Devices.PCA8574 Namespace

# Device Fields

The Device type exposes the following members.

## Fields

| | Name | Description |
|---|---|---|
| | MAX_PINS | The number of available GPIO pins per chip. |

Top

## See Also

**Reference**

Device Class

IO.Devices.PCA8574 Namespace

# DeviceMAX_PINS Field

The number of available GPIO pins per chip.

**Namespace:** IO.Devices.PCA8574
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int MAX_PINS = 8
```

### Field Value

Type: Int32

## See Also

### Reference

Device Class
IO.Devices.PCA8574 Namespace

# IO.Devices.PCA8574.GPIO Namespace

PCA8574 (and similar) I$^2$C GPIO Expander GPIO Pin Services

## Classes

| | Class | Description |
|---|---|---|
| | Pin | Encapsulates PCA8574 (and similar) I$^2$C GPIO Expander pins. |

# Pin Class

Encapsulates PCA8574 (and similar) I$^2$C GPIO Expander pins.

## Inheritance Hierarchy

SystemObject IO.Devices.PCA8574.GPIOPin

**Namespace:** IO.Devices.PCA8574.GPIO
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public class Pin : Pin
```

The Pin type exposes the following members.

## Constructors

| | Name | Description |
|---|---|---|
| | Pin | Constructor for a single GPIO pin. |

Top

## Properties

| | Name | Description |
|---|---|---|
| | state | Read/Write GPIO state property. |

Top

## See Also

### Reference

IO.Devices.PCA8574.GPIO Namespace

# Pin Constructor

Constructor for a single GPIO pin.

**Namespace:** IO.Devices.PCA8574.GPIO
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public Pin(
        Device dev,
        int num,
        Direction dir,
        bool state = false
)
```

### Parameters

*dev*
Type: IO.Devices.PCA8574Device
PCA8574 (or similar) device.

*num*
Type: SystemInt32
GPIO pin number.

*dir*
Type: IO.Interfaces.GPIODirection
Data direction.

*state* **(Optional)**
Type: SystemBoolean
Initial GPIO output state.

## See Also

### Reference

Pin Class

IO.Devices.PCA8574.GPIO Namespace

# Pin Properties

The Pin type exposes the following members.

## Properties

| | Name | Description |
|---|---|---|
| | state | Read/Write GPIO state property. |

Top

## See Also

**Reference**

Pin Class

IO.Devices.PCA8574.GPIO Namespace

# Pinstate Property

Read/Write GPIO state property.

**Namespace:** IO.Devices.PCA8574.GPIO
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public bool state { get; set; }
```

### Property Value

Type: Boolean

### Implements

Pinstate

## See Also

### Reference

Pin Class
IO.Devices.PCA8574.GPIO Namespace

# IO.Devices.PCA9534 Namespace

PCA9534 (and similar) $I^2C$ GPIO Expander Device Services

## Classes

| | Class | Description |
|---|---|---|
| | Device | Encapsulates PCA9534 (and similar) $I^2C$ GPIO Expanders. |

# Device Class

Encapsulates PCA9534 (and similar) I$^2$C GPIO Expanders.

## Inheritance Hierarchy

SystemObject IO.Devices.PCA9534Device

**Namespace:** IO.Devices.PCA9534
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public class Device
```

The Device type exposes the following members.

## Constructors

| | Name | Description |
|---|---|---|
| | Device | Constructor for a PCA9534 (or similar) GPIO Expander. |

Top

## Properties

| | Name | Description |
|---|---|---|
| | Config | This read-only property returns the last |

| | | |
|---|---|---|
| | | value written to the configuration register. |
| | Latch | This read-only property returns the last value written to the output port register. |

Top

## Methods

| | Name | Description |
|---|---|---|
| | Read | Return actual state of the GPIO pins. |
| | Read(Byte) | Read from the specified PCA9534 device register. |
| | Write(Byte) | Write all GPIO pins. |
| | Write(Byte, Byte) | Write to the specified PCA9534 device register. |

Top

## Fields

| | Name | Description |
|---|---|---|
| S | AllInputs | Configure all pins as inputs. |
| S | AllNormal | Configure all inputs as normal polarity. |
| S | AllOff | Turn all outputs off. |
| S | AllOutputs | Configure all pins as outputs. |

| | | |
|---|---|---|
| S | ConfigurationReg | Configuration Register address. |
| S | InputPolarityReg | Input Port Polarity Register address. |
| S | InputPortReg | Input Port Register address. |
| S | MAX_PINS | The number of available GPIO pins per chip. |
| S | OutputPortReg | Output Port Register address. |

Top

## See Also

**Reference**

IO.Devices.PCA9534 Namespace

# Device Constructor

Constructor for a PCA9534 (or similar) GPIO Expander.

**Namespace:** IO.Devices.PCA9534
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public Device(
        Bus bus,
        int addr,
        byte config = 255,
        byte states = 0
)
```

### Parameters

*bus*
Type: IO.Interfaces.I2CBus
$I^2C$ bus controller.

*addr*
Type: SystemInt32
$I^2C$ slave address.

*config* **(Optional)**
Type: SystemByte
GPIO pin configuration.

*states* **(Optional)**
Type: SystemByte
Initial output states.

## See Also

### Reference

Device Class

IO.Devices.PCA9534 Namespace

# Device Properties

The Device type exposes the following members.

## Properties

| | Name | Description |
|---|---|---|
| | Config | This read-only property returns the last value written to the configuration register. |
| | Latch | This read-only property returns the last value written to the output port register. |

Top

## See Also

**Reference**

Device Class

IO.Devices.PCA9534 Namespace

# DeviceConfig Property

This read-only property returns the last value written to the configuration register.

**Namespace:** IO.Devices.PCA9534
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public byte Config { get; }
```

### Property Value

Type: Byte

## See Also

### Reference

Device Class
IO.Devices.PCA9534 Namespace

# DeviceLatch Property

This read-only property returns the last value written to the output port register.

**Namespace:** IO.Devices.PCA9534
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public byte Latch { get; }
```

### Property Value

Type: Byte

## See Also

### Reference

Device Class
IO.Devices.PCA9534 Namespace

# Device Methods

## Methods

| | Name | Description |
|---|---|---|
| | Read | Return actual state of the GPIO pins. |
| | Read(Byte) | Read from the specified PCA9534 device register. |
| | Write(Byte) | Write all GPIO pins. |
| | Write(Byte, Byte) | Write to the specified PCA9534 device register. |

Top

## See Also

**Reference**

Device Class

IO.Devices.PCA9534 Namespace

# DeviceRead Method

## Overload List

| | Name | Description |
|---|---|---|
| | Read | Return actual state of the GPIO pins. |
| | Read(Byte) | Read from the specified PCA9534 device register. |

Top

## See Also

**Reference**

Device Class

IO.Devices.PCA9534 Namespace

# DeviceRead Method

Return actual state of the GPIO pins.

**Namespace:** IO.Devices.PCA9534
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public byte Read()
```

### Return Value

Type: Byte
Pin states (MSB = GPIO7).

## See Also

### Reference

Device Class
Read Overload
IO.Devices.PCA9534 Namespace

# DeviceRead Method (Byte)

Read from the specified PCA9534 device register.

**Namespace:** IO.Devices.PCA9534
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

**C#** **VB** **F#** Copy

```
public byte Read(
        byte addr
)
```

### Parameters

*addr*
Type: SystemByte
Register address.

### Return Value

Type: Byte
Register contents.

## See Also

### Reference

Device Class
Read Overload
IO.Devices.PCA9534 Namespace

# DeviceWrite Method

## Overload List

| | Name | Description |
|---|---|---|
| | Write(Byte) | Write all GPIO pins. |
| | Write(Byte, Byte) | Write to the specified PCA9534 device register. |

Top

## See Also

**Reference**

Device Class

IO.Devices.PCA9534 Namespace

# DeviceWrite Method (Byte)

Write all GPIO pins.

**Namespace:** IO.Devices.PCA9534
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

**C#** **VB** **F#** Copy

```
public void Write(
        byte data
)
```

### Parameters

*data*
Type: SystemByte
Data to write to pins (MSB = GPIO7).

## See Also

### Reference

Device Class
Write Overload
IO.Devices.PCA9534 Namespace

# DeviceWrite Method (Byte, Byte)

Write to the specified PCA9534 device register.

**Namespace:** IO.Devices.PCA9534
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public void Write(
        byte addr,
        byte data
)
```

### Parameters

*addr*
Type: SystemByte
Register address.

*data*
Type: SystemByte
Data to written.

## See Also

### Reference

Device Class
Write Overload
IO.Devices.PCA9534 Namespace

# Device Fields

The Device type exposes the following members.

## Fields

| | Name | Description |
|---|---|---|
| S | AllInputs | Configure all pins as inputs. |
| S | AllNormal | Configure all inputs as normal polarity. |
| S | AllOff | Turn all outputs off. |
| S | AllOutputs | Configure all pins as outputs. |
| S | ConfigurationReg | Configuration Register address. |
| S | InputPolarityReg | Input Port Polarity Register address. |
| S | InputPortReg | Input Port Register address. |
| S | MAX_PINS | The number of available GPIO pins per chip. |
| S | OutputPortReg | Output Port Register address. |

Top

## See Also

**Reference**

Device Class

IO.Devices.PCA9534 Namespace

# DeviceAllInputs Field

Configure all pins as inputs.

**Namespace:** IO.Devices.PCA9534
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const byte AllInputs = 255
```

### Field Value

Type: Byte

## See Also

### Reference

Device Class
IO.Devices.PCA9534 Namespace

# DeviceAllNormal Field

Configure all inputs as normal polarity.

**Namespace:** IO.Devices.PCA9534
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const byte AllNormal = 0
```

### Field Value

Type: Byte

## See Also

### Reference

Device Class
IO.Devices.PCA9534 Namespace

# DeviceAllOff Field

Turn all outputs off.

**Namespace:** IO.Devices.PCA9534
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const byte AllOff = 0
```

### Field Value

Type: Byte

## See Also

### Reference

Device Class
IO.Devices.PCA9534 Namespace

# DeviceAllOutputs Field

Configure all pins as outputs.

**Namespace:** IO.Devices.PCA9534
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const byte AllOutputs = 0
```

### Field Value

Type: Byte

## See Also

### Reference

Device Class
IO.Devices.PCA9534 Namespace

# DeviceConfigurationReg Field

Configuration Register address.

**Namespace:** IO.Devices.PCA9534
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const byte ConfigurationReg = 3
```

### Field Value

Type: Byte

## See Also

### Reference

Device Class
IO.Devices.PCA9534 Namespace

# DeviceInputPolarityReg Field

Input Port Polarity Register address.

**Namespace:** IO.Devices.PCA9534
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const byte InputPolarityReg = 2
```

### Field Value

Type: Byte

## See Also

### Reference

Device Class
IO.Devices.PCA9534 Namespace

# DeviceInputPortReg Field

Input Port Register address.

**Namespace:** IO.Devices.PCA9534
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const byte InputPortReg = 0
```

### Field Value

Type: Byte

## See Also

### Reference

Device Class
IO.Devices.PCA9534 Namespace

# DeviceMAX_PINS Field

The number of available GPIO pins per chip.

**Namespace:** IO.Devices.PCA9534
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int MAX_PINS = 8
```

### Field Value

Type: Int32

## See Also

### Reference

Device Class
IO.Devices.PCA9534 Namespace

# DeviceOutputPortReg Field

Output Port Register address.

**Namespace:** IO.Devices.PCA9534
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const byte OutputPortReg = 1
```

### Field Value

Type: Byte

## See Also

### Reference

Device Class
IO.Devices.PCA9534 Namespace

# IO.Devices.PCA9534.GPIO Namespace

PCA9534 (and similar) $I^2C$ GPIO Expander GPIO Pin Services

## Classes

| | Class | Description |
|---|---|---|
| | Pin | Encapsulates PCA9534 (and similar) $I^2C$ GPIO Expander pins. |

# Pin Class

Encapsulates PCA9534 (and similar) I$^2$C GPIO Expander pins.

## Inheritance Hierarchy

SystemObject IO.Devices.PCA9534.GPIOPin

**Namespace:** IO.Devices.PCA9534.GPIO
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public class Pin : Pin
```

The Pin type exposes the following members.

## Constructors

| | Name | Description |
|---|---|---|
| | Pin | Constructor for a single GPIO pin. |

Top

## Properties

| | Name | Description |
|---|---|---|
| | state | Read/Write GPIO state property. |

Top

## See Also

### Reference

IO.Devices.PCA9534.GPIO Namespace

# Pin Constructor

Constructor for a single GPIO pin.

**Namespace:** IO.Devices.PCA9534.GPIO
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public Pin(
        Device dev,
        int num,
        Direction dir,
        bool state = false
)
```

### Parameters

*dev*
Type: IO.Devices.PCA9534Device
PCA9534 (or similar) device.

*num*
Type: SystemInt32
GPIO pin number.

*dir*
Type: IO.Interfaces.GPIODirection
Data direction.

*state* **(Optional)**
Type: SystemBoolean
Initial GPIO output state.

## See Also

### Reference

Pin Class

IO.Devices.PCA9534.GPIO Namespace

# Pin Properties

The Pin type exposes the following members.

## Properties

| | Name | Description |
|---|---|---|
| | state | Read/Write GPIO state property. |

Top

## See Also

**Reference**

Pin Class

IO.Devices.PCA9534.GPIO Namespace

# Pinstate Property

Read/Write GPIO state property.

**Namespace:** IO.Devices.PCA9534.GPIO
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public bool state { get; set; }
```

### Property Value

Type: Boolean

### Implements

Pinstate

## See Also

### Reference

Pin Class
IO.Devices.PCA9534.GPIO Namespace

# IO.Devices.PCA9685 Namespace

PCA9685 I$^2$C PWM Controller Device Services

## Classes

| | Class | Description |
|---|---|---|
| | Device | Encapsulates the PCA9685 I$^2$C PWM Controller. |

# Device Class

Encapsulates the PCA9685 I$^2$C PWM Controller.

## Inheritance Hierarchy

SystemObject IO.Devices.PCA9685Device

**Namespace:** IO.Devices.PCA9685
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public class Device
```

The Device type exposes the following members.

## Constructors

| | Name | Description |
|---|---|---|
| | Device | Constructor for a single PCA9685 device. |

Top

## Properties

| | Name | Description |
|---|---|---|
| | Frequency | Read-only property returning the configured PWM pulse frequency. |

Top

## Methods

| | Name | Description |
|---|---|---|
| | ReadChannel | Read PCA9685 output channel data. |
| | WriteChannel | Write PCA9685 output channel data. |

Top

## Fields

| | Name | Description |
|---|---|---|
| S | INTERNAL_CLOCK | Select internal 25 MHz clock oscillator. |
| S | MAX_CHANNEL | Maximum PCA9685 output channel number. |
| S | MAX_CLOCK | Maximum clock frequency. |
| S | MIN_CHANNEL | Minimum PCA9685 output channel number. |
| S | MIN_CLOCK | Minimum clock frequency. |

Top

## See Also

**Reference**

IO.Devices.PCA9685 Namespace

# Device Constructor

Constructor for a single PCA9685 device.

**Namespace:** IO.Devices.PCA9685
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public Device(
        Bus bus,
        int addr,
        int freq = 50,
        int clock = 0
)
```

### Parameters

*bus*
Type: IO.Interfaces.I2CBus
$I^2C$ bus controller object.

*addr*
Type: SystemInt32
$I^2C$ slave address.

*freq* **(Optional)**
Type: SystemInt32
PWM pulse frequency. Default is 50 Hz.

*clock* **(Optional)**
Type: SystemInt32
PCA9685 clock source. Use `INTERNAL_CLOCK`c> to select the

internal 25 MHz clock generator.

## See Also

### Reference

Device Class

IO.Devices.PCA9685 Namespace

# Device Properties

The Device type exposes the following members.

## Properties

| | Name | Description |
|---|---|---|
| | Frequency | Read-only property returning the configured PWM pulse frequency. |

Top

## See Also

**Reference**

Device Class

IO.Devices.PCA9685 Namespace

# DeviceFrequency Property

Read-only property returning the configured PWM pulse frequency.

**Namespace:** IO.Devices.PCA9685
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public int Frequency { get; }
```

### Property Value

Type: Int32

## See Also

### Reference

Device Class
IO.Devices.PCA9685 Namespace

# Device Methods

The Device type exposes the following members.

## Methods

| | Name | Description |
|---|---|---|
| | ReadChannel | Read PCA9685 output channel data. |
| | WriteChannel | Write PCA9685 output channel data. |

Top

## See Also

**Reference**

Device Class

IO.Devices.PCA9685 Namespace

# DeviceReadChannel Method

Read PCA9685 output channel data.

**Namespace:** IO.Devices.PCA9685
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public void ReadChannel(
        byte channel,
        ref byte[] data
)
```

### Parameters

*channel*
Type: SystemByte
Output channel number.

*data*
Type: SystemByte
Output channel data (4 bytes).

## See Also

### Reference

Device Class
IO.Devices.PCA9685 Namespace

# DeviceWriteChannel Method

Write PCA9685 output channel data.

**Namespace:** IO.Devices.PCA9685
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public void WriteChannel(
        byte channel,
        byte[] data
)
```

### Parameters

*channel*
Type: SystemByte
Output channel number.

*data*
Type: SystemByte
Output channel data.

## See Also

### Reference

Device Class
IO.Devices.PCA9685 Namespace

# Device Fields

The Device type exposes the following members.

## Fields

| | Name | Description |
|---|---|---|
| S | INTERNAL_CLOCK | Select internal 25 MHz clock oscillator. |
| S | MAX_CHANNEL | Maximum PCA9685 output channel number. |
| S | MAX_CLOCK | Maximum clock frequency. |
| S | MIN_CHANNEL | Minimum PCA9685 output channel number. |
| S | MIN_CLOCK | Minimum clock frequency. |

Top

## See Also

**Reference**

Device Class

IO.Devices.PCA9685 Namespace

# DeviceINTERNAL_CLOCK Field

Select internal 25 MHz clock oscillator.

**Namespace:** IO.Devices.PCA9685
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int INTERNAL_CLOCK = 0
```

### Field Value

Type: Int32

## See Also

### Reference

Device Class
IO.Devices.PCA9685 Namespace

# DeviceMAX_CHANNEL Field

Maximum PCA9685 output channel number.

**Namespace:** IO.Devices.PCA9685
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int MAX_CHANNEL = 15
```

### Field Value

Type: Int32

## See Also

### Reference

Device Class
IO.Devices.PCA9685 Namespace

# DeviceMAX_CLOCK Field

Maximum clock frequency.

**Namespace:** IO.Devices.PCA9685
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int MAX_CLOCK = 50000000
```

### Field Value

Type: Int32

## See Also

### Reference

Device Class
IO.Devices.PCA9685 Namespace

# DeviceMIN_CHANNEL Field

Minimum PCA9685 output channel number.

**Namespace:** IO.Devices.PCA9685
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int MIN_CHANNEL = 0
```

### Field Value

Type: Int32

## See Also

### Reference

Device Class
IO.Devices.PCA9685 Namespace

# DeviceMIN_CLOCK Field

Minimum clock frequency.

**Namespace:** IO.Devices.PCA9685
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int MIN_CLOCK = 1
```

### Field Value

Type: Int32

## See Also

### Reference

Device Class
IO.Devices.PCA9685 Namespace

# IO.Devices.PCA9685.GPIO Namespace

PCA9685 I$^2$C PWM Controller GPIO Pin Services

## Classes

| | Class | Description |
|---|---|---|
| | Pin | Encapsulates PCA9685 GPIO outputs. |

# Pin Class

Encapsulates PCA9685 GPIO outputs.

## Inheritance Hierarchy

SystemObject IO.Devices.PCA9685.GPIOPin

**Namespace:** IO.Devices.PCA9685.GPIO
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public class Pin : Pin
```

The Pin type exposes the following members.

## Constructors

| | Name | Description |
|---|---|---|
| | Pin | Constructor for a single GPIO output pin. |

Top

## Properties

| | Name | Description |
|---|---|---|
| | state | Read/Write GPIO output state property. |

Top

## See Also

### Reference

IO.Devices.PCA9685.GPIO Namespace

# Pin Constructor

Constructor for a single GPIO output pin.

**Namespace:** IO.Devices.PCA9685.GPIO
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public Pin(
        Device dev,
        int channel,
        bool state = false
)
```

### Parameters

*dev*
Type: IO.Devices.PCA9685Device
PCA9685 device object.

*channel*
Type: SystemInt32
Output channel number.

*state* **(Optional)**
Type: SystemBoolean
Initial GPIO output state.

## See Also

### Reference

Pin Class

# IO.Devices.PCA9685.GPIO Namespace

# Pin Properties

The Pin type exposes the following members.

## Properties

| | Name | Description |
|---|---|---|
| | state | Read/Write GPIO output state property. |

Top

## See Also

**Reference**

Pin Class

IO.Devices.PCA9685.GPIO Namespace

# Pinstate Property

Read/Write GPIO output state property.

**Namespace:** IO.Devices.PCA9685.GPIO
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public bool state { get; set; }
```

### Property Value

Type: Boolean

### Implements

Pinstate

## See Also

### Reference

Pin Class
IO.Devices.PCA9685.GPIO Namespace

# IO.Devices.PCA9685.PWM Namespace

PCA9685 I$^2$C PWM Controller PWM Output Services

## Classes

| | Class | Description |
|---|---|---|
| | Output | Encapsulates PCA9685 PWM outputs. |

# Output Class

Encapsulates PCA9685 PWM outputs.

## Inheritance Hierarchy

SystemObject IO.Devices.PCA9685.PWMOutput

**Namespace:** IO.Devices.PCA9685.PWM
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public class Output : Output
```

The Output type exposes the following members.

## Constructors

| | Name | Description |
|---|---|---|
| | Output | Constructor for a single PWM output. |

Top

## Properties

| | Name | Description |
|---|---|---|
| | dutycycle | Write-only property for setting the PWM output duty cycle. Allowed values |

are 0.0 to 100.0 percent.

Top

## See Also

### Reference

IO.Devices.PCA9685.PWM Namespace

# Output Constructor

Constructor for a single PWM output.

**Namespace:** IO.Devices.PCA9685.PWM
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public Output(
        Device dev,
        int channel,
        double dutycycle = 0
)
```

### Parameters

*dev*
Type: IO.Devices.PCA9685Device
PCA9685 device object.

*channel*
Type: SystemInt32
Output channel number.

*dutycycle* **(Optional)**
Type: SystemDouble
Initial PWM output duty cycle.

## See Also

### Reference

Output Class

# IO.Devices.PCA9685.PWM Namespace

# Output Properties

The Output type exposes the following members.

## Properties

| | Name | Description |
|---|---|---|
| | dutycycle | Write-only property for setting the PWM output duty cycle. Allowed values are 0.0 to 100.0 percent. |

Top

## See Also

**Reference**

Output Class
IO.Devices.PCA9685.PWM Namespace

# Outputdutycycle Property

Write-only property for setting the PWM output duty cycle. Allowed values are 0.0 to 100.0 percent.

**Namespace:** IO.Devices.PCA9685.PWM
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public double dutycycle { set; }
```

### Property Value

Type: Double

### Implements

Outputdutycycle

## See Also

### Reference

Output Class
IO.Devices.PCA9685.PWM Namespace

# IO.Devices.PCA9685.Servo Namespace

PCA9685 I$^2$C PWM Controller Servo Output Services

## Classes

| | Class | Description |
|---|---|---|
| | Output | Encapsulates PCA9685 servo outputs. |

# Output Class

Encapsulates PCA9685 servo outputs.

## Inheritance Hierarchy

SystemObject IO.Devices.PCA9685.ServoOutput

**Namespace:** IO.Devices.PCA9685.Servo
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public class Output : Output
```

The Output type exposes the following members.

## Constructors

| | Name | Description |
|---|---|---|
| | Output | Constructor for a single servo output. |

Top

## Properties

| | Name | Description |
|---|---|---|
| | position | Write-only property for setting the normalized servo position. Allowed |

values are -0.0 to+1.0.

Top

## See Also

### Reference

IO.Devices.PCA9685.Servo Namespace

# Output Constructor

Constructor for a single servo output.

**Namespace:** IO.Devices.PCA9685.Servo
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public Output(
        Device dev,
        int channel,
        double position = 0
)
```

### Parameters

*dev*
Type: IO.Devices.PCA9685Device
PCA9685 device object.

*channel*
Type: SystemInt32
Output channel number.

*position* **(Optional)**
Type: SystemDouble
Initial servo position.

## See Also

### Reference

Output Class

# IO.Devices.PCA9685.Servo Namespace

# Output Properties

The Output type exposes the following members.

## Properties

| | Name | Description |
|---|---|---|
| | position | Write-only property for setting the normalized servo position. Allowed values are -0.0 to+1.0. |

Top

## See Also

**Reference**

Output Class
IO.Devices.PCA9685.Servo Namespace

# Outputposition Property

Write-only property for setting the normalized servo position. Allowed values are -0.0 to+1.0.

**Namespace:** IO.Devices.PCA9685.Servo
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public double position { set; }
```

### Property Value

Type: Double

### Implements

Outputposition

## See Also

### Reference

Output Class
IO.Devices.PCA9685.Servo Namespace

# IO.Devices.Pmod.HYGRO Namespace

Digilent Pmod HYGRO Temperature and Humdity Sensor (HDC1080) Services.

## Classes

| | Class | Description |
|---|---|---|
| | Device | Encapsulate the Digilent Pmod HYGRO Temperature and Humidity Sensor (HDC1080). |

# Device Class

Encapsulate the Digilent Pmod HYGRO Temperature and Humidity Sensor (HDC1080).

## Inheritance Hierarchy

SystemObject IO.Devices.HDC1080Device
IO.Devices.Pmod.HYGRODevice

**Namespace:** IO.Devices.Pmod.HYGRO
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public class Device : Device
```

The Device type exposes the following members.

## Constructors

| | Name | Description |
|---|---|---|
| | Device | Constructor for a Digilent Pmod HYGRO Temperature and Humidity Sensor (HDC1080). |

Top

## Properties

| | Name | Description |
|---|---|---|
| | Celsius | Read-only property returning the temperature in degrees Celsius. (Inherited from Device.) |
| | DeviceID | Read-only property returning the device ID. (Inherited from Device.) |
| | Fahrenheit | Read-only property returning the temperature in degrees Fahrenheit. (Inherited from Device.) |
| | Humidity | Read-only property returning the percentage relative humidity. (Inherited from Device.) |
| | Kelvins | Read-only property returning the temperature in Kelvins. (Inherited from Device.) |
| | ManufacturerID | Read-only property returning the manufacturer ID. (Inherited from Device.) |

Top

# Methods

| | Name | Description |
|---|---|---|
| | Read | Read from an HDC1080 device register. (Inherited from Device.) |

| | | |
|---|---|---|
| | Write | Write to an HDC1080 device register. (Inherited from Device.) |

Top

## See Also

### Reference

IO.Devices.Pmod.HYGRO Namespace

# Device Constructor

Constructor for a Digilent Pmod HYGRO Temperature and Humidity Sensor (HDC1080).

**Namespace:** IO.Devices.Pmod.HYGRO
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

**C#** **VB** **F#** Copy

```
public Device(
        Bus bus
)
```

### Parameters

*bus*
Type: IO.Interfaces.I2CBus
$I^2$ bus object.

## See Also

### Reference

Device Class
IO.Devices.Pmod.HYGRO Namespace

# Device Properties

The Device type exposes the following members.

## Properties

| | Name | Description |
|---|---|---|
| | Celsius | Read-only property returning the temperature in degrees Celsius. (Inherited from Device.) |
| | DeviceID | Read-only property returning the device ID. (Inherited from Device.) |
| | Fahrenheit | Read-only property returning the temperature in degrees Fahrenheit. (Inherited from Device.) |
| | Humidity | Read-only property returning the percentage relative humidity. (Inherited from Device.) |
| | Kelvins | Read-only property returning the temperature in Kelvins. (Inherited from Device.) |
| | ManufacturerID | Read-only property returning the manufacturer ID. |

(Inherited from Device.)

Top

## See Also

### Reference

Device Class
IO.Devices.Pmod.HYGRO Namespace

# Device Methods

The Device type exposes the following members.

## Methods

| | Name | Description |
|---|---|---|
| | Read | Read from an HDC1080 device register. (Inherited from Device.) |
| | Write | Write to an HDC1080 device register. (Inherited from Device.) |

Top

## See Also

**Reference**

Device Class

IO.Devices.Pmod.HYGRO Namespace

# IO.Devices.SN74HC595 Namespace

SN74HC595 8-Bit Shift Register Device Services

## Classes

| | Class | Description |
|---|---|---|
| | Device | Encapsulates a chain of one or more SN74HC595 8-bit shift registers. |

# Device Class

Encapsulates a chain of one or more SN74HC595 8-bit shift registers.

## Inheritance Hierarchy

SystemObject IO.Devices.SN74HC595Device

**Namespace:** IO.Devices.SN74HC595
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public class Device
```

The Device type exposes the following members.

## Constructors

| | Name | Description |
|---|---|---|
| | Device | Constructor for a chain of one or more SN74HC595 shift registers. |

Top

## Properties

| | Name | Description |
|---|---|---|
| | Length | Read-only property returning the |

| | | |
|---|---|---|
| | | number of stages in the chain. |
| | state | Read/Write shift register chain state property. |

Top

## Methods

| | Name | Description |
|---|---|---|
| | ClrBit | Clear a single bit in the shift register chain. |
| | ReadBit | Read a single bit in the shift register chain. |
| | SetBit | Set a single bit in the shift register chain. |

Top

## Fields

| | Name | Description |
|---|---|---|
| S | SPI_MaxFreq | SPI maximum clock frequency for the SNHC74HC595 shift register. (Most pessimistic datasheet limit at 2V.) |
| S | SPI_Mode | SPI clock mode for the SNHC74HC595 shift register. |

Top

## See Also

**Reference**

IO.Devices.SN74HC595 Namespace

# Device Constructor

Constructor for a chain of one or more SN74HC595 shift registers.

**Namespace:** IO.Devices.SN74HC595
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public Device(
        Device dev,
        int stages = 1,
        byte[] initialstate = null
)
```

### Parameters

*dev*
Type: IO.Interfaces.SPIDevice
SPI device object.

*stages* **(Optional)**
Type: SystemInt32
Number of stages in the chain.

*initialstate* **(Optional)**
Type: SystemByte
Initial shift register chain state.

## See Also

### Reference

Device Class

# IO.Devices.SN74HC595 Namespace

# Device Properties

The Device type exposes the following members.

## Properties

| | Name | Description |
|---|---|---|
| | Length | Read-only property returning the number of stages in the chain. |
| | state | Read/Write shift register chain state property. |

Top

## See Also

**Reference**

Device Class

IO.Devices.SN74HC595 Namespace

# DeviceLength Property

Read-only property returning the number of stages in the chain.

**Namespace:** IO.Devices.SN74HC595
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public int Length { get; }
```

### Property Value

Type: Int32

## See Also

### Reference

Device Class
IO.Devices.SN74HC595 Namespace

# Devicestate Property

Read/Write shift register chain state property.

**Namespace:** IO.Devices.SN74HC595
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public byte[] state { get; set; }
```

### Property Value

Type: Byte

## See Also

### Reference

Device Class
IO.Devices.SN74HC595 Namespace

# Device Methods

The Device type exposes the following members.

## Methods

| | Name | Description |
|---|---|---|
| | ClrBit | Clear a single bit in the shift register chain. |
| | ReadBit | Read a single bit in the shift register chain. |
| | SetBit | Set a single bit in the shift register chain. |

Top

## See Also

**Reference**

Device Class

IO.Devices.SN74HC595 Namespace

# DeviceClrBit Method

Clear a single bit in the shift register chain.

**Namespace:** IO.Devices.SN74HC595
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public void ClrBit(
        int index,
        byte mask
)
```

### Parameters

*index*
Type: SystemInt32
Shift register stage number. Zero indicates the first register stage.

*mask*
Type: SystemByte
Shift register bit mask.

## See Also

### Reference

Device Class
IO.Devices.SN74HC595 Namespace

# DeviceReadBit Method

Read a single bit in the shift register chain.

**Namespace:** IO.Devices.SN74HC595
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public bool ReadBit(
        int index,
        byte mask
)
```

### Parameters

*index*
Type: SystemInt32
Shift register stage number. Zero indicates the first register stage.

*mask*
Type: SystemByte
Shift register bit mask.

### Return Value

Type: Boolean
Boolean bit value.

## See Also

### Reference

Device Class

# IO.Devices.SN74HC595 Namespace

# DeviceSetBit Method

Set a single bit in the shift register chain.

**Namespace:** IO.Devices.SN74HC595
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public void SetBit(
        int index,
        byte mask
)
```

### Parameters

*index*
Type: SystemInt32
Shift register stage number. Zero indicates the first register stage.

*mask*
Type: SystemByte
Shift register bit mask.

## See Also

### Reference

Device Class
IO.Devices.SN74HC595 Namespace

# Device Fields

The Device type exposes the following members.

## Fields

| | Name | Description |
|---|---|---|
| | SPI_MaxFreq | SPI maximum clock frequency for the SNHC74HC595 shift register. (Most pessimistic datasheet limit at 2V.) |
| | SPI_Mode | SPI clock mode for the SNHC74HC595 shift register. |

Top

## See Also

**Reference**

Device Class

IO.Devices.SN74HC595 Namespace

# DeviceSPI_MaxFreq Field

SPI maximum clock frequency for the SNHC74HC595 shift register. (Most pessimistic datasheet limit at 2V.)

**Namespace:** IO.Devices.SN74HC595
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int SPI_MaxFreq = 4000000
```

### Field Value

Type: Int32

## See Also

### Reference

Device Class
IO.Devices.SN74HC595 Namespace

# DeviceSPI_Mode Field

SPI clock mode for the SNHC74HC595 shift register.

**Namespace:** IO.Devices.SN74HC595
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int SPI_Mode = 0
```

### Field Value

Type: Int32

## See Also

### Reference

Device Class
IO.Devices.SN74HC595 Namespace

# IO.Devices.SN74HC595.GPIO Namespace

SN74HC595 8-Bit Shift Register GPIO Pin Services

## Classes

| | Class | Description |
|---|---|---|
| | Pin | Encapsulates SN74HC595 GPIO outputs. |

# Pin Class

Encapsulates SN74HC595 GPIO outputs.

## Inheritance Hierarchy

SystemObject IO.Devices.SN74HC595.GPIOPin

**Namespace:** IO.Devices.SN74HC595.GPIO
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public class Pin : Pin
```

The Pin type exposes the following members.

## Constructors

| | Name | Description |
|---|---|---|
| | Pin | Constructor for a single GPIO output pin. |

Top

## Properties

| | Name | Description |
|---|---|---|
| | state | Read/Write GPIO pin state property. |

Top

## See Also

### Reference

IO.Devices.SN74HC595.GPIO Namespace

# Pin Constructor

Constructor for a single GPIO output pin.

**Namespace:** IO.Devices.SN74HC595.GPIO
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public Pin(
        Device dev,
        int pos,
        bool state = false
)
```

### Parameters

*dev*
Type: IO.Devices.SN74HC595Device
SN74HC595 device object.

*pos*
Type: SystemInt32
Bit position, numbered left to right. Zero indicates the most significant bit of the first shift register stage.

*state* **(Optional)**
Type: SystemBoolean
Initial GPIO output state.

## See Also

### Reference

Pin Class
IO.Devices.SN74HC595.GPIO Namespace

# Pin Properties

The Pin type exposes the following members.

## Properties

| | Name | Description |
|---|---|---|
| | state | Read/Write GPIO pin state property. |

Top

## See Also

**Reference**

Pin Class

IO.Devices.SN74HC595.GPIO Namespace

# Pinstate Property

Read/Write GPIO pin state property.

**Namespace:** IO.Devices.SN74HC595.GPIO
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public bool state { get; set; }
```

### Property Value

Type: Boolean

### Implements

Pinstate

## See Also

### Reference

Pin Class
IO.Devices.SN74HC595.GPIO Namespace

# IO.Devices.TH02 Namespace

TH02 $I^2C$ Temperature/Humidity Sensor Services

## Classes

| | Class | Description |
|---|---|---|
| | Device | Encapsulates the TH02 temperature and humidity sensor. |

# Device Class

Encapsulates the TH02 temperature and humidity sensor.

## Inheritance Hierarchy

SystemObject IO.Devices.TH02Device
IO.Devices.Grove.Temperature_HumidityDevice

**Namespace:** IO.Devices.TH02
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public class Device : Sensor, Sensor
```

The Device type exposes the following members.

## Constructors

| | Name | Description |
|---|---|---|
| | Device | Constructor for an TH02 temperature and humidity sensor object. |

Top

## Properties

| | Name | Description |
|---|---|---|

| | | |
|---|---|---|
| | Celsius | Read-only property returning the temperature in degrees Celsius. |
| | DeviceID | Read-only property returning the device ID. |
| | Fahrenheit | Read-only property returning the temperature in degrees Fahrenheit. |
| | Humidity | Read-only property returning the percentage relative humidity. |
| | Kelvins | Read-only property returning the temperature in Kelvins. |

Top

## See Also

**Reference**

IO.Devices.TH02 Namespace

# Device Constructor

Constructor for an TH02 temperature and humidity sensor object.

**Namespace:** IO.Devices.TH02
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

**C#** **VB** **F#** Copy

```
public Device(
        Bus bus
)
```

### Parameters

*bus*
Type: IO.Interfaces.I2CBus
$I^2C$ bus controller.

## See Also

### Reference

Device Class
IO.Devices.TH02 Namespace

# Device Properties

The Device type exposes the following members.

## Properties

| | Name | Description |
|---|---|---|
| | Celsius | Read-only property returning the temperature in degrees Celsius. |
| | DeviceID | Read-only property returning the device ID. |
| | Fahrenheit | Read-only property returning the temperature in degrees Fahrenheit. |
| | Humidity | Read-only property returning the percentage relative humidity. |
| | Kelvins | Read-only property returning the temperature in Kelvins. |

Top

## See Also

### Reference

Device Class
IO.Devices.TH02 Namespace

# DeviceCelsius Property

Read-only property returning the temperature in degrees Celsius.

**Namespace:** IO.Devices.TH02
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public double Celsius { get; }
```

### Property Value

Type: Double

### Implements

SensorCelsius

## See Also

### Reference

Device Class
IO.Devices.TH02 Namespace

# DeviceDeviceID Property

Read-only property returning the device ID.

**Namespace:** IO.Devices.TH02
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public byte DeviceID { get; }
```

### Property Value

Type: Byte

## See Also

### Reference

Device Class
IO.Devices.TH02 Namespace

# DeviceFahrenheit Property

Read-only property returning the temperature in degrees Fahrenheit.

**Namespace:** IO.Devices.TH02
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public double Fahrenheit { get; }
```

### Property Value

Type: Double

### Implements

SensorFahrenheit

## See Also

### Reference

Device Class
IO.Devices.TH02 Namespace

# DeviceHumidity Property

Read-only property returning the percentage relative humidity.

**Namespace:** IO.Devices.TH02
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public double Humidity { get; }
```

### Property Value

Type: Double

### Implements

SensorHumidity

## See Also

### Reference

Device Class
IO.Devices.TH02 Namespace

# DeviceKelvins Property

Read-only property returning the temperature in Kelvins.

**Namespace:** IO.Devices.TH02
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public double Kelvins { get; }
```

### Property Value

Type: Double

### Implements

SensorKelvins

## See Also

### Reference

Device Class
IO.Devices.TH02 Namespace

# IO.Devices.Thermistor Namespace

Thermistor Modeling Services

## Classes

| | Class | Description |
|---|---|---|
| | NTC_B | Encapsulate an NTC thermistor. |

# NTC_B Class

Encapsulate an NTC thermistor.

## Inheritance Hierarchy

SystemObject IO.Devices.ThermistorNTC_B

**Namespace:** IO.Devices.Thermistor
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public class NTC_B
```

The NTC_B type exposes the following members.

## Constructors

| | Name | Description |
|---|---|---|
| | NTC_B | Constructor for a single NTC thermistor object instance. |

Top

## Methods

| | Name | Description |
|---|---|---|
| | Kelvins | Kelvin temperature as a function of the |

thermistor resistance.

Top

## See Also

### Reference

IO.Devices.Thermistor Namespace

# NTC_B Constructor

Constructor for a single NTC thermistor object instance.

**Namespace:** IO.Devices.Thermistor
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public NTC_B(
        double B,
        double R0,
        double T0 = 298.15
)
```

### Parameters

*B*
Type: SystemDouble
Thermistor B parameter.

*R0*
Type: SystemDouble
Thermistor resistance in ohms at the specified reference temperature.

*T0* **(Optional)**
Type: SystemDouble
Thermistor reference temperature in Kelvins.

## See Also

### Reference

NTC_B Class
IO.Devices.Thermistor Namespace

# NTC_B Methods

The NTC_B type exposes the following members.

## Methods

| | Name | Description |
|---|---|---|
| | Kelvins | Kelvin temperature as a function of the thermistor resistance. |

Top

## See Also

**Reference**

NTC_B Class

IO.Devices.Thermistor Namespace

# NTC_BKelvins Method

Kelvin temperature as a function of the thermistor resistance.

**Namespace:** IO.Devices.Thermistor
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public double Kelvins(
        double R
)
```

### Parameters

*R*
Type: SystemDouble
Thermistor resistance in ohms.

### Return Value

Type: Double
Temperature in Kelvins.

## See Also

### Reference

NTC_B Class
IO.Devices.Thermistor Namespace

# IO.Devices.USB.Munts Namespace

Vendor and Product Identifiers for Munts Technologies (http://tech.munts.com) USB devices

## Classes

| | Class | Description |
|---|---|---|
| | HID | USB device constants for Munts Technologies USB HID devices. |
| | Serial | USB device constants for Munts Technologies USB serial port devices. |

# HID Class

USB device constants for Munts Technologies USB HID devices.

## Inheritance Hierarchy

SystemObject IO.Devices.USB.MuntsHID

**Namespace:** IO.Devices.USB.Munts
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static class HID
```

The HID type exposes the following members.

## Fields

| | Name | Description |
|---|---|---|
| | Product | Product ID for Munts Technologies USB hid devices. |
| | Vendor | Vendor ID for Munts Technologies |

Top

## See Also

### Reference

# IO.Devices.USB.Munts Namespace

# HID Fields

The HID type exposes the following members.

## Fields

| | Name | Description |
|---|---|---|
| | Product | Product ID for Munts Technologies USB hid devices. |
| | Vendor | Vendor ID for Munts Technologies |

Top

## See Also

**Reference**

HID Class

IO.Devices.USB.Munts Namespace

# HIDProduct Field

Product ID for Munts Technologies USB hid devices.

**Namespace:** IO.Devices.USB.Munts
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int Product = 2810
```

### Field Value

Type: Int32

## See Also

### Reference

HID Class
IO.Devices.USB.Munts Namespace

# HIDVendor Field

Vendor ID for Munts Technologies

**Namespace:** IO.Devices.USB.Munts
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int Vendor = 5840
```

### Field Value

Type: Int32

## See Also

### Reference

HID Class
IO.Devices.USB.Munts Namespace

# Serial Class

USB device constants for Munts Technologies USB serial port devices.

## Inheritance Hierarchy

SystemObject IO.Devices.USB.MuntsSerial

**Namespace:** IO.Devices.USB.Munts
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static class Serial
```

The Serial type exposes the following members.

## Fields

| | Name | Description |
|---|---|---|
| | Product | Product ID for Munts Technologies USB serial port devices. |
| | Vendor | Vendor ID for Munts Technologies |

Top

## See Also

### Reference

# IO.Devices.USB.Munts Namespace

# Serial Fields

The Serial type exposes the following members.

## Fields

| | Name | Description |
|---|---|---|
| | Product | Product ID for Munts Technologies USB serial port devices. |
| | Vendor | Vendor ID for Munts Technologies |

Top

## See Also

**Reference**

Serial Class

IO.Devices.USB.Munts Namespace

# SerialProduct Field

Product ID for Munts Technologies USB serial port devices.

**Namespace:** IO.Devices.USB.Munts
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int Product = 2811
```

### Field Value

Type: Int32

## See Also

### Reference

Serial Class
IO.Devices.USB.Munts Namespace

# SerialVendor Field

Vendor ID for Munts Technologies

**Namespace:** IO.Devices.USB.Munts
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int Vendor = 5840
```

### Field Value

Type: Int32

## See Also

### Reference

Serial Class
IO.Devices.USB.Munts Namespace

# IO.Interfaces.ADC Namespace

Abstract Interface for ADC (Analog to Digital Converter) Inputs

## Classes

| | Class | Description |
|---|---|---|
| | Input | Encapsulates ADC voltage inputs. |

## Interfaces

| | Interface | Description |
|---|---|---|
| | Sample | Abstract interface for ADC inputs returning an integer sample value. |
| | Voltage | Abstract interface for ADC inputs returning a floating point voltage value. |

# Input Class

Encapsulates ADC voltage inputs.

## Inheritance Hierarchy

SystemObject IO.Interfaces.ADCInput

**Namespace:** IO.Interfaces.ADC
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public class Input : Voltage
```

The Input type exposes the following members.

## Constructors

| | Name | Description |
|---|---|---|
| | Input | Create an ADC voltage input. |

Top

## Properties

| | Name | Description |
|---|---|---|
| | voltage | Read-only property returning the analog input voltage. |

Top

## See Also

### Reference

IO.Interfaces.ADC Namespace

# Input Constructor

Create an ADC voltage input.

**Namespace:** IO.Interfaces.ADC
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public Input(
        Sample input,
        double reference,
        double gain = 1
)
```

### Parameters

*input*
Type: IO.Interfaces.ADCSample
ADC sample object.

*reference*
Type: SystemDouble
ADC reference in volts.

*gain* **(Optional)**
Type: SystemDouble
ADC input gain in volts per volt.

## See Also

### Reference

Input Class

# IO.Interfaces.ADC Namespace

# Input Properties

The Input type exposes the following members.

## Properties

| | Name | Description |
|---|---|---|
| | voltage | Read-only property returning the analog input voltage. |

Top

## See Also

**Reference**

Input Class

IO.Interfaces.ADC Namespace

# Inputvoltage Property

Read-only property returning the analog input voltage.

**Namespace:** IO.Interfaces.ADC
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public double voltage { get; }
```

### Property Value

Type: Double

### Implements

Voltagevoltage

## See Also

### Reference

Input Class
IO.Interfaces.ADC Namespace

# Sample Interface

Abstract interface for ADC inputs returning an integer sample value.

**Namespace:** IO.Interfaces.ADC
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public interface Sample
```

The Sample type exposes the following members.

## Properties

| | Name | Description |
|---|---|---|
| | resolution | Read-only property returning the number of bits of resolution. |
| | sample | Read-only property returning an integer analog sample value. |

Top

## See Also

### Reference

IO.Interfaces.ADC Namespace

# Sample Properties

The Sample type exposes the following members.

## Properties

| | Name | Description |
|---|---|---|
| | resolution | Read-only property returning the number of bits of resolution. |
| | sample | Read-only property returning an integer analog sample value. |

Top

## See Also

**Reference**

Sample Interface

IO.Interfaces.ADC Namespace

# Sampleresolution Property

Read-only property returning the number of bits of resolution.

**Namespace:** IO.Interfaces.ADC
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
int resolution { get; }
```

### Property Value

Type: Int32

## See Also

### Reference

Sample Interface
IO.Interfaces.ADC Namespace

# Samplesample Property

Read-only property returning an integer analog sample value.

**Namespace:** IO.Interfaces.ADC
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
int sample { get; }
```

### Property Value

Type: Int32

## See Also

### Reference

Sample Interface
IO.Interfaces.ADC Namespace

# Voltage Interface

Abstract interface for ADC inputs returning a floating point voltage value.

**Namespace:** IO.Interfaces.ADC
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public interface Voltage
```

The Voltage type exposes the following members.

## Properties

| | Name | Description |
|---|---|---|
| | voltage | Read-only property returning a floating point analog voltage value. |

Top

## See Also

### Reference

IO.Interfaces.ADC Namespace

# Voltage Properties

The Voltage type exposes the following members.

## Properties

| | Name | Description |
|---|---|---|
| | voltage | Read-only property returning a floating point analog voltage value. |

Top

## See Also

**Reference**

Voltage Interface
IO.Interfaces.ADC Namespace

# Voltagevoltage Property

Read-only property returning a floating point analog voltage value.

**Namespace:** IO.Interfaces.ADC
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
double voltage { get; }
```

### Property Value

Type: Double

## See Also

### Reference

Voltage Interface
IO.Interfaces.ADC Namespace

# IO.Interfaces.DAC Namespace

Abstract Interface for DAC (Digital to Analog Converter) Outputs

## Classes

| | Class | Description |
|---|---|---|
| | Output | Encapsulates DAC voltage outputs. |

## Interfaces

| | Interface | Description |
|---|---|---|
| | Sample | Abstract interface for DAC outputs accepting an integer output sample value. |
| | Voltage | Abstract interface for DAC outputs accepting a floating point output voltage value. |

# Output Class

Encapsulates DAC voltage outputs.

## Inheritance Hierarchy

SystemObject IO.Interfaces.DACOutput

**Namespace:** IO.Interfaces.DAC
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public class Output : Voltage
```

The Output type exposes the following members.

## Constructors

| | Name | Description |
|---|---|---|
| | Output | Create an DAC voltage output. |

Top

## Properties

| | Name | Description |
|---|---|---|
| | voltage | Write-only for setting the DAC output voltage. |

Top

## See Also

### Reference

IO.Interfaces.DAC Namespace

# Output Constructor

Create an DAC voltage output.

**Namespace:** IO.Interfaces.DAC
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public Output(
        Sample output,
        double reference,
        double gain = 1
)
```

### Parameters

*output*
Type: IO.Interfaces.DACSample
DAC output object.

*reference*
Type: SystemDouble
DAC output reference in volts.

*gain* **(Optional)**
Type: SystemDouble
DAC output gain in volts per volt.

## See Also

### Reference

Output Class

# IO.Interfaces.DAC Namespace

# Output Properties

The Output type exposes the following members.

## Properties

| | Name | Description |
|---|---|---|
| | voltage | Write-only for setting the DAC output voltage. |

Top

## See Also

**Reference**

Output Class

IO.Interfaces.DAC Namespace

# Outputvoltage Property

Write-only for setting the DAC output voltage.

**Namespace:** IO.Interfaces.DAC
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public double voltage { set; }
```

### Property Value

Type: Double

### Implements

Voltagevoltage

## See Also

### Reference

Output Class
IO.Interfaces.DAC Namespace

# Sample Interface

Abstract interface for DAC outputs accepting an integer output sample value.

**Namespace:** IO.Interfaces.DAC
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public interface Sample
```

The Sample type exposes the following members.

## Properties

| | Name | Description |
|---|---|---|
| | resolution | Read-only property returning the number of bits of resolution. |
| | sample | Write-only property for setting the DAC output level. |

Top

## See Also

### Reference

IO.Interfaces.DAC Namespace

# Sample Properties

The Sample type exposes the following members.

## Properties

| | Name | Description |
|---|---|---|
| | resolution | Read-only property returning the number of bits of resolution. |
| | sample | Write-only property for setting the DAC output level. |

Top

## See Also

**Reference**

Sample Interface

IO.Interfaces.DAC Namespace

# Sampleresolution Property

Read-only property returning the number of bits of resolution.

**Namespace:** IO.Interfaces.DAC
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
int resolution { get; }
```

### Property Value

Type: Int32

## See Also

### Reference

Sample Interface
IO.Interfaces.DAC Namespace

# Samplesample Property

Write-only property for setting the DAC output level.

**Namespace:** IO.Interfaces.DAC
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
int sample { set; }
```

### Property Value

Type: Int32

## See Also

### Reference

Sample Interface
IO.Interfaces.DAC Namespace

# Voltage Interface

Abstract interface for DAC outputs accepting a floating point output voltage value.

**Namespace:** IO.Interfaces.DAC
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public interface Voltage
```

The Voltage type exposes the following members.

## Properties

| | Name | Description |
|---|---|---|
| | voltage | Write-only property for setting the DAC output voltage. |

Top

## See Also

### Reference

IO.Interfaces.DAC Namespace

# Voltage Properties

The Voltage type exposes the following members.

## Properties

| | Name | Description |
|---|---|---|
| | voltage | Write-only property for setting the DAC output voltage. |

Top

## See Also

**Reference**

Voltage Interface
IO.Interfaces.DAC Namespace

# Voltagevoltage Property

Write-only property for setting the DAC output voltage.

**Namespace:** IO.Interfaces.DAC
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
double voltage { set; }
```

### Property Value

Type: Double

## See Also

### Reference

Voltage Interface
IO.Interfaces.DAC Namespace

# IO.Interfaces.GPIO Namespace

Abstract Interface for GPIO (General Purpose Input/Output) Pins

## Interfaces

| | Interface | Description |
|---|---|---|
| | Pin | Abstract interface for GPIO pins. |

## Enumerations

| | Enumeration | Description |
|---|---|---|
| | Direction | GPIO pin data direction settings. |

# Direction Enumeration

GPIO pin data direction settings.

**Namespace:** IO.Interfaces.GPIO
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public enum Direction
```

## Members

| Member name | Value | Description |
|---|---|---|
| Input | 0 | Input pin (read only) |
| Output | 1 | Output pin (read or write) |

## See Also

### Reference

IO.Interfaces.GPIO Namespace

# Pin Interface

Abstract interface for GPIO pins.

**Namespace:** IO.Interfaces.GPIO
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public interface Pin
```

The Pin type exposes the following members.

## Properties

| | Name | Description |
|---|---|---|
| | state | Read/Write GPIO state property. |

Top

## See Also

### Reference

IO.Interfaces.GPIO Namespace

# Pin Properties

The Pin type exposes the following members.

## Properties

| | Name | Description |
|---|---|---|
| | state | Read/Write GPIO state property. |

Top

## See Also

**Reference**

Pin Interface

IO.Interfaces.GPIO Namespace

# Pinstate Property

Read/Write GPIO state property.

**Namespace:** IO.Interfaces.GPIO
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
bool state { get; set; }
```

### Property Value

Type: Boolean

## See Also

### Reference

Pin Interface
IO.Interfaces.GPIO Namespace

# IO.Interfaces.Humidity Namespace

Abstract Interface for Humidity Sensors

## Interfaces

| | Interface | Description |
|---|---|---|
| | Sensor | Abstract interface for humidity sensors. |

# Sensor Interface

Abstract interface for humidity sensors.

**Namespace:** IO.Interfaces.Humidity
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public interface Sensor
```

The Sensor type exposes the following members.

## Properties

| | Name | Description |
|---|---|---|
| | Humidity | Read-only property returning the percentage relative humidity. |

Top

## See Also

### Reference

IO.Interfaces.Humidity Namespace

# Sensor Properties

The Sensor type exposes the following members.

## Properties

| | Name | Description |
|---|---|---|
| | Humidity | Read-only property returning the percentage relative humidity. |

Top

## See Also

**Reference**

Sensor Interface

IO.Interfaces.Humidity Namespace

# SensorHumidity Property

Read-only property returning the percentage relative humidity.

**Namespace:** IO.Interfaces.Humidity
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
double Humidity { get; }
```

### Property Value

Type: Double

## See Also

### Reference

Sensor Interface
IO.Interfaces.Humidity Namespace

# IO.Interfaces.I2C Namespace

Abstract Interface for I²C (Inter-Integrated Circuit) Bus Controllers

## Classes

| | Class | Description |
|---|---|---|
| | Device | Encapsulates a single I²C slave device. |
| | Speeds | I²C bus speed constants. |

## Interfaces

| | Interface | Description |
|---|---|---|
| | Bus | Abstract interface for I²C bus controllers. |

# Bus Interface

Abstract interface for I²C bus controllers.

**Namespace:** IO.Interfaces.I2C
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public interface Bus
```

The Bus type exposes the following members.

## Methods

| | Name | Description |
|---|---|---|
| | Read | Read bytes from an I²C slave device. |
| | Transaction | Write and read bytes to and from an I²C slave device. |
| | Write | Write bytes to an I²C slave device. |

Top

## See Also

### Reference

IO.Interfaces.I2C Namespace

# Bus Methods

The Bus type exposes the following members.

## Methods

| | Name | Description |
|---|---|---|
| | Read | Read bytes from an $I^2C$ slave device. |
| | Transaction | Write and read bytes to and from an $I^2C$ slave device. |
| | Write | Write bytes to an $I^2C$ slave device. |

Top

## See Also

**Reference**

Bus Interface

IO.Interfaces.I2C Namespace

# BusRead Method

Read bytes from an I²C slave device.

**Namespace:** IO.Interfaces.I2C
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
void Read(
        int slaveaddr,
        byte[] resp,
        int resplen
)
```

### Parameters

*slaveaddr*
Type: SystemInt32
I²C slave address.

*resp*
Type: SystemByte
Response buffer.

*resplen*
Type: SystemInt32
Number of bytes to read.

## See Also

### Reference

Bus Interface
IO.Interfaces.I2C Namespace

# BusTransaction Method

Write and read bytes to and from an I²C slave device.

**Namespace:** IO.Interfaces.I2C
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
void Transaction(
        int slaveaddr,
        byte[] cmd,
        int cmdlen,
        byte[] resp,
        int resplen,
        int delayus = 0
)
```

### Parameters

*slaveaddr*
Type: SystemInt32
I²C slave address.

*cmd*
Type: SystemByte
Command buffer.

*cmdlen*
Type: SystemInt32
Number of bytes to write.

*resp*

Type: SystemByte
Response buffer.

*resplen*

Type: SystemInt32
Number of bytes to read.

*delayus* **(Optional)**

Type: SystemInt32
Delay in microseconds between the $I^2C$ write and read cycles.
Allowed values are 0 to 65535 microseconds.

## See Also

### Reference

Bus Interface
IO.Interfaces.I2C Namespace

# BusWrite Method

Write bytes to an I²C slave device.

**Namespace:** IO.Interfaces.I2C
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
void Write(
        int slaveaddr,
        byte[] cmd,
        int cmdlen
)
```

### Parameters

*slaveaddr*
Type: SystemInt32
I²C slave address.

*cmd*
Type: SystemByte
Command buffer.

*cmdlen*
Type: SystemInt32
Number of bytes to write.

## See Also

### Reference

Bus Interface
IO.Interfaces.I2C Namespace

# Device Class

Encapsulates a single I²C slave device.

## Inheritance Hierarchy

SystemObject IO.Interfaces.I2CDevice

**Namespace:** IO.Interfaces.I2C
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public class Device
```

The Device type exposes the following members.

## Constructors

| | Name | Description |
|---|---|---|
| | Device | Create an I²C slave device. |

Top

## Methods

| | Name | Description |
|---|---|---|
| | Read | Read bytes from an I²C slave device. |
| | | |

| | | |
|---|---|---|
| | Transaction | Write and read bytes to and from an $I^2C$ slave device. |
| | Write | Write bytes to an $I^2C$ slave device. |

Top

## See Also

### Reference

IO.Interfaces.I2C Namespace

# Device Constructor

Create an I²C slave device.

**Namespace:** IO.Interfaces.I2C
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public Device(
        Bus bus,
        int slaveaddr
)
```

### Parameters

*bus*
Type: IO.Interfaces.I2CBus
I²C bus controller object.

*slaveaddr*
Type: SystemInt32
I²C slave address.

## See Also

### Reference

Device Class
IO.Interfaces.I2C Namespace

# Device Methods

The Device type exposes the following members.

## Methods

| | Name | Description |
|---|---|---|
| | Read | Read bytes from an $I^2C$ slave device. |
| | Transaction | Write and read bytes to and from an $I^2C$ slave device. |
| | Write | Write bytes to an $I^2C$ slave device. |

Top

## See Also

**Reference**

Device Class

IO.Interfaces.I2C Namespace

# DeviceRead Method

Read bytes from an I²C slave device.

**Namespace:** IO.Interfaces.I2C
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public void Read(
        byte[] resp,
        int resplen
)
```

### Parameters

*resp*
Type: SystemByte
Response buffer.

*resplen*
Type: SystemInt32
Number of bytes to read.

## See Also

### Reference

Device Class
IO.Interfaces.I2C Namespace

# DeviceTransaction Method

Write and read bytes to and from an I²C slave device.

**Namespace:** IO.Interfaces.I2C
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public void Transaction(
        byte[] cmd,
        int cmdlen,
        byte[] resp,
        int resplen,
        int delayus = 0
)
```

### Parameters

*cmd*
Type: SystemByte
Command buffer.

*cmdlen*
Type: SystemInt32
Number of bytes to write.

*resp*
Type: SystemByte
Response buffer.

*resplen*
Type: SystemInt32
Number of bytes to read.

*delayus* **(Optional)**

Type: SystemInt32

Delay in microseconds between the $I^2C$ write and read cycles. Allowed values are 0 to 65535 microseconds.

## See Also

### Reference

Device Class

IO.Interfaces.I2C Namespace

# DeviceWrite Method

Write bytes to an I²C slave device.

**Namespace:** IO.Interfaces.I2C
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public void Write(
        byte[] cmd,
        int cmdlen
)
```

### Parameters

*cmd*
Type: SystemByte
Command buffer.

*cmdlen*
Type: SystemInt32
Number of bytes to write.

## See Also

### Reference

Device Class
IO.Interfaces.I2C Namespace

# Speeds Class

$I^2C$ bus speed constants.

## Inheritance Hierarchy

SystemObject IO.Interfaces.I2CSpeeds

**Namespace:** IO.Interfaces.I2C
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static class Speeds
```

The Speeds type exposes the following members.

## Fields

| | Name | Description |
|---|---|---|
| | FastMode | Fast Mode |
| | FastModePlus | Fast Mode Plus |
| | StandardMode | Standard Mode |

Top

## See Also

## Reference

IO.Interfaces.I2C Namespace

# Speeds Fields

The Speeds type exposes the following members.

## Fields

| | Name | Description |
|---|---|---|
| | FastMode | Fast Mode |
| | FastModePlus | Fast Mode Plus |
| | StandardMode | Standard Mode |

Top

## See Also

**Reference**

Speeds Class

IO.Interfaces.I2C Namespace

# SpeedsFastMode Field

Fast Mode

**Namespace:** IO.Interfaces.I2C
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int FastMode = 400000
```

### Field Value

Type: Int32

## See Also

### Reference

Speeds Class
IO.Interfaces.I2C Namespace

# SpeedsFastModePlus Field

Fast Mode Plus

**Namespace:** IO.Interfaces.I2C
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int FastModePlus = 1000000
```

### Field Value

Type: Int32

## See Also

### Reference

Speeds Class
IO.Interfaces.I2C Namespace

# SpeedsStandardMode Field

Standard Mode

**Namespace:** IO.Interfaces.I2C
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int StandardMode = 100000
```

### Field Value

Type: Int32

## See Also

### Reference

Speeds Class
IO.Interfaces.I2C Namespace

# IO.Interfaces.Message.Text Namespace

Abstract Interface for Text Message Relays

## Interfaces

| | Interface | Description |
|---|---|---|
| | Relay | Abstract interface for a text message relay. |

# Relay Interface

Abstract interface for a text message relay.

**Namespace:** IO.Interfaces.Message.Text
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public interface Relay
```

The Relay type exposes the following members.

## Methods

| | Name | Description |
|---|---|---|
| | Send | Method for sending a text message. |

Top

## See Also

### Reference

IO.Interfaces.Message.Text Namespace

# Relay Methods

The Relay type exposes the following members.

## Methods

| | Name | Description |
|---|---|---|
| | Send | Method for sending a text message. |

Top

## See Also

**Reference**

Relay Interface

IO.Interfaces.Message.Text Namespace

# RelaySend Method

Method for sending a text message.

**Namespace:** IO.Interfaces.Message.Text
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
void Send(
        string sender,
        string recipient,
        string subject,
        string message
)
```

### Parameters

*sender*
Type: SystemString
Message originator.

*recipient*
Type: SystemString
Message recipient.

*subject*
Type: SystemString
Message subject.

*message*
Type: SystemString
Message text.

## See Also

### Reference

Relay Interface

IO.Interfaces.Message.Text Namespace

# IO.Interfaces.Message64 Namespace

Abstract Interface for 64-Byte Message Services

## Classes

| | Class | Description |
|---|---|---|
| | Message | Encapsulates 64-byte messages. |

## Interfaces

| | Interface | Description |
|---|---|---|
| | Messenger | Abstract interface for sending and receiving 64-byte messages. |

# Message Class

Encapsulates 64-byte messages.

## Inheritance Hierarchy

SystemObject IO.Interfaces.Message64Message

**Namespace:** IO.Interfaces.Message64
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public class Message
```

The Message type exposes the following members.

## Constructors

| | Name | Description |
|---|---|---|
| | Message | Create a message object without initializing the payload. |
| | Message(Byte) | Create a message object with an initialized payload. |

Top

## Fields

| | Name | Description |
|---|---|---|
| | payload | Message payload. |
| S | Size | Message payload size. |

Top

## See Also

### Reference

IO.Interfaces.Message64 Namespace

# Message Constructor

## Overload List

| | Name | Description |
|---|---|---|
| | Message | Create a message object without initializing the payload. |
| | Message(Byte) | Create a message object with an initialized payload. |

Top

## See Also

**Reference**

Message Class

IO.Interfaces.Message64 Namespace

# Message Constructor

Create a message object without initializing the payload.

**Namespace:** IO.Interfaces.Message64
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public Message()
```

## See Also

### Reference

Message Class
Message Overload
IO.Interfaces.Message64 Namespace

# Message Constructor (Byte)

Create a message object with an initialized payload.

**Namespace:** IO.Interfaces.Message64
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public Message(
        byte fill
)
```

### Parameters

*fill*

Type: SystemByte
Value to initialize the payload with.

## See Also

### Reference

Message Class
Message Overload
IO.Interfaces.Message64 Namespace

# Message Fields

The Message type exposes the following members.

## Fields

| | Name | Description |
|---|---|---|
| | payload | Message payload. |
| S | Size | Message payload size. |

Top

## See Also

**Reference**

Message Class

IO.Interfaces.Message64 Namespace

# Messagepayload Field

Message payload.

**Namespace:** IO.Interfaces.Message64
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public byte[] payload
```

### Field Value

Type: Byte

## See Also

### Reference

Message Class
IO.Interfaces.Message64 Namespace

# MessageSize Field

Message payload size.

**Namespace:** IO.Interfaces.Message64
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int Size = 64
```

### Field Value

Type: Int32

## See Also

### Reference

Message Class
IO.Interfaces.Message64 Namespace

# Messenger Interface

Abstract interface for sending and receiving 64-byte messages.

**Namespace:** IO.Interfaces.Message64
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public interface Messenger
```

The Messenger type exposes the following members.

## Methods

| | Name | Description |
|---|---|---|
| | Receive | Receive a 64-byte message. |
| | Send | Send a 64-byte message. |
| | Transaction | Send a 64-byte command and receive a 64-byte response. |

Top

## See Also

### Reference

IO.Interfaces.Message64 Namespace

# Messenger Methods

The Messenger type exposes the following members.

## Methods

| | Name | Description |
|---|---|---|
| | Receive | Receive a 64-byte message. |
| | Send | Send a 64-byte message. |
| | Transaction | Send a 64-byte command and receive a 64-byte response. |

Top

## See Also

**Reference**

Messenger Interface
IO.Interfaces.Message64 Namespace

# MessengerReceive Method

Receive a 64-byte message.

**Namespace:** IO.Interfaces.Message64
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
void Receive(
        Message resp
)
```

### Parameters

*resp*
Type: IO.Interfaces.Message64Message
Message received.

## See Also

### Reference

Messenger Interface
IO.Interfaces.Message64 Namespace

# MessengerSend Method

Send a 64-byte message.

**Namespace:** IO.Interfaces.Message64
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

**C#** **VB** **F#** Copy

```
void Send(
        Message cmd
)
```

### Parameters

*cmd*
Type: IO.Interfaces.Message64Message
Message to be sent.

## See Also

### Reference

Messenger Interface
IO.Interfaces.Message64 Namespace

# MessengerTransaction Method

Send a 64-byte command and receive a 64-byte response.

**Namespace:** IO.Interfaces.Message64
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

**C#** **VB** **F#** Copy

```
void Transaction(
        Message cmd,
        Message resp
)
```

### Parameters

*cmd*
Type: IO.Interfaces.Message64Message
Command to be sent.

*resp*
Type: IO.Interfaces.Message64Message
Response to be received.

## See Also

### Reference

Messenger Interface
IO.Interfaces.Message64 Namespace

# IO.Interfaces.Motor Namespace

Abstract Interface For Variable Speed Motor Outputs

## Classes

| | Class | Description |
|---|---|---|
| | Velocities | Motor velocity contants. |

## Interfaces

| | Interface | Description |
|---|---|---|
| | Output | Abstract interface for variable speed motor outputs. |

# Output Interface

Abstract interface for variable speed motor outputs.

**Namespace:** IO.Interfaces.Motor
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public interface Output
```

The Output type exposes the following members.

## Properties

| | Name | Description |
|---|---|---|
| | velocity | Write-only motor velocity property. |

Top

## See Also

### Reference

IO.Interfaces.Motor Namespace

# Output Properties

The Output type exposes the following members.

## Properties

| | Name | Description |
|---|---|---|
| | velocity | Write-only motor velocity property. |

Top

## See Also

**Reference**

Output Interface
IO.Interfaces.Motor Namespace

# Outputvelocity Property

Write-only motor velocity property.

**Namespace:** IO.Interfaces.Motor
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
double velocity { set; }
```

### Property Value

Type: Double

## See Also

### Reference

Output Interface
IO.Interfaces.Motor Namespace

# Velocities Class

Motor velocity contants.

## Inheritance Hierarchy

SystemObject IO.Interfaces.MotorVelocities

**Namespace:** IO.Interfaces.Motor
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static class Velocities
```

The Velocities type exposes the following members.

## Fields

| | Name | Description |
|---|---|---|
| s | Maximum | Maximum velocity (full speed forward). |
| s | Minimum | Minimum velocity (full speed reverse). |
| s | Stop | Zero velocity (motor stopped). |

Top

## See Also

## Reference

IO.Interfaces.Motor Namespace

# Velocities Fields

The Velocities type exposes the following members.

## Fields

| | Name | Description |
|---|---|---|
| | Maximum | Maximum velocity (full speed forward). |
| | Minimum | Minimum velocity (full speed reverse). |
| | Stop | Zero velocity (motor stopped). |

Top

## See Also

**Reference**

Velocities Class

IO.Interfaces.Motor Namespace

# VelocitiesMaximum Field

Maximum velocity (full speed forward).

**Namespace:** IO.Interfaces.Motor
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const double Maximum = 1
```

### Field Value

Type: Double

## See Also

### Reference

Velocities Class
IO.Interfaces.Motor Namespace

# VelocitiesMinimum Field

Minimum velocity (full speed reverse).

**Namespace:** IO.Interfaces.Motor
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const double Minimum = -1
```

### Field Value

Type: Double

## See Also

### Reference

Velocities Class
IO.Interfaces.Motor Namespace

# VelocitiesStop Field

Zero velocity (motor stopped).

**Namespace:** IO.Interfaces.Motor
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const double Stop = 0
```

### Field Value

Type: Double

## See Also

### Reference

Velocities Class
IO.Interfaces.Motor Namespace

# IO.Interfaces.PWM Namespace

Abstract Interface for PWM (Pulse Width Modulated) Outputs

## Classes

| | Class | Description |
|---|---|---|
| | DutyCycles | PWM dutycycle contants. |

## Interfaces

| | Interface | Description |
|---|---|---|
| | Output | Abstract interface for PWM outputs. |

# DutyCycles Class

PWM dutycycle contants.

## Inheritance Hierarchy

SystemObject IO.Interfaces.PWMDutyCycles

**Namespace:** IO.Interfaces.PWM
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static class DutyCycles
```

The DutyCycles type exposes the following members.

## Fields

| | Name | Description |
|---|---|---|
| s | Maximum | Maximum duty cycle (percent). |
| s | Minimum | Minimum duty cycle (percent). |

Top

## See Also

### Reference

IO.Interfaces.PWM Namespace

# DutyCycles Fields

The DutyCycles type exposes the following members.

## Fields

| | Name | Description |
|---|---|---|
| | Maximum | Maximum duty cycle (percent). |
| | Minimum | Minimum duty cycle (percent). |

Top

## See Also

**Reference**

DutyCycles Class
IO.Interfaces.PWM Namespace

# DutyCyclesMaximum Field

Maximum duty cycle (percent).

**Namespace:** IO.Interfaces.PWM
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const double Maximum = 100
```

### Field Value

Type: Double

## See Also

### Reference

DutyCycles Class
IO.Interfaces.PWM Namespace

# DutyCyclesMinimum Field

Minimum duty cycle (percent).

**Namespace:** IO.Interfaces.PWM
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const double Minimum = 0
```

### Field Value

Type: Double

## See Also

### Reference

DutyCycles Class
IO.Interfaces.PWM Namespace

# Output Interface

Abstract interface for PWM outputs.

**Namespace:** IO.Interfaces.PWM
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public interface Output
```

The Output type exposes the following members.

## Properties

| | Name | Description |
|---|---|---|
| | dutycycle | Write-only PWM duty cycle property. |

Top

## See Also

### Reference

IO.Interfaces.PWM Namespace

# Output Properties

The Output type exposes the following members.

## Properties

| | Name | Description |
|---|---|---|
| | dutycycle | Write-only PWM duty cycle property. |

Top

## See Also

**Reference**

Output Interface
IO.Interfaces.PWM Namespace

# Outputdutycycle Property

Write-only PWM duty cycle property.

**Namespace:** IO.Interfaces.PWM
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
double dutycycle { set; }
```

### Property Value

Type: Double

## See Also

### Reference

Output Interface
IO.Interfaces.PWM Namespace

# IO.Interfaces.Servo Namespace

Abstract Interface for Servo Outputs

## Classes

| | Class | Description |
|---|---|---|
| | Positions | Servo position contants. |

## Interfaces

| | Interface | Description |
|---|---|---|
| | Output | Abstract interface for servo outputs. |

# Output Interface

Abstract interface for servo outputs.

**Namespace:** IO.Interfaces.Servo
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public interface Output
```

The Output type exposes the following members.

## Properties

| | Name | Description |
|---|---|---|
| | position | Write-only servo position property. |

Top

## See Also

### Reference

IO.Interfaces.Servo Namespace

# Output Properties

The Output type exposes the following members.

## Properties

| | Name | Description |
|---|---|---|
| | position | Write-only servo position property. |

Top

## See Also

**Reference**

Output Interface
IO.Interfaces.Servo Namespace

# Outputposition Property

Write-only servo position property.

**Namespace:** IO.Interfaces.Servo
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
double position { set; }
```

### Property Value

Type: Double

## See Also

### Reference

Output Interface
IO.Interfaces.Servo Namespace

# Positions Class

Servo position contants.

## Inheritance Hierarchy

SystemObject IO.Interfaces.ServoPositions

**Namespace:** IO.Interfaces.Servo
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static class Positions
```

The Positions type exposes the following members.

## Fields

| | Name | Description |
|---|---|---|
| S | Maximum | Maximum displacement position. |
| S | Minimum | Minimum displacement position. |
| S | Neutral | Zero displacement (neutral) position. |

Top

## See Also

## Reference

IO.Interfaces.Servo Namespace

# Positions Fields

The Positions type exposes the following members.

## Fields

| | Name | Description |
|---|---|---|
| S | Maximum | Maximum displacement position. |
| S | Minimum | Minimum displacement position. |
| S | Neutral | Zero displacement (neutral) position. |

Top

## See Also

**Reference**

Positions Class

IO.Interfaces.Servo Namespace

# PositionsMaximum Field

Maximum displacement position.

**Namespace:** IO.Interfaces.Servo
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const double Maximum = 1
```

### Field Value

Type: Double

## See Also

### Reference

Positions Class
IO.Interfaces.Servo Namespace

# PositionsMinimum Field

Minimum displacement position.

**Namespace:** IO.Interfaces.Servo
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const double Minimum = -1
```

### Field Value

Type: Double

## See Also

### Reference

Positions Class
IO.Interfaces.Servo Namespace

# PositionsNeutral Field

Zero displacement (neutral) position.

**Namespace:** IO.Interfaces.Servo
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const double Neutral = 0
```

### Field Value

Type: Double

## See Also

### Reference

Positions Class
IO.Interfaces.Servo Namespace

# IO.Interfaces.SPI Namespace

Abstract Interface for SPI (Serial Peripheral Interconnect) Slave Devices

## Interfaces

| | Interface | Description |
|---|---|---|
| | Device | Abstract interface for SPI slave devices. |

# Device Interface

Abstract interface for SPI slave devices.

**Namespace:** IO.Interfaces.SPI
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public interface Device
```

The Device type exposes the following members.

## Methods

| | Name | Description |
|---|---|---|
| | Read | Read bytes from an SPI slave device. |
| | Transaction | Write bytes to and read bytes from an SPI slave device. |
| | Write | Write bytes to an SPI slave device. |

Top

## See Also

### Reference

IO.Interfaces.SPI Namespace

# Device Methods

The Device type exposes the following members.

## Methods

| | Name | Description |
|---|---|---|
| | Read | Read bytes from an SPI slave device. |
| | Transaction | Write bytes to and read bytes from an SPI slave device. |
| | Write | Write bytes to an SPI slave device. |

Top

## See Also

**Reference**

Device Interface

IO.Interfaces.SPI Namespace

# DeviceRead Method

Read bytes from an SPI slave device.

**Namespace:** IO.Interfaces.SPI
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
void Read(
        byte[] resp,
        int resplen
)
```

### Parameters

*resp*
Type: SystemByte
Response buffer.

*resplen*
Type: SystemInt32
Number of bytes to read.

## See Also

### Reference

Device Interface
IO.Interfaces.SPI Namespace

# DeviceTransaction Method

Write bytes to and read bytes from an SPI slave device.

**Namespace:** IO.Interfaces.SPI
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
void Transaction(
        byte[] cmd,
        int cmdlen,
        byte[] resp,
        int resplen,
        int delayus = 0
)
```

### Parameters

*cmd*
Type: SystemByte
Command buffer.

*cmdlen*
Type: SystemInt32
Number of bytes to write.

*resp*
Type: SystemByte
Response buffer.

*resplen*
Type: SystemInt32
Number of bytes to read.

*delayus* **(Optional)**

Type: SystemInt32

Delay in microseconds between write and read operations.

## See Also

### Reference

Device Interface

IO.Interfaces.SPI Namespace

# DeviceWrite Method

Write bytes to an SPI slave device.

**Namespace:** IO.Interfaces.SPI
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
void Write(
        byte[] cmd,
        int cmdlen
)
```

### Parameters

*cmd*
Type: SystemByte
Command buffer.

*cmdlen*
Type: SystemInt32
Number of bytes to write.

## See Also

### Reference

Device Interface
IO.Interfaces.SPI Namespace

# IO.Interfaces.Stepper Namespace

Abstract Interface for Stepper Motor Outputs

## Interfaces

| | Interface | Description |
|---|---|---|
| | Output | Abstract interface for stepper motor outputs. |

# Output Interface

Abstract interface for stepper motor outputs.

**Namespace:** IO.Interfaces.Stepper
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public interface Output
```

The Output type exposes the following members.

## Properties

| | Name | Description |
|---|---|---|
| | StepsPerRotation | Read-only property returning the number of steps a stepper motor has. |

Top

## Methods

| | Name | Description |
|---|---|---|
| | Move | Move the stepper motor a specified number of steps at a specified rate. |
| | Spin | Spin (i.e. continuous rotation) the stepper |

moter at a specified rate.

Top

## See Also

### Reference

IO.Interfaces.Stepper Namespace

# Output Properties

The Output type exposes the following members.

## Properties

| | Name | Description |
|---|---|---|
| | StepsPerRotation | Read-only property returning the number of steps a stepper motor has. |

Top

## See Also

**Reference**

Output Interface
IO.Interfaces.Stepper Namespace

# OutputStepsPerRotation Property

Read-only property returning the number of steps a stepper motor has.

**Namespace:** IO.Interfaces.Stepper
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
int StepsPerRotation { get; }
```

### Property Value

Type: Int32

## See Also

### Reference

Output Interface
IO.Interfaces.Stepper Namespace

# Output Methods

The Output type exposes the following members.

## Methods

| | Name | Description |
|---|---|---|
| | Move | Move the stepper motor a specified number of steps at a specified rate. |
| | Spin | Spin (i.e. continuous rotation) the stepper moter at a specified rate. |

Top

## See Also

**Reference**

Output Interface
IO.Interfaces.Stepper Namespace

# OutputMove Method

Move the stepper motor a specified number of steps at a specified rate.

**Namespace:** IO.Interfaces.Stepper
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
void Move(
        int steps,
        float rate
)
```

### Parameters

*steps*
Type: SystemInt32
Number of steps to move. Negative values indicate reverse motion and positive values indicate forward motion. (The directions are nominal and depend on how the stepper motor coils are wired). Zero indicates the stepper motor should be stopped.

*rate*
Type: SystemSingle
The rate of motion, in steps per second. Negative values indicate reverse motion and positive values indicate forward motion. (The directions are nominal and depend on how the stepper motor coils are wired). Zero indicates the stepper motor should be stopped.

## See Also

### Reference

Output Interface

IO.Interfaces.Stepper Namespace

# OutputSpin Method

Spin (i.e. continuous rotation) the stepper moter at a specified rate.

**Namespace:** IO.Interfaces.Stepper
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
void Spin(
        float rate
)
```

### Parameters

*rate*

Type: SystemSingle
The rate of motion, in steps per second. Negative values indicate reverse motion and positive values indicate forward motion. (The directions are nominal and depend on how the stepper motor coils are wired). Zero indicates the stepper motor should be stopped.

## Remarks

Not all stepper motor drivers support continuous rotation.

## See Also

### Reference

Output Interface

# IO.Interfaces.Stepper Namespace

# IO.Interfaces.Temperature Namespace

Abstract Interface for Temperature Sensors

## Classes

| | Class | Description |
|---|---|---|
| | Conversions | Temperature conversion functions. |

## Interfaces

| | Interface | Description |
|---|---|---|
| | Sensor | Abstract interface for temperature sensors. |

# Conversions Class

Temperature conversion functions.

## Inheritance Hierarchy

SystemObject IO.Interfaces.TemperatureConversions

**Namespace:** IO.Interfaces.Temperature
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public class Conversions
```

The Conversions type exposes the following members.

## Constructors

| | Name | Description |
|---|---|---|
| | Conversions | Initializes a new instance of the Conversions class |

Top

## Methods

| | Name | Description |
|---|---|---|
| S | CelsiusToFahrenheit | Convert degrees Celsius to |

| | | |
|---|---|---|
| | | degrees Fahrenheit. |
| | CelsiusToKelvins | Convert degrees Celsius to Kelvins. |
| | FahrenheitToCelsius | Convert degrees Fahrenheit to degrees Celsius. |
| | FahrenheitToKelvins | Convert degrees Fahrenheit to Kelvins. |
| | KelvinsToCelsius | Convert Kelvins to degrees Celsius. |
| | KelvinsToFahrenheit | Convert Kelvins to degrees Fahrenheit. |

Top

## See Also

**Reference**

IO.Interfaces.Temperature Namespace

# Conversions Constructor

Initializes a new instance of the Conversions class

**Namespace:** IO.Interfaces.Temperature
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public Conversions()
```

## See Also

### Reference

Conversions Class
IO.Interfaces.Temperature Namespace

# Conversions Methods

The Conversions type exposes the following members.

## Methods

| | Name | Description |
|---|---|---|
| | CelsiusToFahrenheit | Convert degrees Celsius to degrees Fahrenheit. |
| | CelsiusToKelvins | Convert degrees Celsius to Kelvins. |
| | FahrenheitToCelsius | Convert degrees Fahrenheit to degrees Celsius. |
| | FahrenheitToKelvins | Convert degrees Fahrenheit to Kelvins. |
| | KelvinsToCelsius | Convert Kelvins to degrees Celsius. |
| | KelvinsToFahrenheit | Convert Kelvins to degrees Fahrenheit. |

Top

## See Also

**Reference**

Conversions Class

# IO.Interfaces.Temperature Namespace

# ConversionsCelsiusToFahrenheit Method

Convert degrees Celsius to degrees Fahrenheit.

**Namespace:** IO.Interfaces.Temperature
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static double CelsiusToFahrenheit(
        double celsius
)
```

### Parameters

*celsius*
Type: SystemDouble
Temperature in degrees Celsius.

### Return Value

Type: Double
Temperature in degrees Fahrenheit.

## See Also

### Reference

Conversions Class
IO.Interfaces.Temperature Namespace

# ConversionsCelsiusToKelvins Method

Convert degrees Celsius to Kelvins.

**Namespace:** IO.Interfaces.Temperature
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

**C#** **VB** **F#** Copy

```
public static double CelsiusToKelvins(
        double celsius
)
```

### Parameters

*celsius*
Type: SystemDouble
Temperature in degrees Celsius.

### Return Value

Type: Double
Temperature in Kelvins.

## See Also

### Reference

Conversions Class
IO.Interfaces.Temperature Namespace

# ConversionsFahrenheitToCelsius Method

Convert degrees Fahrenheit to degrees Celsius.

**Namespace:** IO.Interfaces.Temperature
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static double FahrenheitToCelsius(
        double fahrenheit
)
```

### Parameters

*fahrenheit*
Type: SystemDouble
Temperature in degrees Fahrenheit.

### Return Value

Type: Double
Temperature in degrees Celsius.

## See Also

### Reference

Conversions Class
IO.Interfaces.Temperature Namespace

# ConversionsFahrenheitToKelvins Method

Convert degrees Fahrenheit to Kelvins.

**Namespace:** IO.Interfaces.Temperature
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static double FahrenheitToKelvins(
        double fahrenheit
)
```

### Parameters

*fahrenheit*
Type: SystemDouble
Temperature in degrees Fahrenheit.

### Return Value

Type: Double
Temperature in Kelvins.

## See Also

### Reference

Conversions Class
IO.Interfaces.Temperature Namespace

# ConversionsKelvinsToCelsius Method

Convert Kelvins to degrees Celsius.

**Namespace:** IO.Interfaces.Temperature
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static double KelvinsToCelsius(
        double kelvins
)
```

### Parameters

*kelvins*
Type: SystemDouble
Temperature in Kelvins.

### Return Value

Type: Double
Temperature in degrees Celsius.

## See Also

### Reference

Conversions Class
IO.Interfaces.Temperature Namespace

# ConversionsKelvinsToFahrenheit Method

Convert Kelvins to degrees Fahrenheit.

**Namespace:** IO.Interfaces.Temperature
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static double KelvinsToFahrenheit(
        double kelvins
)
```

### Parameters

*kelvins*
Type: SystemDouble
Temperature in Kelivns.

### Return Value

Type: Double
Temperature in degrees Fahrenheit.

## See Also

### Reference

Conversions Class
IO.Interfaces.Temperature Namespace

# Sensor Interface

Abstract interface for temperature sensors.

**Namespace:** IO.Interfaces.Temperature
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public interface Sensor
```

The Sensor type exposes the following members.

## Properties

| | Name | Description |
|---|---|---|
| | Celsius | Read-only property returning the temperature in degrees Celsius. |
| | Fahrenheit | Read-only property returning the temperature in degrees Fahrenheit. |
| | Kelvins | Read-only property returning the temperature in Kelvins. |

Top

## See Also

### Reference

# IO.Interfaces.Temperature Namespace

# Sensor Properties

The Sensor type exposes the following members.

## Properties

| | Name | Description |
|---|---|---|
| | Celsius | Read-only property returning the temperature in degrees Celsius. |
| | Fahrenheit | Read-only property returning the temperature in degrees Fahrenheit. |
| | Kelvins | Read-only property returning the temperature in Kelvins. |

Top

## See Also

**Reference**

Sensor Interface

IO.Interfaces.Temperature Namespace

# SensorCelsius Property

Read-only property returning the temperature in degrees Celsius.

**Namespace:** IO.Interfaces.Temperature
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
double Celsius { get; }
```

### Property Value

Type: Double

## See Also

### Reference

Sensor Interface
IO.Interfaces.Temperature Namespace

# SensorFahrenheit Property

Read-only property returning the temperature in degrees Fahrenheit.

**Namespace:** IO.Interfaces.Temperature
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
double Fahrenheit { get; }
```

### Property Value

Type: Double

## See Also

### Reference

Sensor Interface
IO.Interfaces.Temperature Namespace

# SensorKelvins Property

Read-only property returning the temperature in Kelvins.

**Namespace:** IO.Interfaces.Temperature
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
double Kelvins { get; }
```

### Property Value

Type: Double

## See Also

### Reference

Sensor Interface
IO.Interfaces.Temperature Namespace

# IO.Interfaces.Watchdog Namespace

Abstract Interface for Watchdog Timers

## Interfaces

| | Interface | Description |
|---|---|---|
| | Timer | Abstract interface for watchdog timers. |

# Timer Interface

Abstract interface for watchdog timers.

**Namespace:** IO.Interfaces.Watchdog
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public interface Timer
```

The Timer type exposes the following members.

## Properties

| | Name | Description |
|---|---|---|
| | timeout | Read/Write watchdog timer period property. |

Top

## Methods

| | Name | Description |
|---|---|---|
| | Kick | Reset the watchdog timer. |

Top

## See Also

## Reference

IO.Interfaces.Watchdog Namespace

# Timer Properties

The Timer type exposes the following members.

## Properties

| | Name | Description |
|---|---|---|
| | timeout | Read/Write watchdog timer period property. |

Top

## See Also

**Reference**

Timer Interface

IO.Interfaces.Watchdog Namespace

# Timertimeout Property

Read/Write watchdog timer period property.

**Namespace:** IO.Interfaces.Watchdog
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
int timeout { get; set; }
```

### Property Value

Type: Int32

## See Also

### Reference

Timer Interface
IO.Interfaces.Watchdog Namespace

# Timer Methods

The Timer type exposes the following members.

## Methods

| | Name | Description |
|---|---|---|
| | Kick | Reset the watchdog timer. |

Top

## See Also

**Reference**

Timer Interface

IO.Interfaces.Watchdog Namespace

# TimerKick Method

Reset the watchdog timer.

**Namespace:** IO.Interfaces.Watchdog
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
void Kick()
```

## See Also

### Reference

Timer Interface
IO.Interfaces.Watchdog Namespace

# IO.Objects.Email.Mail Namespace

Email Sending Services using `/usr/bin/mail`

## Classes

| | Class | Description |
|---|---|---|
| | Relay | This class implements the `IO.Interfaces.Messaging.Text.Relay` interface, for sending an email message via the Unix program `/usr/bin/mail`. |

# Relay Class

This class implements the `IO.Interfaces.Messaging.Text.Relay` interface, for sending an email message via the Unix program `/usr/bin/mail`.

## Inheritance Hierarchy

SystemObject IO.Objects.Email.MailRelay

**Namespace:** IO.Objects.Email.Mail
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public class Relay : Relay
```

The Relay type exposes the following members.

## Constructors

| | Name | Description |
|---|---|---|
| | Relay | Constructor for a single mail relay. |

Top

## Methods

| | Name | Description |
|---|---|---|

| | | |
|---|---|---|
| | Send(String, String, String) | Method for sending an email message. |
| | Send(String, String, String, String) | Method for sending an email message. |
| | Send(String, String, String, String, String) | Method for sending an email message with an attachment. |

Top

## See Also

**Reference**

IO.Objects.Email.Mail Namespace

# Relay Constructor

Constructor for a single mail relay.

**Namespace:** IO.Objects.Email.Mail
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public Relay()
```

## See Also

### Reference

Relay Class
IO.Objects.Email.Mail Namespace

# Relay Methods

## Methods

| | Name | Description |
|---|---|---|
| | Send(String, String, String) | Method for sending an email message. |
| | Send(String, String, String, String) | Method for sending an email message. |
| | Send(String, String, String, String, String) | Method for sending an email message with an attachment. |

Top

## See Also

**Reference**

Relay Class

IO.Objects.Email.Mail Namespace

# RelaySend Method

## Overload List

| | Name | Description |
|---|---|---|
| | Send(String, String, String) | Method for sending an email message. |
| | Send(String, String, String, String) | Method for sending an email message. |
| | Send(String, String, String, String, String) | Method for sending an email message with an attachment. |

Top

## See Also

**Reference**

Relay Class

IO.Objects.Email.Mail Namespace

# RelaySend Method (String, String, String)

Method for sending an email message.

**Namespace:** IO.Objects.Email.Mail
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public void Send(
        string recipient,
        string subject,
        string message
)
```

### Parameters

*recipient*
Type: SystemString
Originator email address.

*subject*
Type: SystemString
Recipient email address.

*message*
Type: SystemString
Email message body.

## See Also

**Reference**

Relay Class

Send Overload

IO.Objects.Email.Mail Namespace

# RelaySend Method (String, String, String, String)

Method for sending an email message.

**Namespace:** IO.Objects.Email.Mail
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public void Send(
        string sender,
        string recipient,
        string subject,
        string message
)
```

### Parameters

*sender*
Type: SystemString
Originator email address.

*recipient*
Type: SystemString
Recipient email address.

*subject*
Type: SystemString
Subject of the email message.

*message*

Type: SystemString
Email message body.

### Implements

RelaySend(String, String, String, String)

## See Also

### Reference

Relay Class
Send Overload
IO.Objects.Email.Mail Namespace

# RelaySend Method (String, String, String, String, String)

Method for sending an email message with an attachment.

**Namespace:** IO.Objects.Email.Mail
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public void Send(
        string sender,
        string recipient,
        string subject,
        string message,
        string attachment
)
```

### Parameters

*sender*
Type: SystemString
Originator email address.

*recipient*
Type: SystemString
Recipient email address.

*subject*
Type: SystemString
Subject of the email message.

*message*

Type: SystemString
Email message body.

*attachment*

Type: SystemString
Attachment file name.

## See Also

### Reference

Relay Class
Send Overload
IO.Objects.Email.Mail Namespace

# IO.Objects.Email.SMTP Namespace

Email Sending Services using SMTP to `localhost:25`

## Classes

| | Class | Description |
|---|---|---|
| | Relay | This class implements the `IO.Interfaces.Messaging.Text.Relay` interface, for sending an email message via SMTP (Simple Mail Transfer Protocol). |

# Relay Class

This class implements the `IO.Interfaces.Messaging.Text.Relay` interface, for sending an email message via SMTP (Simple Mail Transfer Protocol).

## Inheritance Hierarchy

SystemObject IO.Objects.Email.SMTPRelay

**Namespace:** IO.Objects.Email.SMTP
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public class Relay : Relay
```

The Relay type exposes the following members.

## Constructors

| | Name | Description |
|---|---|---|
| | Relay | Constructor for a single SMTP mail relay. |

Top

## Methods

| | Name | Description |
|---|---|---|

| | | |
|---|---|---|
| | Send(String, String, String) | Method for sending an email message. |
| | Send(String, String, String, String) | Method for sending an email message. |
| | Send(String, String, String, String, String) | Method for sending an email message with an attachment. |

Top

## See Also

**Reference**

IO.Objects.Email.SMTP Namespace

# Relay Constructor

Constructor for a single SMTP mail relay.

**Namespace:** IO.Objects.Email.SMTP
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public Relay(
        string server = "localhost",
        int port = 25
)
```

### Parameters

*server* **(Optional)**
Type: SystemString
SMTP server domain name or IP addaress.

*port* **(Optional)**
Type: SystemInt32
SMTP server TCP port number.

## See Also

### Reference

Relay Class
IO.Objects.Email.SMTP Namespace

# Relay Methods

## Methods

| | Name | Description |
|---|---|---|
| | Send(String, String, String) | Method for sending an email message. |
| | Send(String, String, String, String) | Method for sending an email message. |
| | Send(String, String, String, String, String) | Method for sending an email message with an attachment. |

Top

## See Also

**Reference**

Relay Class

IO.Objects.Email.SMTP Namespace

# RelaySend Method

## Overload List

| | Name | Description |
|---|---|---|
| | Send(String, String, String) | Method for sending an email message. |
| | Send(String, String, String, String) | Method for sending an email message. |
| | Send(String, String, String, String, String) | Method for sending an email message with an attachment. |

Top

## See Also

**Reference**

Relay Class

IO.Objects.Email.SMTP Namespace

# RelaySend Method (String, String, String)

Method for sending an email message.

**Namespace:** IO.Objects.Email.SMTP
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public void Send(
        string recipient,
        string subject,
        string message
)
```

### Parameters

*recipient*
Type: SystemString
Originator email address.

*subject*
Type: SystemString
Recipient email address.

*message*
Type: SystemString
Email message body.

## See Also

**Reference**

Relay Class

Send Overload

IO.Objects.Email.SMTP Namespace

# RelaySend Method (String, String, String, String)

Method for sending an email message.

**Namespace:** IO.Objects.Email.SMTP
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public void Send(
        string sender,
        string recipient,
        string subject,
        string message
)
```

### Parameters

*sender*
Type: SystemString
Originator email address.

*recipient*
Type: SystemString
Recipient email address.

*subject*
Type: SystemString
Subject of the email message.

*message*

Type: SystemString
Email message body.

### Implements

RelaySend(String, String, String, String)

## See Also

### Reference

Relay Class
Send Overload
IO.Objects.Email.SMTP Namespace

# RelaySend Method (String, String, String, String, String)

Method for sending an email message with an attachment.

**Namespace:** IO.Objects.Email.SMTP
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public void Send(
        string sender,
        string recipient,
        string subject,
        string message,
        string attachment
)
```

### Parameters

*sender*
Type: SystemString
Originator email address.

*recipient*
Type: SystemString
Recipient email address.

*subject*
Type: SystemString
Subject of the email message.

*message*

Type: SystemString
Email message body.

*attachment*

Type: SystemString
Attachment file name.

## See Also

### Reference

Relay Class
Send Overload
IO.Objects.Email.SMTP Namespace

# IO.Objects.GPIO.PWM Namespace

PWM Controlled GPIO Output Services

## Classes

| | Class | Description |
|---|---|---|
| | Pin | Encapsulates GPIO output pins implemented with PWM outputs. |

# Pin Class

Encapsulates GPIO output pins implemented with PWM outputs.

## Inheritance Hierarchy

SystemObject IO.Objects.GPIO.PWMPin

**Namespace:** IO.Objects.GPIO.PWM
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public class Pin : Pin
```

The Pin type exposes the following members.

## Constructors

| | Name | Description |
|---|---|---|
| | Pin | Constructor for a single GPIO pin. |

Top

## Properties

| | Name | Description |
|---|---|---|
| | state | Read/Write GPIO state property. |

Top

## Remarks

Use cases for this class include ON/OFF things like an LED or a solenoid valve driven from a PWM output.

## Remarks

Depending on the PWM hardware implementation, the OFF duty cycle may be slightly greater than 0 % and/or the ON duty cycle may be slightly less than 100 %.

## See Also

**Reference**

IO.Objects.GPIO.PWM Namespace

# Pin Constructor

Constructor for a single GPIO pin.

**Namespace:** IO.Objects.GPIO.PWM
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public Pin(
        Output outp,
        bool state = false,
        double dutycycle = 100
)
```

### Parameters

*outp*
Type: IO.Interfaces.PWMOutput
PWM output instance.

*state* **(Optional)**
Type: SystemBoolean
Initial GPIO output state.

*dutycycle* **(Optional)**
Type: SystemDouble
Initial PWM output duty cycle. Allowed values are 0.0 to 100.0 percent.

## See Also

### Reference

Pin Class
IO.Objects.GPIO.PWM Namespace

# Pin Properties

The Pin type exposes the following members.

## Properties

| | Name | Description |
|---|---|---|
| | state | Read/Write GPIO state property. |

Top

## See Also

**Reference**

Pin Class
IO.Objects.GPIO.PWM Namespace

# Pinstate Property

Read/Write GPIO state property.

**Namespace:** IO.Objects.GPIO.PWM
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public bool state { get; set; }
```

### Property Value

Type: Boolean

### Implements

Pinstate

## See Also

### Reference

Pin Class
IO.Objects.GPIO.PWM Namespace

# IO.Objects.Message64.UDP Namespace

64-Byte Message Services over UDP

## Classes

| | Class | Description |
|---|---|---|
| | Messenger | 64-Byte Message Transport Client Services using UDP (User Datagram Protocol). |

# Messenger Class

64-Byte Message Transport Client Services using UDP (User Datagram Protocol).

## Inheritance Hierarchy

SystemObject IO.Objects.Message64.UDPMessenger

**Namespace:** IO.Objects.Message64.UDP
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public class Messenger : Messenger
```

The Messenger type exposes the following members.

## Constructors

| | Name | Description |
|---|---|---|
| | Messenger | Constructor for a 64-byte Messenger instance using UDP. |

Top

## Methods

| | Name | Description |
|---|---|---|

| | | |
|---|---|---|
| | Receive | Receive a 64-byte response message from a raw HID device. |
| | Send | Send a 64-byte command message to a raw HID device. |
| | Transaction | Send a 64-byte command message and receive a 64-byte response message. |

Top

## See Also

**Reference**

IO.Objects.Message64.UDP Namespace

# Messenger Constructor

Constructor for a 64-byte Messenger instance using UDP.

**Namespace:** IO.Objects.Message64.UDP
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public Messenger(
        string host,
        int port,
        int timeoutms = 1000
)
```

### Parameters

*host*
Type: SystemString
UDP server domain name or IP address.

*port*
Type: SystemInt32
UDP server port number.

*timeoutms* **(Optional)**
Type: SystemInt32
Receive timeout in milliseconds. Zero indicates wait forever.

## See Also

### Reference

Messenger Class

# IO.Objects.Message64.UDP Namespace

# Messenger Methods

The Messenger type exposes the following members.

## Methods

| | Name | Description |
|---|---|---|
| | Receive | Receive a 64-byte response message from a raw HID device. |
| | Send | Send a 64-byte command message to a raw HID device. |
| | Transaction | Send a 64-byte command message and receive a 64-byte response message. |

Top

## See Also

**Reference**

Messenger Class

IO.Objects.Message64.UDP Namespace

# MessengerReceive Method

Receive a 64-byte response message from a raw HID device.

**Namespace:** IO.Objects.Message64.UDP
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

**C#** **VB** **F#** Copy

```
public void Receive(
        Message resp
)
```

### Parameters

*resp*
Type: IO.Interfaces.Message64Message
64-byte response message.

### Implements

MessengerReceive(Message)

## See Also

### Reference

Messenger Class
IO.Objects.Message64.UDP Namespace

# MessengerSend Method

Send a 64-byte command message to a raw HID device.

**Namespace:** IO.Objects.Message64.UDP
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

**C#** **VB** **F#** Copy

```
public void Send(
        Message cmd
)
```

### Parameters

*cmd*
Type: IO.Interfaces.Message64Message
64-byte command message.

### Implements

MessengerSend(Message)

## See Also

### Reference

Messenger Class
IO.Objects.Message64.UDP Namespace

# MessengerTransaction Method

Send a 64-byte command message and receive a 64-byte response message.

**Namespace:** IO.Objects.Message64.UDP
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public void Transaction(
        Message cmd,
        Message resp
)
```

### Parameters

*cmd*
Type: IO.Interfaces.Message64Message
64-byte command message.

*resp*
Type: IO.Interfaces.Message64Message
64-byte response message.

### Implements

MessengerTransaction(Message, Message)

## See Also

### Reference

Messenger Class

# IO.Objects.Message64.UDP Namespace

# IO.Objects.Motor.PWM Namespace

PWM Controlled Motor Services

## Classes

| | Class | Description |
|---|---|---|
| | Output | Encapsulates motors controlled by PWM and GPIO outputs. |

# Output Class

Encapsulates motors controlled by PWM and GPIO outputs.

## Inheritance Hierarchy

SystemObject IO.Objects.Motor.PWMOutput

**Namespace:** IO.Objects.Motor.PWM
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public class Output : Output
```

The Output type exposes the following members.

## Constructors

| | Name | Description |
|---|---|---|
| | Output(Output, Output, Double) | Constructor for a single motor, using two PWM outputs for clockwise and counterclockwise rotation control. |
| | Output(Pin, Output, Double) | Constructor for a single motor, using one GPIO pin for direction control, and one PWM output for speed control. |

Top

## Properties

| | Name | Description |
|---|---|---|
| | velocity | Write-only property for setting the normalized motor velocity. Allowed values are -1.0 (full speed reverse) to +1.0 (full speed forward). |

Top

## See Also

**Reference**

IO.Objects.Motor.PWM Namespace

# Output Constructor

## Overload List

| | Name | Description |
|---|---|---|
| | Output(Output, Output, Double) | Constructor for a single motor, using two PWM outputs for clockwise and counterclockwise rotation control. |
| | Output(Pin, Output, Double) | Constructor for a single motor, using one GPIO pin for direction control, and one PWM output for speed control. |

Top

## See Also

**Reference**

Output Class

IO.Objects.Motor.PWM Namespace

# Output Constructor (Output, Output, Double)

Constructor for a single motor, using two PWM outputs for clockwise and counterclockwise rotation control.

**Namespace:** IO.Objects.Motor.PWM
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public Output(
        Output clockwise,
        Output counterclockwise,
        double velocity = 0
)
```

### Parameters

*clockwise*
Type: IO.Interfaces.PWMOutput
PWM output instance (for clockwise rotation control).

*counterclockwise*
Type: IO.Interfaces.PWMOutput
PWM output instance (for counterclockwise rotation control).

*velocity* **(Optional)**
Type: SystemDouble
Initial motor velocity.

## See Also

### Reference

Output Class
Output Overload
IO.Objects.Motor.PWM Namespace

# Output Constructor (Pin, Output, Double)

Constructor for a single motor, using one GPIO pin for direction control, and one PWM output for speed control.

**Namespace:** IO.Objects.Motor.PWM
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

**C#** **VB** **F#** Copy

```
public Output(
        Pin direction,
        Output speed,
        double velocity = 0
)
```

### Parameters

*direction*
Type: IO.Interfaces.GPIOPin
GPIO pin instance (for direction control).

*speed*
Type: IO.Interfaces.PWMOutput
PWM output instance (for speed control).

*velocity* **(Optional)**
Type: SystemDouble
Initial motor velocity.

## See Also

### Reference

Output Class
Output Overload
IO.Objects.Motor.PWM Namespace

# Output Properties

The Output type exposes the following members.

## Properties

| | Name | Description |
|---|---|---|
| | velocity | Write-only property for setting the normalized motor velocity. Allowed values are -1.0 (full speed reverse) to +1.0 (full speed forward). |

Top

## See Also

**Reference**

Output Class

IO.Objects.Motor.PWM Namespace

# Outputvelocity Property

Write-only property for setting the normalized motor velocity. Allowed values are -1.0 (full speed reverse) to +1.0 (full speed forward).

**Namespace:** IO.Objects.Motor.PWM
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public double velocity { set; }
```

### Property Value

Type: Double

### Implements

Outputvelocity

## See Also

### Reference

Output Class
IO.Objects.Motor.PWM Namespace

# IO.Objects.Motor.Servo Namespace

Servo Controlled Motor (e.g. continuous rotation servo) Services

## Classes

| | Class | Description |
|---|---|---|
| | Output | Encapsulates motors controlled by servo outputs (e.g. continuous rotation servos). |

# Output Class

Encapsulates motors controlled by servo outputs (e.g. continuous rotation servos).

## Inheritance Hierarchy

SystemObject IO.Objects.Motor.ServoOutput

**Namespace:** IO.Objects.Motor.Servo
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public class Output : Output
```

The Output type exposes the following members.

## Constructors

| | Name | Description |
|---|---|---|
| | Output | Constructor for a single motor output. |

Top

## Properties

| | Name | Description |
|---|---|---|
| | velocity | Write-only property for setting the |

| | |
|---|---|
| | normalized motor velocity. Allowed values are -1.0 (full speed reverse) to +1.0 (full speed forward. |

Top

## See Also

### Reference

IO.Objects.Motor.Servo Namespace

# Output Constructor

Constructor for a single motor output.

**Namespace:** IO.Objects.Motor.Servo
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public Output(
        Output servo,
        double velocity = 0
)
```

### Parameters

*servo*
Type: IO.Interfaces.ServoOutput
Servo output instance.

*velocity* **(Optional)**
Type: SystemDouble
Initial motor velocity.

## See Also

### Reference

Output Class
IO.Objects.Motor.Servo Namespace

# Output Properties

The Output type exposes the following members.

## Properties

| | Name | Description |
|---|---|---|
| | velocity | Write-only property for setting the normalized motor velocity. Allowed values are -1.0 (full speed reverse) to +1.0 (full speed forward. |

Top

## See Also

**Reference**

Output Class

IO.Objects.Motor.Servo Namespace

# Outputvelocity Property

Write-only property for setting the normalized motor velocity. Allowed values are -1.0 (full speed reverse) to +1.0 (full speed forward.

**Namespace:** IO.Objects.Motor.Servo
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public double velocity { set; }
```

### Property Value

Type: Double

### Implements

Outputvelocity

## See Also

### Reference

Output Class
IO.Objects.Motor.Servo Namespace

# IO.Objects.RemoteIO Namespace

Remote I/O Device Framework, for sending commands and receiving response to/from Remote I/O Protocol devices.

## Classes

| | Class | Description |
|---|---|---|
| | ADC | Encapsulates remote A/D inputs. |
| | DAC | Encapsulates remote D/A outputs. |
| | Device | Encasulates a remote I/O device. |
| | GPIO | Encapsulates remote GPIO pins. |
| | I2C | Encapsulates remote $I^2C$ buses. |
| | PWM | Encapsulates remote PWM outputs. |
| | SPI | Encapsulates remote SPI slave devices. |

## Enumerations

| | Enumeration | Description |
|---|---|---|
| | MessageTypes | Remote I/O protocol message types |
| | PeripheralTypes | Types of remote peripherals |

# ADC Class

Encapsulates remote A/D inputs.

## Inheritance Hierarchy

SystemObject IO.Objects.RemoteIOADC

**Namespace:** IO.Objects.RemoteIO
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public class ADC : Sample
```

The ADC type exposes the following members.

## Constructors

| | Name | Description |
|---|---|---|
| | ADC | Create a remote A/D input. |

Top

## Properties

| | Name | Description |
|---|---|---|
| | resolution | Read-only property returning the number of bits of resolution. |

| | | |
|---|---|---|
| | sample | Read-only property returning an integer analog input sample. |

Top

## See Also

### Reference

IO.Objects.RemoteIO Namespace

# ADC Constructor

Create a remote A/D input.

**Namespace:** IO.Objects.RemoteIO
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public ADC(
        Device dev,
        int num
)
```

### Parameters

*dev*
Type: IO.Objects.RemoteIODevice
Remote I/O device object.

*num*
Type: SystemInt32
A/D input number: 0 to 127.

## Remarks

Use `Device.ADC_Create()` instead of this constructor.

## See Also

### Reference

ADC Class

# IO.Objects.RemoteIO Namespace

# ADC Properties

The ADC type exposes the following members.

## Properties

| | Name | Description |
|---|---|---|
| | resolution | Read-only property returning the number of bits of resolution. |
| | sample | Read-only property returning an integer analog input sample. |

Top

## See Also

**Reference**

ADC Class

IO.Objects.RemoteIO Namespace

# ADCresolution Property

Read-only property returning the number of bits of resolution.

**Namespace:** IO.Objects.RemoteIO
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public int resolution { get; }
```

### Property Value

Type: Int32

### Implements

Sampleresolution

## See Also

### Reference

ADC Class
IO.Objects.RemoteIO Namespace

# ADCsample Property

Read-only property returning an integer analog input sample.

**Namespace:** IO.Objects.RemoteIO
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public int sample { get; }
```

### Property Value

Type: Int32

### Implements

Samplesample

## See Also

### Reference

ADC Class
IO.Objects.RemoteIO Namespace

# DAC Class

Encapsulates remote D/A outputs.

## Inheritance Hierarchy

SystemObject IO.Objects.RemoteIODAC

**Namespace:** IO.Objects.RemoteIO
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public class DAC : Sample
```

The DAC type exposes the following members.

## Constructors

| | Name | Description |
|---|---|---|
| | DAC | Create a remote D/A output. |

Top

## Properties

| | Name | Description |
|---|---|---|
| | resolution | Read-only property returning the number of bits of resolution. |

| | | |
|---|---|---|
| | sample | Write-only property for writing an integer analog sample to a DAC output. |

Top

## See Also

### Reference

IO.Objects.RemoteIO Namespace

# DAC Constructor

Create a remote D/A output.

**Namespace:** IO.Objects.RemoteIO
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public DAC(
        Device dev,
        int num,
        int sample = 0
)
```

### Parameters

*dev*
Type: IO.Objects.RemoteIODevice
Remote I/O device object.

*num*
Type: SystemInt32
D/A output number: 0 to 127.

*sample* **(Optional)**
Type: SystemInt32
Initial DAC output sample.

## Remarks

Use `Device.DAC_Create()` instead of this constructor.

## See Also

### Reference

DAC Class

IO.Objects.RemoteIO Namespace

# DAC Properties

The DAC type exposes the following members.

## Properties

| | Name | Description |
|---|---|---|
| | resolution | Read-only property returning the number of bits of resolution. |
| | sample | Write-only property for writing an integer analog sample to a DAC output. |

Top

## See Also

**Reference**

DAC Class

IO.Objects.RemoteIO Namespace

# DACresolution Property

Read-only property returning the number of bits of resolution.

**Namespace:** IO.Objects.RemoteIO
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public int resolution { get; }
```

### Property Value

Type: Int32

### Implements

Sampleresolution

## See Also

### Reference

DAC Class
IO.Objects.RemoteIO Namespace

# DACsample Property

Write-only property for writing an integer analog sample to a DAC output.

**Namespace:** IO.Objects.RemoteIO
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public int sample { set; }
```

### Property Value

Type: Int32

### Implements

Samplesample

## See Also

### Reference

DAC Class
IO.Objects.RemoteIO Namespace

# Device Class

Encasulates a remote I/O device.
Encasulates a remote I/O device.

## Inheritance Hierarchy

SystemObject IO.Objects.RemoteIODevice

**Namespace:** IO.Objects.RemoteIO
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public class Device
```

The Device type exposes the following members.

## Constructors

| | Name | Description |
|---|---|---|
| | Device(Messenger) | Create a Remote I/O device object. |
| | Device(Int32, Int32, String, Int32) | Create a Remote I/O Device object instance for the specified USB Raw HID Remote I/O Server. |

Top

## Properties

| | Name | Description |
|---|---|---|
| | Capabilities | Capability string from the Remote I/O device. |
| | Version | Version string from the Remote I/O device. |

Top

## Methods

| | Name | Description |
|---|---|---|
| | ADC_Available | Query available A/D inputs. |
| | ADC_Create | Create a remote A/D input. |
| | DAC_Available | Query available D/A outputs. |
| | DAC_Create | Create a remote D/A output. |
| | Dispatcher | Command dispatcher. |
| | GPIO_Available | Query available GPIO pins. |
| | GPIO_Create | Create a remote GPIO pin object. |
| | I2C_Available | Query available $I^2C$ buses. |
| | I2C_Create | Create a remote $I^2C$ bus controller. |
| | PWM_Available | Query available PWM outputs. |

| | | |
|---|---|---|
| | PWM_Create | Create a remote PWM output. |
| | SPI_Available | Query available SPI slave devices. |
| | SPI_Create | Create a remote SPI slave device. |

Top

## Fields

| | Name | Description |
|---|---|---|
| S | MAX_CHANNELS | Maximum number of channels each subsystem can support. |
| S | Unavailable | Designator for an unavailable channel. |

Top

## See Also

**Reference**

IO.Objects.RemoteIO Namespace

# Device Constructor

## Overload List

| | Name | Description |
|---|---|---|
| | Device(Messenger) | Create a Remote I/O device object. |
| | Device(Int32, Int32, String, Int32) | Create a Remote I/O Device object instance for the specified USB Raw HID Remote I/O Server. |

Top

## See Also

**Reference**

Device Class

IO.Objects.RemoteIO Namespace

# Device Constructor (Messenger)

Create a Remote I/O device object.

**Namespace:** IO.Objects.RemoteIO
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

**C#** **VB** **F#** Copy

```
public Device(
        Messenger m
)
```

### Parameters

*m*
Type: IO.Interfaces.Message64Messenger
Message transport object

## See Also

### Reference

Device Class
Device Overload
IO.Objects.RemoteIO Namespace

# Device Constructor (Int32, Int32, String, Int32)

Create a Remote I/O Device object instance for the specified USB Raw HID Remote I/O Server.

**Namespace:** IO.Objects.RemoteIO
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public Device(
        int VID = 5840,
        int PID = 2810,
        string serialnumber = null,
        int timeoutms = 1000
)
```

### Parameters

*VID* **(Optional)**
Type: SystemInt32
Vendor ID

*PID* **(Optional)**
Type: SystemInt32
Product

*serialnumber* **(Optional)**
Type: SystemString
Serial number

*timeoutms* **(Optional)**

Type: SystemInt32
Time in milliseconds to wait for read and write operations to complete. Zero means wait forever.

## See Also

### Reference

Device Class
Device Overload
IO.Objects.RemoteIO Namespace

# Device Properties

The Device type exposes the following members.

## Properties

| | Name | Description |
|---|---|---|
| | Capabilities | Capability string from the Remote I/O device. |
| | Version | Version string from the Remote I/O device. |

Top

## See Also

**Reference**

Device Class

IO.Objects.RemoteIO Namespace

# DeviceCapabilities Property

Capability string from the Remote I/O device.

**Namespace:** IO.Objects.RemoteIO
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public string Capabilities { get; }
```

### Property Value

Type: String

## See Also

### Reference

Device Class
IO.Objects.RemoteIO Namespace

# DeviceVersion Property

Version string from the Remote I/O device.

**Namespace:** IO.Objects.RemoteIO
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public string Version { get; }
```

### Property Value

Type: String

## See Also

### Reference

Device Class
IO.Objects.RemoteIO Namespace

# Device Methods

The Device type exposes the following members.

## Methods

| | Name | Description |
|---|---|---|
| | ADC_Available | Query available A/D inputs. |
| | ADC_Create | Create a remote A/D input. |
| | DAC_Available | Query available D/A outputs. |
| | DAC_Create | Create a remote D/A output. |
| | Dispatcher | Command dispatcher. |
| | GPIO_Available | Query available GPIO pins. |
| | GPIO_Create | Create a remote GPIO pin object. |
| | I2C_Available | Query available $I^2C$ buses. |
| | I2C_Create | Create a remote $I^2C$ bus controller. |
| | PWM_Available | Query available PWM outputs. |
| | PWM_Create | Create a remote PWM output. |
| | SPI_Available | Query available SPI slave devices. |
| | | |

| | |
|---|---|
| SPI_Create | Create a remote SPI slave device. |

Top

## See Also

### Reference

Device Class
IO.Objects.RemoteIO Namespace

# DeviceADC_Available Method

Query available A/D inputs.

**Namespace:** IO.Objects.RemoteIO
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public List<int> ADC_Available()
```

### Return Value

Type: ListInt32
List of available A/D input numbers.

## See Also

### Reference

Device Class
IO.Objects.RemoteIO Namespace

# DeviceADC_Create Method

Create a remote A/D input.

**Namespace:** IO.Objects.RemoteIO
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public Sample ADC_Create(
        int num
)
```

### Parameters

*num*
Type: SystemInt32
A/D input number: 0 to 127.

### Return Value

Type: Sample
A/D input object.

## See Also

### Reference

Device Class
IO.Objects.RemoteIO Namespace

# DeviceDAC_Available Method

Query available D/A outputs.

**Namespace:** IO.Objects.RemoteIO
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public List<int> DAC_Available()
```

### Return Value

Type: ListInt32
List of available D/A output numbers.

## See Also

### Reference

Device Class
IO.Objects.RemoteIO Namespace

# DeviceDAC_Create Method

Create a remote D/A output.

**Namespace:** IO.Objects.RemoteIO
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public Sample DAC_Create(
        int num,
        int sample = 0
)
```

### Parameters

*num*
Type: SystemInt32
D/A output number: 0 to 127.

*sample* **(Optional)**
Type: SystemInt32
Initial DAC output sample.

### Return Value

Type: Sample
D/A output object.

## See Also

### Reference

Device Class

# IO.Objects.RemoteIO Namespace

# DeviceDispatcher Method

Command dispatcher.

**Namespace:** IO.Objects.RemoteIO
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

**C#** **VB** **F#** Copy

```
public void Dispatcher(
        Message cmd,
        Message resp
)
```

### Parameters

*cmd*
Type: IO.Interfaces.Message64Message
Command to be sent.

*resp*
Type: IO.Interfaces.Message64Message
Response to be received.

## See Also

### Reference

Device Class
IO.Objects.RemoteIO Namespace

# DeviceGPIO_Available Method

Query available GPIO pins.

**Namespace:** IO.Objects.RemoteIO
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public List<int> GPIO_Available()
```

### Return Value

Type: ListInt32
List of available GPIO pin numbers.

## See Also

### Reference

Device Class
IO.Objects.RemoteIO Namespace

# DeviceGPIO_Create Method

Create a remote GPIO pin object.

**Namespace:** IO.Objects.RemoteIO
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public Pin GPIO_Create(
        int num,
        Direction dir,
        bool state = false
)
```

### Parameters

*num*
Type: SystemInt32
GPIO pin number: 0 to 127.

*dir*
Type: IO.Interfaces.GPIODirection
GPIO pin data direction: Input or Output.

*state* **(Optional)**
Type: SystemBoolean
Initial GPIO output state.

### Return Value

Type: Pin
GPIO pin object.

## See Also

### Reference

Device Class

IO.Objects.RemoteIO Namespace

# DeviceI2C_Available Method

Query available I²C buses.

**Namespace:** IO.Objects.RemoteIO
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public List<int> I2C_Available()
```

### Return Value

Type: ListInt32
List of available I²C bus numbers.

## See Also

### Reference

Device Class
IO.Objects.RemoteIO Namespace

# DeviceI2C_Create Method

Create a remote I²C bus controller.

**Namespace:** IO.Objects.RemoteIO
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public Bus I2C_Create(
        int num,
        int speed = 100000
)
```

### Parameters

*num*
Type: SystemInt32
I²C bus number: 0 to 127.

*speed* **(Optional)**
Type: SystemInt32
I²C bus clock frequency in Hz

### Return Value

Type: Bus
I²C bus controller object.

## See Also

### Reference

Device Class

# IO.Objects.RemoteIO Namespace

# DevicePWM_Available Method

Query available PWM outputs.

**Namespace:** IO.Objects.RemoteIO
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public List<int> PWM_Available()
```

### Return Value

Type: ListInt32
List of available PWM output numbers.

## See Also

### Reference

Device Class
IO.Objects.RemoteIO Namespace

# DevicePWM_Create Method

Create a remote PWM output.

**Namespace:** IO.Objects.RemoteIO
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public Output PWM_Create(
        int num,
        int freq,
        double duty = 0
)
```

### Parameters

*num*
Type: SystemInt32
PWM output number: 0 to 127.

*freq*
Type: SystemInt32
PWM pulse frequency in Hz.

*duty* **(Optional)**
Type: SystemDouble
Initial PWM output duty cycle.

### Return Value

Type: Output
PWM output object.

## See Also

### Reference

Device Class

IO.Objects.RemoteIO Namespace

# DeviceSPI_Available Method

Query available SPI slave devices.

**Namespace:** IO.Objects.RemoteIO
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public List<int> SPI_Available()
```

### Return Value

Type: ListInt32
List of available SPI slave device numbers.

## See Also

### Reference

Device Class
IO.Objects.RemoteIO Namespace

# DeviceSPI_Create Method

Create a remote SPI slave device.

**Namespace:** IO.Objects.RemoteIO
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public Device SPI_Create(
        int num,
        int mode,
        int wordsize,
        int speed
)
```

### Parameters

*num*
Type: SystemInt32
SPI slave device number: 0 to 127.

*mode*
Type: SystemInt32
SPI transfer mode: 0 to 3.

*wordsize*
Type: SystemInt32
SPI transfer word size: 8, 16, or 32.

*speed*
Type: SystemInt32
SPI transfer speed in bits per second.

### Return Value

Type: Device

SPI slave device object.

## Remarks

The actual SPI transfer rate will be the highest realizable rate that does not exceed the value specified in `speed`.

## See Also

### Reference

Device Class

IO.Objects.RemoteIO Namespace

# Device Fields

The Device type exposes the following members.

## Fields

| | Name | Description |
|---|---|---|
| | MAX_CHANNELS | Maximum number of channels each subsystem can support. |
| | Unavailable | Designator for an unavailable channel. |

Top

## See Also

**Reference**

Device Class

IO.Objects.RemoteIO Namespace

# DeviceMAX_CHANNELS Field

Maximum number of channels each subsystem can support.

**Namespace:** IO.Objects.RemoteIO
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int MAX_CHANNELS = 128
```

### Field Value

Type: Int32

## See Also

### Reference

Device Class
IO.Objects.RemoteIO Namespace

# DeviceUnavailable Field

Designator for an unavailable channel.

**Namespace:** IO.Objects.RemoteIO
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int Unavailable = -1
```

### Field Value

Type: Int32

## See Also

### Reference

Device Class
IO.Objects.RemoteIO Namespace

# GPIO Class

Encapsulates remote GPIO pins.

## Inheritance Hierarchy

SystemObject IO.Objects.RemoteIOGPIO

**Namespace:** IO.Objects.RemoteIO
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public class GPIO : Pin
```

The GPIO type exposes the following members.

## Constructors

| | Name | Description |
|---|---|---|
| | GPIO | Create a remote GPIO pin. |

Top

## Properties

| | Name | Description |
|---|---|---|
| | state | Read/Write GPIO state property. |

Top

## See Also

### Reference

IO.Objects.RemoteIO Namespace

# GPIO Constructor

Create a remote GPIO pin.

**Namespace:** IO.Objects.RemoteIO
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public GPIO(
        Device dev,
        int num,
        Direction dir,
        bool state = false
)
```

### Parameters

*dev*
Type: IO.Objects.RemoteIODevice
Remote I/O device object.

*num*
Type: SystemInt32
GPIO pin number: 0 to 127.

*dir*
Type: IO.Interfaces.GPIODirection
GPIO pin data direction: Input or Output.

*state* **(Optional)**
Type: SystemBoolean
Initial GPIO output state.

## Remarks

Use `Device.GPIO_Create()` instead of this constructor.

## See Also

### Reference

GPIO Class
IO.Objects.RemoteIO Namespace

# GPIO Properties

The GPIO type exposes the following members.

## Properties

| | Name | Description |
|---|---|---|
| | state | Read/Write GPIO state property. |

Top

## See Also

**Reference**

GPIO Class

IO.Objects.RemoteIO Namespace

# GPIOstate Property

Read/Write GPIO state property.

**Namespace:** IO.Objects.RemoteIO
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public bool state { get; set; }
```

### Property Value

Type: Boolean

### Implements

Pinstate

## See Also

### Reference

GPIO Class
IO.Objects.RemoteIO Namespace

# I2C Class

Encapsulates remote I²C buses.

## Inheritance Hierarchy

SystemObject IO.Objects.RemoteIOI2C

**Namespace:** IO.Objects.RemoteIO
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public class I2C : Bus
```

The I2C type exposes the following members.

## Constructors

| | Name | Description |
|---|---|---|
| | I2C | Create a remote I²C bus controller. |

Top

## Methods

| | Name | Description |
|---|---|---|
| | Read | Read bytes from an I²C slave device. |
| | | |

| | | |
|---|---|---|
| | Transaction | Write and read bytes to and from an $I^2C$ slave device. |
| | Write | Write bytes to an $I^2C$ slave device. |

Top

## See Also

### Reference

IO.Objects.RemoteIO Namespace

# I2C Constructor

Create a remote I²C bus controller.

**Namespace:** IO.Objects.RemoteIO
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public I2C(
        Device dev,
        int num,
        int speed = 100000
)
```

### Parameters

*dev*
Type: IO.Objects.RemoteIODevice
Remote I/O device object.

*num*
Type: SystemInt32
I²C bus number: 0 to 127.

*speed* **(Optional)**
Type: SystemInt32
I²C bus clock frequency in Hz

## Remarks

Use `Device.I2C_Create()` instead of this constructor.

## See Also

### Reference

I2C Class

IO.Objects.RemoteIO Namespace

# I2C Methods

The I2C type exposes the following members.

## Methods

| | Name | Description |
|---|---|---|
| | Read | Read bytes from an $I^2C$ slave device. |
| | Transaction | Write and read bytes to and from an $I^2C$ slave device. |
| | Write | Write bytes to an $I^2C$ slave device. |

Top

## See Also

**Reference**

I2C Class

IO.Objects.RemoteIO Namespace

# I2CRead Method

Read bytes from an I²C slave device.

**Namespace:** IO.Objects.RemoteIO
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public void Read(
        int slaveaddr,
        byte[] resp,
        int resplen
)
```

### Parameters

*slaveaddr*
Type: SystemInt32
I²C slave address.

*resp*
Type: SystemByte
Response buffer.

*resplen*
Type: SystemInt32
Number of bytes to read.

### Implements

BusRead(Int32, Byte, Int32)

## See Also

### Reference

I2C Class

IO.Objects.RemoteIO Namespace

# I2CTransaction Method

Write and read bytes to and from an I²C slave device.

**Namespace:** IO.Objects.RemoteIO
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public void Transaction(
        int slaveaddr,
        byte[] cmd,
        int cmdlen,
        byte[] resp,
        int resplen,
        int delayus
)
```

### Parameters

*slaveaddr*
Type: SystemInt32
I²C slave address.

*cmd*
Type: SystemByte
Command buffer.

*cmdlen*
Type: SystemInt32
Number of bytes to write.

*resp*

Type: SystemByte
Response buffer.

*resplen*

Type: SystemInt32
Number of bytes to read.

*delayus*

Type: SystemInt32
Delay in microseconds between the $I^2C$ write and read cycles.
Allowed values are 0 to 65535 microseconds.

### Implements

BusTransaction(Int32, Byte, Int32, Byte, Int32, Int32)

## See Also

### Reference

I2C Class
IO.Objects.RemoteIO Namespace

# I2CWrite Method

Write bytes to an I²C slave device.

**Namespace:** IO.Objects.RemoteIO
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public void Write(
        int slaveaddr,
        byte[] cmd,
        int cmdlen
)
```

### Parameters

*slaveaddr*
Type: SystemInt32
I²C slave address.

*cmd*
Type: SystemByte
Command buffer.

*cmdlen*
Type: SystemInt32
Number of bytes to write.

### Implements

BusWrite(Int32, Byte, Int32)

## See Also

### Reference

I2C Class

IO.Objects.RemoteIO Namespace

# MessageTypes Enumeration

Remote I/O protocol message types

**Namespace:** IO.Objects.RemoteIO
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public enum MessageTypes
```

## Members

| Member name | Value | Description |
|---|---|---|
| LOOPBACK_REQUEST | 0 | Loopback request |
| LOOPBACK_RESPONSE | 1 | Loopback response |
| VERSION_REQUEST | 2 | Version string request |
| VERSION_RESPONSE | 3 | Version string response |
| CAPABILITY_REQUEST | 4 | Capability |

| | | | |
|---|---|---|---|
| | | | string request |
| | CAPABILITY_RESPONSE | 5 | Capability string response |
| | GPIO_PRESENT_REQUEST | 6 | GPIO pins available request |
| | GPIO_PRESENT_RESPONSE | 7 | GPIO pins available response |
| | GPIO_CONFIGURE_REQUEST | 8 | GPIO pins configure request |
| | GPIO_CONFIGURE_RESPONSE | 9 | GPIO pins configure response |
| | GPIO_READ_REQUEST | 10 | GPIO pins read request |
| | GPIO_READ_RESPONSE | 11 | GPIO pins read response |
| | GPIO_WRITE_REQUEST | 12 | GPIO pins write request |

| | | | |
|---|---|---|---|
| | GPIO_WRITE_RESPONSE | 13 | GPIO pins write response |
| | I2C_PRESENT_REQUEST | 14 | $I^2C$ buses available request |
| | I2C_PRESENT_RESPONSE | 15 | $I^2C$ buses available response |
| | I2C_CONFIGURE_REQUEST | 16 | $I^2C$ bus configure request |
| | I2C_CONFIGURE_RESPONSE | 17 | $I^2C$ bus configure response |
| | I2C_TRANSACTION_REQUEST | 18 | $I^2C$ bus transaction request |
| | I2C_TRANSACTION_RESPONSE | 19 | $I^2C$ bus transaction response |
| | SPI_PRESENT_REQUEST | 20 | SPI slave devices available request |
| | SPI_PRESENT_RESPONSE | 21 | SPI slave devices |

| | | | |
|---|---|---|---|
| | | | available response |
| | SPI_CONFIGURE_REQUEST | 22 | SPI slave device configure request |
| | SPI_CONFIGURE_RESPONSE | 23 | SPI slave device configure response |
| | SPI_TRANSACTION_REQUEST | 24 | SPI bus transaction request |
| | SPI_TRANSACTION_RESPONSE | 25 | SPI bus transaction response |
| | ADC_PRESENT_REQUEST | 26 | ADC inputs available request |
| | ADC_PRESENT_RESPONSE | 27 | ADC inputs available response |
| | ADC_CONFIGURE_REQUEST | 28 | ADC input configure request |
| | ADC_CONFIGURE_RESPONSE | 29 | ADC input configure |

| | | | |
|---|---|---|---|
| | | | response |
| | ADC_READ_REQUEST | 30 | ADC input read request |
| | ADC_READ_RESPONSE | 31 | ADC input read response |
| | DAC_PRESENT_REQUEST | 32 | DAC outputs available request |
| | DAC_PRESENT_RESPONSE | 33 | DAC outputs available response |
| | DAC_CONFIGURE_REQUEST | 34 | DAC input configure request |
| | DAC_CONFIGURE_RESPONSE | 35 | DAC input configure response |
| | DAC_WRITE_REQUEST | 36 | DAC output write request |
| | DAC_WRITE_RESPONSE | 37 | DAC output write response |

| | | | |
|---|---|---|---|
| | PWM_PRESENT_REQUEST | 38 | PWM outputs available request |
| | PWM_PRESENT_RESPONSE | 39 | PWM outputs available response |
| | PWM_CONFIGURE_REQUEST | 40 | PWM input configure request |
| | PWM_CONFIGURE_RESPONSE | 41 | PWM input configure response |
| | PWM_WRITE_REQUEST | 42 | PWM output write request |
| | PWM_WRITE_RESPONSE | 43 | PWM output write response |

## See Also

### Reference

IO.Objects.RemoteIO Namespace

# PeripheralTypes Enumeration

Types of remote peripherals

**Namespace:** IO.Objects.RemoteIO
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public enum PeripheralTypes
```

## Members

| Member name | Value | Description |
|---|---|---|
| ADC | 0 | A/D inputs |
| DAC | 1 | D/A outputs |
| GPIO | 2 | GPIO pins |
| I2C | 3 | $I^2C$ bus controllers |
| PWM | 4 | SPI slave devices |
| SPI | 5 | PWM outputs |

## See Also

### Reference

IO.Objects.RemoteIO Namespace

# PWM Class

Encapsulates remote PWM outputs.

## Inheritance Hierarchy

SystemObject IO.Objects.RemoteIOPWM

**Namespace:** IO.Objects.RemoteIO
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public class PWM : Output
```

The PWM type exposes the following members.

## Constructors

| | Name | Description |
|---|---|---|
| | PWM | Create a remote PWM output. |

Top

## Properties

| | Name | Description |
|---|---|---|
| | dutycycle | Write-only property for setting the PWM output duty cycle. |

Top

## See Also

### Reference

IO.Objects.RemoteIO Namespace

# PWM Constructor

Create a remote PWM output.

**Namespace:** IO.Objects.RemoteIO
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public PWM(
        Device dev,
        int num,
        int freq,
        double duty = 0
)
```

### Parameters

*dev*
Type: IO.Objects.RemoteIODevice
Remote I/O device object.

*num*
Type: SystemInt32
PWM output number: 0 to 127.

*freq*
Type: SystemInt32
PWM pulse frequency in Hz.

*duty* **(Optional)**
Type: SystemDouble
Initial PWM output duty cycle.

## Remarks

Use `Device.PWM_Create()` instead of this constructor.

## See Also

### Reference

PWM Class
IO.Objects.RemoteIO Namespace

# PWM Properties

The PWM type exposes the following members.

## Properties

| | Name | Description |
|---|---|---|
| | dutycycle | Write-only property for setting the PWM output duty cycle. |

Top

## See Also

**Reference**

PWM Class

IO.Objects.RemoteIO Namespace

# PWMdutycycle Property

Write-only property for setting the PWM output duty cycle.

**Namespace:** IO.Objects.RemoteIO
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public double dutycycle { set; }
```

### Property Value

Type: Double

### Implements

Outputdutycycle

## See Also

### Reference

PWM Class
IO.Objects.RemoteIO Namespace

# SPI Class

Encapsulates remote SPI slave devices.

## Inheritance Hierarchy

SystemObject IO.Objects.RemoteIOSPI

**Namespace:** IO.Objects.RemoteIO
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public class SPI : Device
```

The SPI type exposes the following members.

## Constructors

| | Name | Description |
|---|---|---|
| | SPI | Create a remote SPI slave device. |

Top

## Methods

| | Name | Description |
|---|---|---|
| | Read | Read bytes from an SPI slave device. |
| | | |

| | | |
|---|---|---|
| | Transaction | Write and read bytes to and from an SPI slave device. |
| | Write | Write bytes to an SPI slave device. |

Top

## See Also

### Reference

IO.Objects.RemoteIO Namespace

# SPI Constructor

Create a remote SPI slave device.

**Namespace:** IO.Objects.RemoteIO
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public SPI(
        Device dev,
        int num,
        int mode,
        int wordsize,
        int speed
)
```

### Parameters

*dev*
Type: IO.Objects.RemoteIODevice
Remote I/O device object.

*num*
Type: SystemInt32
SPI slave device number: 0 to 127.

*mode*
Type: SystemInt32
SPI transfer mode: 0 to 3.

*wordsize*
Type: SystemInt32
SPI transfer word size: 8, 16, or 32.

*speed*

Type: SystemInt32

SPI transfer speed in bits per second.

## Remarks

Use `Device.SPI_Create()` instead of this constructor.

The actual SPI transfer rate will be the highest realizable rate that does not exceed the value specified in `speed`.

## See Also

### Reference

SPI Class

IO.Objects.RemoteIO Namespace

# SPI Methods

The SPI type exposes the following members.

## Methods

| | Name | Description |
|---|---|---|
| | Read | Read bytes from an SPI slave device. |
| | Transaction | Write and read bytes to and from an SPI slave device. |
| | Write | Write bytes to an SPI slave device. |

Top

## See Also

**Reference**

SPI Class

IO.Objects.RemoteIO Namespace

# SPIRead Method

Read bytes from an SPI slave device.

**Namespace:** IO.Objects.RemoteIO
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public void Read(
        byte[] resp,
        int resplen
)
```

### Parameters

*resp*
Type: SystemByte
Response buffer.

*resplen*
Type: SystemInt32
Number of bytes to read: 1 to 60.

### Implements

DeviceRead(Byte, Int32)

## See Also

### Reference

SPI Class
IO.Objects.RemoteIO Namespace

# SPITransaction Method

Write and read bytes to and from an SPI slave device.

**Namespace:** IO.Objects.RemoteIO
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public void Transaction(
        byte[] cmd,
        int cmdlen,
        byte[] resp,
        int resplen,
        int delayus = 0
)
```

### Parameters

*cmd*
Type: SystemByte
Command buffer.

*cmdlen*
Type: SystemInt32
Number of bytes to write: 0 to 57.

*resp*
Type: SystemByte
Response buffer.

*resplen*
Type: SystemInt32
Number of bytes to read: 0 to 60.

*delayus* **(Optional)**

Type: SystemInt32

Delay in microseconds between write and read operations: 0 to 65535.

### Implements

DeviceTransaction(Byte, Int32, Byte, Int32, Int32)

## See Also

### Reference

SPI Class

IO.Objects.RemoteIO Namespace

# SPIWrite Method

Write bytes to an SPI slave device.

**Namespace:** IO.Objects.RemoteIO
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public void Write(
        byte[] cmd,
        int cmdLen
)
```

### Parameters

*cmd*
Type: SystemByte
Command buffer.

*cmdlen*
Type: SystemInt32
Number of bytes to write: 1 to 57.

### Implements

DeviceWrite(Byte, Int32)

## See Also

### Reference

SPI Class
IO.Objects.RemoteIO Namespace

# IO.Objects.RemoteIO.mikroBUS Namespace

Mikroelektronika mikroBUS (https://www.mikroe.com/mikrobus) Remote I/O protocol Server and Socket Services

## Classes

| | Class | Description |
|---|---|---|
| | Shield | Encapsulates mikroBUS shields on Remote I/O Protocol servers providing mikroBUS sockets). |
| | Socket | Encapsulates mikroBUS sockets. |

## Enumerations

| | Enumeration | Description |
|---|---|---|
| | ShieldKinds | Supported mikroBUS shields. |

# Shield Class

Encapsulates mikroBUS shields on Remote I/O Protocol servers providing mikroBUS sockets).

## Inheritance Hierarchy

SystemObject IO.Objects.RemoteIO.mikroBUSShield

**Namespace:** IO.Objects.RemoteIO.mikroBUS
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static class Shield
```

The Shield type exposes the following members.

## Properties

| | Name | Description |
|---|---|---|
| | kind | Returns the kind of mikroBUS shield that is installed on the Remote I/O Protocol server, as obtained from the `SHIELDNAME` environment variable. |

Top

## Fields

| | Name | Description |
|---|---|---|
| | I2CBus | Shared $I^2C$ bus that is common to all sockets on this shield. |

Top

## See Also

### Reference

IO.Objects.RemoteIO.mikroBUS Namespace

# Shield Properties

The Shield type exposes the following members.

## Properties

| | Name | Description |
|---|---|---|
| | kind | Returns the kind of mikroBUS shield that is installed on the Remote I/O Protocol server, as obtained from the `SHIELDNAME` environment variable. |

Top

## See Also

**Reference**

Shield Class

IO.Objects.RemoteIO.mikroBUS Namespace

# Shieldkind Property

Returns the kind of mikroBUS shield that is installed on the Remote I/O Protocol server, as obtained from the `SHIELDNAME` environment variable.

**Namespace:** IO.Objects.RemoteIO.mikroBUS
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static ShieldKinds kind { get; }
```

### Property Value

Type: ShieldKinds

## See Also

### Reference

Shield Class
IO.Objects.RemoteIO.mikroBUS Namespace

# Shield Fields

The Shield type exposes the following members.

## Fields

| | Name | Description |
|---|---|---|
| | I2CBus | Shared $I^2C$ bus that is common to all sockets on this shield. |

Top

## See Also

**Reference**

Shield Class

IO.Objects.RemoteIO.mikroBUS Namespace

# ShieldI2CBus Field

Shared $I^2C$ bus that is common to all sockets on this shield.

**Namespace:** IO.Objects.RemoteIO.mikroBUS
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static Bus I2CBus
```

### Field Value

Type: Bus

## See Also

### Reference

Shield Class
IO.Objects.RemoteIO.mikroBUS Namespace

# ShieldKinds Enumeration

Supported mikroBUS shields.

**Namespace:** IO.Objects.RemoteIO.mikroBUS
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public enum Kinds
```

## Members

| Member name | Value | Description |
|---|---|---|
| Clicker | 0 | Mikroelektronika STM32F4 Clicker with MUNTS-0011 Remote I/O Server firmware, with one mikroBUS socket. |
| PiClick1 | 1 | Raspberry Pi with Mikroelektronika Pi Click Shield MIKROE-1512/1513 for 26-pin expansion header, with one mikroBUS socket (Obsolete.) |

| | | | |
|---|---|---|---|
| | PiClick2 | 2 | Raspberry Pi with Mikroelektronika Pi 2 Click Shield MIKROE-1879 for 40-pin expansion header, with two mikroBUS sockets. |
| | PiClick3 | 3 | Mikroelektronika Pi 3 Click Shield MIKROE-2756 for 40-pin expansion header, with selectable on-board A/D converter and two mikroBUS sockets. |
| | PocketBeagle | 4 | PocketBeagle with female headers on top, with two mikroBUS sockets. |
| | Unknown | 2147483647 | No known mikroBUS shield installed. |

## See Also

### Reference

IO.Objects.RemoteIO.mikroBUS Namespace

# Socket Class

Encapsulates mikroBUS sockets.

## Inheritance Hierarchy

SystemObject IO.Objects.RemoteIO.mikroBUSSocket

**Namespace:** IO.Objects.RemoteIO.mikroBUS
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public class Socket
```

The Socket type exposes the following members.

## Constructors

| | Name | Description |
|---|---|---|
| | Socket | Constructor for a single mikroBUS socket. |

Top

## Properties

| | Name | Description |
|---|---|---|
| | AIN | Returns the ADC input designator for AN. |

| | | |
|---|---|---|
| | AN | Returns the GPIO pin designator for AN. |
| | CS | Returns the GPIO pin designator for CS. |
| | I2CBus | Returns the $I^2C$ bus designator for this socket. |
| | INT | Returns the GPIO pin designator for INT. |
| | MISO | Returns the GPIO pin designator for MISO. |
| | MOSI | Returns the GPIO pin designator for MOSI. |
| | PWM | Returns the GPIO pin designator for PWM. |
| | PWMOut | Returns the PWM output designator for PWM. |
| | RST | Returns the GPIO pin designator for RST. |
| | RX | Returns the GPIO pin designator for RX. |
| | SCK | Returns the GPIO pin designator for SCK. |
| | SCL | Returns the GPIO pin designator for SCL. |
| | SDA | Returns the GPIO pin designator for SDA. |
| | SPIDev | Returns the SPI device designator for |

| | | |
|---|---|---|
| | | this socket. |
| | TX | Returns the GPIO pin designator for TX. |

Top

## See Also

### Reference

IO.Objects.RemoteIO.mikroBUS Namespace

# Socket Constructor

Constructor for a single mikroBUS socket.

**Namespace:** IO.Objects.RemoteIO.mikroBUS
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public Socket(
        int num,
        ShieldKinds shield = ShieldKinds.Unknown
)
```

### Parameters

*num*
Type: SystemInt32
Socket number.

*shield* **(Optional)**
Type: IO.Objects.RemoteIO.mikroBUSShieldKinds
mikroBUS shield kind. Zero indicates automatic detection using the `Shield.kind` property.

## See Also

### Reference

Socket Class
IO.Objects.RemoteIO.mikroBUS Namespace

# Socket Properties

The Socket type exposes the following members.

## Properties

| | Name | Description |
|---|---|---|
| | AIN | Returns the ADC input designator for AN. |
| | AN | Returns the GPIO pin designator for AN. |
| | CS | Returns the GPIO pin designator for CS. |
| | I2CBus | Returns the $I^2C$ bus designator for this socket. |
| | INT | Returns the GPIO pin designator for INT. |
| | MISO | Returns the GPIO pin designator for MISO. |
| | MOSI | Returns the GPIO pin designator for MOSI. |
| | PWM | Returns the GPIO pin designator for PWM. |
| | PWMOut | Returns the PWM output designator for PWM. |
| | | |

| | | |
|---|---|---|
| | RST | Returns the GPIO pin designator for RST. |
| | RX | Returns the GPIO pin designator for RX. |
| | SCK | Returns the GPIO pin designator for SCK. |
| | SCL | Returns the GPIO pin designator for SCL. |
| | SDA | Returns the GPIO pin designator for SDA. |
| | SPIDev | Returns the SPI device designator for this socket. |
| | TX | Returns the GPIO pin designator for TX. |

Top

# See Also

**Reference**

Socket Class

IO.Objects.RemoteIO.mikroBUS Namespace

# SocketAIN Property

Returns the ADC input designator for AN.

**Namespace:** IO.Objects.RemoteIO.mikroBUS
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public int AIN { get; }
```

### Property Value

Type: Int32

## See Also

### Reference

Socket Class
IO.Objects.RemoteIO.mikroBUS Namespace

# SocketAN Property

Returns the GPIO pin designator for AN.

**Namespace:** IO.Objects.RemoteIO.mikroBUS
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public int AN { get; }
```

### Property Value

Type: Int32

## See Also

### Reference

Socket Class
IO.Objects.RemoteIO.mikroBUS Namespace

# SocketCS Property

Returns the GPIO pin designator for CS.

**Namespace:** IO.Objects.RemoteIO.mikroBUS
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public int CS { get; }
```

### Property Value

Type: Int32

## See Also

### Reference

Socket Class
IO.Objects.RemoteIO.mikroBUS Namespace

# SocketI2CBus Property

Returns the I$^2$C bus designator for this socket.

**Namespace:** IO.Objects.RemoteIO.mikroBUS
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public int I2CBus { get; }
```

### Property Value

Type: Int32

## See Also

### Reference

Socket Class
IO.Objects.RemoteIO.mikroBUS Namespace

# SocketINT Property

Returns the GPIO pin designator for INT.

**Namespace:** IO.Objects.RemoteIO.mikroBUS
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public int INT { get; }
```

### Property Value

Type: Int32

## See Also

### Reference

Socket Class
IO.Objects.RemoteIO.mikroBUS Namespace

# SocketMISO Property

Returns the GPIO pin designator for MISO.

**Namespace:** IO.Objects.RemoteIO.mikroBUS
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public int MISO { get; }
```

### Property Value

Type: Int32

## See Also

### Reference

Socket Class
IO.Objects.RemoteIO.mikroBUS Namespace

# SocketMOSI Property

Returns the GPIO pin designator for MOSI.

**Namespace:** IO.Objects.RemoteIO.mikroBUS
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public int MOSI { get; }
```

### Property Value

Type: Int32

## See Also

### Reference

Socket Class
IO.Objects.RemoteIO.mikroBUS Namespace

# SocketPWM Property

Returns the GPIO pin designator for PWM.

**Namespace:** IO.Objects.RemoteIO.mikroBUS
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public int PWM { get; }
```

### Property Value

Type: Int32

## See Also

### Reference

Socket Class
IO.Objects.RemoteIO.mikroBUS Namespace

# SocketPWMOut Property

Returns the PWM output designator for PWM.

**Namespace:** IO.Objects.RemoteIO.mikroBUS
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public int PWMOut { get; }
```

### Property Value

Type: Int32

## See Also

### Reference

Socket Class
IO.Objects.RemoteIO.mikroBUS Namespace

# SocketRST Property

Returns the GPIO pin designator for RST.

**Namespace:** IO.Objects.RemoteIO.mikroBUS
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public int RST { get; }
```

### Property Value

Type: Int32

## See Also

### Reference

Socket Class
IO.Objects.RemoteIO.mikroBUS Namespace

# SocketRX Property

Returns the GPIO pin designator for RX.

**Namespace:** IO.Objects.RemoteIO.mikroBUS
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public int RX { get; }
```

### Property Value

Type: Int32

## See Also

### Reference

Socket Class
IO.Objects.RemoteIO.mikroBUS Namespace

# SocketSCK Property

Returns the GPIO pin designator for SCK.

**Namespace:** IO.Objects.RemoteIO.mikroBUS
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public int SCK { get; }
```

### Property Value

Type: Int32

## See Also

### Reference

Socket Class
IO.Objects.RemoteIO.mikroBUS Namespace

# SocketSCL Property

Returns the GPIO pin designator for SCL.

**Namespace:** IO.Objects.RemoteIO.mikroBUS
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public int SCL { get; }
```

### Property Value

Type: Int32

## See Also

### Reference

Socket Class
IO.Objects.RemoteIO.mikroBUS Namespace

# SocketSDA Property

Returns the GPIO pin designator for SDA.

**Namespace:** IO.Objects.RemoteIO.mikroBUS
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public int SDA { get; }
```

### Property Value

Type: Int32

## See Also

### Reference

Socket Class
IO.Objects.RemoteIO.mikroBUS Namespace

# SocketSPIDev Property

Returns the SPI device designator for this socket.

**Namespace:** IO.Objects.RemoteIO.mikroBUS
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public int SPIDev { get; }
```

### Property Value

Type: Int32

## See Also

### Reference

Socket Class
IO.Objects.RemoteIO.mikroBUS Namespace

# SocketTX Property

Returns the GPIO pin designator for TX.

**Namespace:** IO.Objects.RemoteIO.mikroBUS
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public int TX { get; }
```

### Property Value

Type: Int32

## See Also

### Reference

Socket Class
IO.Objects.RemoteIO.mikroBUS Namespace

# IO.Objects.RemoteIO.Platforms Namespace

I/O Resource Definitions for particular Remote I/O Protocol Server Platforms

## Classes

| | Class | Description |
|---|---|---|
| | BeagleBone | I/O resources (channel numbers) available on a BeagleBone Remote I/O Protocol Server |
| | MUNTS_0018 | I/O resources (channel numbers) available on a MUNTS-0018 Tutorial I/O Board Remote I/O Protocol Server. |
| | PocketBeagle | I/O resources (channel numbers) available on a BeagleBone Remote I/O Protocol Server |
| | RaspberryPi | I/O resources (channel numbers) available on a Raspberry Pi Remote I/O Protocol Server |

# BeagleBone Class

I/O resources (channel numbers) available on a BeagleBone Remote I/O Protocol Server

## Inheritance Hierarchy

SystemObject IO.Objects.RemoteIO.PlatformsBeagleBone

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static class BeagleBone
```

The BeagleBone type exposes the following members.

## Fields

| | Name | Description |
|---|---|---|
| | AIN0 | Analog input channel number for the remote BeagleBone 1.8V analog input `AIN0` at pin `P9.39`. |
| | AIN1 | Analog input channel number for the remote BeagleBone 1.8V analog input `AIN1` at pin `P9.40`. |
| | AIN2 | Analog input channel number for the |

| | | |
|---|---|---|
| | | remote BeagleBone 1.8V analog input `AIN2` at pin `P9.37`. |
| S | AIN3 | Analog input channel number for the remote BeagleBone 1.8V analog input `AIN3` at pin `P9.38`. |
| S | AIN4 | Analog input channel number for the remote BeagleBone 1.8V analog input `AIN4` at pin `P9.33`. |
| S | AIN5 | Analog input channel number for the remote BeagleBone 1.8V analog input `AIN5` at pin `P9.36`. |
| S | AIN6 | Analog input channel number for the remote BeagleBone 1.8V analog input `AIN6` at pin `P9.35`. |
| S | EHRPWM1A | PWM output channel number for the remote BeagleBone PWM output `EHRPWM1A` at pins `P8.36` or `P9.14`. |
| S | EHRPWM1B | PWM output channel number for the remote BeagleBone PWM output `EHRPWM1B` at pins `P8.34` or `P9.16`. |
| S | EHRPWM2A | PWM output channel number for the remote BeagleBone PWM output `EHRPWM2A` at pins `P8.19` or `P8.45`. |
| S | EHRPWM2B | PWM output channel number for the remote BeagleBone PWM output `EHRPWM2B` at pins `P8.13` or `P8.46`. |
| S | | |

| | | |
|---|---|---|
| | GPIO10 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO10` at pin `P8.31`. |
| S | GPIO11 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO11` at pin `P8.32`. |
| S | GPIO110 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO110` at pin `P9.31`. |
| S | GPIO111 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO111` at pin `P9.29`. |
| S | GPIO112 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO112` at pin `P9.30`. |
| S | GPIO113 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO113` at pin `P9.28`. |
| S | GPIO115 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO115` at pin `P9.27`. |
| S | GPIO117 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO117` at pin `P9.25`. |
| S | GPIO12 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO12` at pin `P9.20`. |

| | | |
|---|---|---|
| S | GPIO13 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO13` at pin `P9.19`. |
| S | GPIO14 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO14` at pin `P9.26`. |
| S | GPIO15 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO15` at pin `P9.24`. |
| S | GPIO2 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO2` at pin `P9.22`. |
| S | GPIO20 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO20` at pin `P9.41`. |
| S | GPIO22 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO22` at pin `P8.19`. |
| S | GPIO23 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO23` at pin `P8.13`. |
| S | GPIO26 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO26` at pin `P8.14`. |
| S | GPIO27 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO27` at pin `P8.17`. |

| | | |
|---|---|---|
| S | GPIO3 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO3` at pin `P9.21`. |
| S | GPIO30 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO30` at pin `P9.11`. |
| S | GPIO31 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO31` at pin `P9.13`. |
| S | GPIO32 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO32` at pin `P8.25`. |
| S | GPIO33 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO33` at pin `P8.24`. |
| S | GPIO34 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO34` at pin `P8.5`. |
| S | GPIO35 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO35` at pin `P8.6`. |
| S | GPIO36 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO36` at pin `P8.23`. |
| S | GPIO37 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO37` at pin `P8.22`. |

| | | |
|---|---|---|
| S | GPIO38 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO38` at pin `P8.3`. |
| S | GPIO39 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO39` at pin `P8.4`. |
| S | GPIO4 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO4` at pin `P9.18`. |
| S | GPIO44 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO44` at pin `P8.12`. |
| S | GPIO45 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO45` at pin `P8.11`. |
| S | GPIO46 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO46` at pin `P8.16`. |
| S | GPIO47 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO47` at pin `P8.15`. |
| S | GPIO48 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO48` at pin `P9.15`. |
| S | GPIO49 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO49` at pin |

| | | |
|---|---|---|
| | | `P9.23`. |
| S | GPIO5 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO5` at pin `P9.17`. |
| S | GPIO50 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO50` at pin `P9.14`. |
| S | GPIO51 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO51` at pin `P9.16`. |
| S | GPIO60 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO60` at pin `P9.12`. |
| S | GPIO61 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO61` at pin `P8.26`. |
| S | GPIO62 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO62` at pin `P8.21`. |
| S | GPIO63 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO63` at pin `P8.20`. |
| S | GPIO65 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO65` at pin `P8.18`. |
| S | GPIO66 | GPIO channel number for the remote |

| | | |
|---|---|---|
| | | BeagleBone GPIO pin `GPIO66` at pin `P8.7`. |
| S | GPIO67 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO67` at pin `P8.8`. |
| S | GPIO68 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO68` at pin `P8.10`. |
| S | GPIO69 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO69` at pin `P8.9`. |
| S | GPIO7 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO7` at pin `P9.42`. |
| S | GPIO70 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO70` at pin `P8.45`. |
| S | GPIO71 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO71` at pin `P8.46`. |
| S | GPIO72 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO72` at pin `P8.43`. |
| S | GPIO73 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO73` at pin `P8.44`. |
| S | | |

| | | |
|---|---|---|
| | GPIO74 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO74` at pin `P8.41`. |
| S | GPIO75 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO75` at pin `P8.42`. |
| S | GPIO76 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO76` at pin `P8.39`. |
| S | GPIO77 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO77` at pin `P8.40`. |
| S | GPIO78 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO78` at pin `P8.37`. |
| S | GPIO79 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO79` at pin `P8.38`. |
| S | GPIO8 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO8` at pin `P8.35`. |
| S | GPIO80 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO80` at pin `P8.36`. |
| S | GPIO81 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO81` at pin `P8.34`. |

| | | |
|---|---|---|
| S | GPIO86 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO86` at pin `P8.27`. |
| S | GPIO87 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO87` at pin `P8.29`. |
| S | GPIO88 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO88` at pin `P8.28`. |
| S | GPIO89 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO89` at pin `P8.30`. |
| S | GPIO9 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO9` at pin `P8.33`. |
| S | I2C2 | $I^2C$ bus channel number for the remote BeagleBone $I^2C$ bus at pins `P9.19` and `P9.20`. |
| S | SPI0_0 | SPI slave channel number for the remote BeagleBone SPI slave select `SPI0_0` at pin `P9.17` |
| S | SPI1_0 | SPI slave channel number for the remote BeagleBone SPI slave select `SPI1_0` at pin `P9.28` |
| S | SPI1_1 | SPI slave channel number for the remote BeagleBone SPI slave select `SPI1_1` at pin `P9.42` |

| | | |
|---|---|---|
| | USERLED | GPIO channel number for the remote BeagleBone user LED. |

Top

## See Also

### Reference

IO.Objects.RemoteIO.Platforms Namespace

# BeagleBone Fields

The BeagleBone type exposes the following members.

## Fields

| | Name | Description |
|---|---|---|
| S | AIN0 | Analog input channel number for the remote BeagleBone 1.8V analog input `AIN0` at pin `P9.39`. |
| S | AIN1 | Analog input channel number for the remote BeagleBone 1.8V analog input `AIN1` at pin `P9.40`. |
| S | AIN2 | Analog input channel number for the remote BeagleBone 1.8V analog input `AIN2` at pin `P9.37`. |
| S | AIN3 | Analog input channel number for the remote BeagleBone 1.8V analog input `AIN3` at pin `P9.38`. |
| S | AIN4 | Analog input channel number for the remote BeagleBone 1.8V analog input `AIN4` at pin `P9.33`. |
| S | AIN5 | Analog input channel number for the remote BeagleBone 1.8V analog input `AIN5` at pin `P9.36`. |

| | | |
|---|---|---|
| S | AIN6 | Analog input channel number for the remote BeagleBone 1.8V analog input `AIN6` at pin `P9.35`. |
| S | EHRPWM1A | PWM output channel number for the remote BeagleBone PWM output `EHRPWM1A` at pins `P8.36` or `P9.14`. |
| S | EHRPWM1B | PWM output channel number for the remote BeagleBone PWM output `EHRPWM1B` at pins `P8.34` or `P9.16`. |
| S | EHRPWM2A | PWM output channel number for the remote BeagleBone PWM output `EHRPWM2A` at pins `P8.19` or `P8.45`. |
| S | EHRPWM2B | PWM output channel number for the remote BeagleBone PWM output `EHRPWM2B` at pins `P8.13` or `P8.46`. |
| S | GPIO10 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO10` at pin `P8.31`. |
| S | GPIO11 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO11` at pin `P8.32`. |
| S | GPIO110 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO110` at pin `P9.31`. |
| S | GPIO111 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO111` at pin `P9.29`. |

| | | |
|---|---|---|
| S | GPIO112 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO112` at pin `P9.30`. |
| S | GPIO113 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO113` at pin `P9.28`. |
| S | GPIO115 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO115` at pin `P9.27`. |
| S | GPIO117 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO117` at pin `P9.25`. |
| S | GPIO12 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO12` at pin `P9.20`. |
| S | GPIO13 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO13` at pin `P9.19`. |
| S | GPIO14 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO14` at pin `P9.26`. |
| S | GPIO15 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO15` at pin `P9.24`. |
| S | GPIO2 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO2` at pin `P9.22`. |

| | | |
|---|---|---|
| S | GPIO20 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO20` at pin `P9.41`. |
| S | GPIO22 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO22` at pin `P8.19`. |
| S | GPIO23 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO23` at pin `P8.13`. |
| S | GPIO26 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO26` at pin `P8.14`. |
| S | GPIO27 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO27` at pin `P8.17`. |
| S | GPIO3 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO3` at pin `P9.21`. |
| S | GPIO30 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO30` at pin `P9.11`. |
| S | GPIO31 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO31` at pin `P9.13`. |
| S | GPIO32 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO32` at pin `P8.25`. |

| | | |
|---|---|---|
| S | GPIO33 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO33` at pin `P8.24`. |
| S | GPIO34 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO34` at pin `P8.5`. |
| S | GPIO35 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO35` at pin `P8.6`. |
| S | GPIO36 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO36` at pin `P8.23`. |
| S | GPIO37 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO37` at pin `P8.22`. |
| S | GPIO38 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO38` at pin `P8.3`. |
| S | GPIO39 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO39` at pin `P8.4`. |
| S | GPIO4 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO4` at pin `P9.18`. |
| S | GPIO44 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO44` at pin |

| | | |
|---|---|---|
| | | `P8.12`. |
| S | GPIO45 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO45` at pin `P8.11`. |
| S | GPIO46 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO46` at pin `P8.16`. |
| S | GPIO47 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO47` at pin `P8.15`. |
| S | GPIO48 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO48` at pin `P9.15`. |
| S | GPIO49 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO49` at pin `P9.23`. |
| S | GPIO5 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO5` at pin `P9.17`. |
| S | GPIO50 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO50` at pin `P9.14`. |
| S | GPIO51 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO51` at pin `P9.16`. |
| S | GPIO60 | GPIO channel number for the remote |

| | | |
|---|---|---|
| | | BeagleBone GPIO pin `GPIO60` at pin `P9.12`. |
| S | GPIO61 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO61` at pin `P8.26`. |
| S | GPIO62 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO62` at pin `P8.21`. |
| S | GPIO63 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO63` at pin `P8.20`. |
| S | GPIO65 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO65` at pin `P8.18`. |
| S | GPIO66 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO66` at pin `P8.7`. |
| S | GPIO67 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO67` at pin `P8.8`. |
| S | GPIO68 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO68` at pin `P8.10`. |
| S | GPIO69 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO69` at pin `P8.9`. |
| S | | |

| | | |
|---|---|---|
| | GPIO7 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO7` at pin `P9.42`. |
| S | GPIO70 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO70` at pin `P8.45`. |
| S | GPIO71 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO71` at pin `P8.46`. |
| S | GPIO72 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO72` at pin `P8.43`. |
| S | GPIO73 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO73` at pin `P8.44`. |
| S | GPIO74 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO74` at pin `P8.41`. |
| S | GPIO75 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO75` at pin `P8.42`. |
| S | GPIO76 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO76` at pin `P8.39`. |
| S | GPIO77 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO77` at pin `P8.40`. |

| | | |
|---|---|---|
| S | GPIO78 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO78` at pin `P8.37`. |
| S | GPIO79 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO79` at pin `P8.38`. |
| S | GPIO8 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO8` at pin `P8.35`. |
| S | GPIO80 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO80` at pin `P8.36`. |
| S | GPIO81 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO81` at pin `P8.34`. |
| S | GPIO86 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO86` at pin `P8.27`. |
| S | GPIO87 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO87` at pin `P8.29`. |
| S | GPIO88 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO88` at pin `P8.28`. |
| S | GPIO89 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO89` at pin `P8.30`. |

| | | |
|---|---|---|
| S | GPIO9 | GPIO channel number for the remote BeagleBone GPIO pin `GPIO9` at pin `P8.33`. |
| S | I2C2 | $I^2C$ bus channel number for the remote BeagleBone $I^2C$ bus at pins `P9.19` and `P9.20`. |
| S | SPI0_0 | SPI slave channel number for the remote BeagleBone SPI slave select `SPI0_0` at pin `P9.17` |
| S | SPI1_0 | SPI slave channel number for the remote BeagleBone SPI slave select `SPI1_0` at pin `P9.28` |
| S | SPI1_1 | SPI slave channel number for the remote BeagleBone SPI slave select `SPI1_1` at pin `P9.42` |
| S | USERLED | GPIO channel number for the remote BeagleBone user LED. |

Top

## See Also

### Reference

BeagleBone Class

IO.Objects.RemoteIO.Platforms Namespace

# BeagleBoneAIN0 Field

Analog input channel number for the remote BeagleBone 1.8V analog input `AIN0` at pin `P9.39`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int AIN0 = 0
```

### Field Value

Type: Int32

## See Also

### Reference

BeagleBone Class
IO.Objects.RemoteIO.Platforms Namespace

# BeagleBoneAIN1 Field

Analog input channel number for the remote BeagleBone 1.8V analog input `AIN1` at pin `P9.40`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int AIN1 = 1
```

### Field Value

Type: Int32

## See Also

### Reference

BeagleBone Class
IO.Objects.RemoteIO.Platforms Namespace

# BeagleBoneAIN2 Field

Analog input channel number for the remote BeagleBone 1.8V analog input `AIN2` at pin `P9.37`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int AIN2 = 2
```

### Field Value

Type: Int32

## See Also

### Reference

BeagleBone Class
IO.Objects.RemoteIO.Platforms Namespace

# BeagleBoneAIN3 Field

Analog input channel number for the remote BeagleBone 1.8V analog input `AIN3` at pin `P9.38`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int AIN3 = 3
```

### Field Value

Type: Int32

## See Also

### Reference

BeagleBone Class
IO.Objects.RemoteIO.Platforms Namespace

# BeagleBoneAIN4 Field

Analog input channel number for the remote BeagleBone 1.8V analog input `AIN4` at pin `P9.33`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int AIN4 = 4
```

### Field Value

Type: Int32

## See Also

### Reference

BeagleBone Class
IO.Objects.RemoteIO.Platforms Namespace

# BeagleBoneAIN5 Field

Analog input channel number for the remote BeagleBone 1.8V analog input `AIN5` at pin `P9.36`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int AIN5 = 5
```

### Field Value

Type: Int32

## See Also

### Reference

BeagleBone Class
IO.Objects.RemoteIO.Platforms Namespace

# BeagleBoneAIN6 Field

Analog input channel number for the remote BeagleBone 1.8V analog input `AIN6` at pin `P9.35`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int AIN6 = 6
```

### Field Value

Type: Int32

## See Also

### Reference

BeagleBone Class
IO.Objects.RemoteIO.Platforms Namespace

# BeagleBoneEHRPWM1A Field

PWM output channel number for the remote BeagleBone PWM output `EHRPWM1A` at pins `P8.36` or `P9.14`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int EHRPWM1A = 0
```

### Field Value

Type: Int32

## See Also

### Reference

BeagleBone Class
IO.Objects.RemoteIO.Platforms Namespace

# BeagleBoneEHRPWM1B Field

PWM output channel number for the remote BeagleBone PWM output `EHRPWM1B` at pins `P8.34` or `P9.16`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public const int EHRPWM1B = 1
```

### Field Value

Type: Int32

## See Also

### Reference

BeagleBone Class
IO.Objects.RemoteIO.Platforms Namespace

# BeagleBoneEHRPWM2A Field

PWM output channel number for the remote BeagleBone PWM output `EHRPWM2A` at pins `P8.19` or `P8.45`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public const int EHRPWM2A = 2
```

### Field Value

Type: Int32

## See Also

### Reference

BeagleBone Class
IO.Objects.RemoteIO.Platforms Namespace

# BeagleBoneEHRPWM2B Field

PWM output channel number for the remote BeagleBone PWM output `EHRPWM2B` at pins `P8.13` or `P8.46`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public const int EHRPWM2B = 3
```

### Field Value

Type: Int32

## See Also

### Reference

BeagleBone Class
IO.Objects.RemoteIO.Platforms Namespace

# BeagleBoneGPIO10 Field

GPIO channel number for the remote BeagleBone GPIO pin `GPIO10` at pin `P8.31`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public const int GPIO10 = 10
```

### Field Value

Type: Int32

## See Also

### Reference

BeagleBone Class
IO.Objects.RemoteIO.Platforms Namespace

# BeagleBoneGPIO11 Field

GPIO channel number for the remote BeagleBone GPIO pin `GPIO11` at pin `P8.32`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO11 = 11
```

### Field Value

Type: Int32

## See Also

### Reference

BeagleBone Class
IO.Objects.RemoteIO.Platforms Namespace

# BeagleBoneGPIO110 Field

GPIO channel number for the remote BeagleBone GPIO pin `GPIO110` at pin `P9.31`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO110 = 110
```

### Field Value

Type: Int32

## Remarks

Conflicts with `EHRPWM0A` or `SPI1 SCLK`.

## See Also

### Reference

BeagleBone Class
IO.Objects.RemoteIO.Platforms Namespace

# BeagleBoneGPIO111 Field

GPIO channel number for the remote BeagleBone GPIO pin `GPIO111` at pin `P9.29`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO111 = 111
```

### Field Value

Type: Int32

## Remarks

Conflicts with `SPI1 MISO`.

## See Also

### Reference

BeagleBone Class
IO.Objects.RemoteIO.Platforms Namespace

# BeagleBoneGPIO112 Field

GPIO channel number for the remote BeagleBone GPIO pin `GPIO112` at pin `P9.30`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO112 = 112
```

### Field Value

Type: Int32

## Remarks

Conflicts with `SPI1 MOSI`.

## See Also

### Reference

BeagleBone Class
IO.Objects.RemoteIO.Platforms Namespace

# BeagleBoneGPIO113 Field

GPIO channel number for the remote BeagleBone GPIO pin `GPIO113` at pin `P9.28`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO113 = 113
```

### Field Value

Type: Int32

## Remarks

Conflicts with `SPI1 SS0`.

## See Also

### Reference

BeagleBone Class
IO.Objects.RemoteIO.Platforms Namespace

# BeagleBoneGPIO115 Field

GPIO channel number for the remote BeagleBone GPIO pin `GPIO115` at pin `P9.27`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO115 = 115
```

### Field Value

Type: Int32

## See Also

### Reference

BeagleBone Class
IO.Objects.RemoteIO.Platforms Namespace

# BeagleBoneGPIO117 Field

GPIO channel number for the remote BeagleBone GPIO pin `GPIO117` at pin `P9.25`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO117 = 117
```

### Field Value

Type: Int32

## See Also

### Reference

BeagleBone Class
IO.Objects.RemoteIO.Platforms Namespace

# BeagleBoneGPIO12 Field

GPIO channel number for the remote BeagleBone GPIO pin `GPIO12` at pin `P9.20`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO12 = 12
```

### Field Value

Type: Int32

## Remarks

Conflicts with `I2C2 SDA`.

## See Also

### Reference

BeagleBone Class
IO.Objects.RemoteIO.Platforms Namespace

# BeagleBoneGPIO13 Field

GPIO channel number for the remote BeagleBone GPIO pin `GPIO13` at pin `P9.19`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO13 = 13
```

### Field Value

Type: Int32

## Remarks

Conflicts with `I2C2 SCL`.

## See Also

### Reference

BeagleBone Class
IO.Objects.RemoteIO.Platforms Namespace

# BeagleBoneGPIO14 Field

GPIO channel number for the remote BeagleBone GPIO pin `GPIO14` at pin `P9.26`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public const int GPIO14 = 14
```

### Field Value

Type: Int32

## Remarks

Conflicts with `UART1 RXD`.

## See Also

### Reference

BeagleBone Class
IO.Objects.RemoteIO.Platforms Namespace

# BeagleBoneGPIO15 Field

GPIO channel number for the remote BeagleBone GPIO pin `GPIO15` at pin `P9.24`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public const int GPIO15 = 15
```

### Field Value

Type: Int32

## Remarks

Conflicts with `UART1 TXD`.

## See Also

### Reference

BeagleBone Class
IO.Objects.RemoteIO.Platforms Namespace

# BeagleBoneGPIO2 Field

GPIO channel number for the remote BeagleBone GPIO pin `GPIO2` at pin `P9.22`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO2 = 2
```

### Field Value

Type: Int32

## Remarks

Conflicts with `EHRPWM0B`, `SPI0 SCK` or `UART2 RXD`.

## See Also

### Reference

BeagleBone Class
IO.Objects.RemoteIO.Platforms Namespace

# BeagleBoneGPIO20 Field

GPIO channel number for the remote BeagleBone GPIO pin `GPIO20` at pin `P9.41`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO20 = 20
```

### Field Value

Type: Int32

## See Also

### Reference

BeagleBone Class
IO.Objects.RemoteIO.Platforms Namespace

# BeagleBoneGPIO22 Field

GPIO channel number for the remote BeagleBone GPIO pin `GPIO22` at pin `P8.19`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public const int GPIO22 = 22
```

### Field Value

Type: Int32

## Remarks

Conflicts with `EHRPWM2A`.

## See Also

### Reference

BeagleBone Class
IO.Objects.RemoteIO.Platforms Namespace

# BeagleBoneGPIO23 Field

GPIO channel number for the remote BeagleBone GPIO pin `GPIO23` at pin `P8.13`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public const int GPIO23 = 23
```

### Field Value

Type: Int32

## Remarks

Conflicts with `EHRPWM2B`.

## See Also

### Reference

BeagleBone Class
IO.Objects.RemoteIO.Platforms Namespace

# BeagleBoneGPIO26 Field

GPIO channel number for the remote BeagleBone GPIO pin `GPIO26` at pin `P8.14`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO26 = 26
```

### Field Value

Type: Int32

## See Also

### Reference

BeagleBone Class
IO.Objects.RemoteIO.Platforms Namespace

# BeagleBoneGPIO27 Field

GPIO channel number for the remote BeagleBone GPIO pin `GPIO27` at pin `P8.17`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO27 = 27
```

### Field Value

Type: Int32

## See Also

### Reference

BeagleBone Class
IO.Objects.RemoteIO.Platforms Namespace

# BeagleBoneGPIO3 Field

GPIO channel number for the remote BeagleBone GPIO pin `GPIO3` at pin `P9.21`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO3 = 3
```

### Field Value

Type: Int32

## Remarks

Conflicts with `SPI0 MISO` or `UART2 TXD`.

## See Also

### Reference

BeagleBone Class
IO.Objects.RemoteIO.Platforms Namespace

# BeagleBoneGPIO30 Field

GPIO channel number for the remote BeagleBone GPIO pin `GPIO30` at pin `P9.11`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public const int GPIO30 = 30
```

### Field Value

Type: Int32

## Remarks

Conflicts with `UART4 RXD`.

## See Also

### Reference

BeagleBone Class
IO.Objects.RemoteIO.Platforms Namespace

# BeagleBoneGPIO31 Field

GPIO channel number for the remote BeagleBone GPIO pin `GPIO31` at pin `P9.13`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public const int GPIO31 = 31
```

### Field Value

Type: Int32

## Remarks

Conflicts with `UART4 TXD`.

## See Also

### Reference

BeagleBone Class
IO.Objects.RemoteIO.Platforms Namespace

# BeagleBoneGPIO32 Field

GPIO channel number for the remote BeagleBone GPIO pin `GPIO32` at pin `P8.25`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public const int GPIO32 = 32
```

### Field Value

Type: Int32

## Remarks

Conflicts with `MMC1 DAT0`.

## See Also

### Reference

BeagleBone Class
IO.Objects.RemoteIO.Platforms Namespace

# BeagleBoneGPIO33 Field

GPIO channel number for the remote BeagleBone GPIO pin `GPIO33` at pin `P8.24`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public const int GPIO33 = 33
```

### Field Value

Type: Int32

## Remarks

Conflicts with `MMC1 DAT1`.

## See Also

### Reference

BeagleBone Class
IO.Objects.RemoteIO.Platforms Namespace

# BeagleBoneGPIO34 Field

GPIO channel number for the remote BeagleBone GPIO pin `GPIO34` at pin `P8.5`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO34 = 34
```

### Field Value

Type: Int32

## Remarks

Conflicts with `MMC1 DAT2`.

## See Also

### Reference

BeagleBone Class
IO.Objects.RemoteIO.Platforms Namespace

# BeagleBoneGPIO35 Field

GPIO channel number for the remote BeagleBone GPIO pin `GPIO35` at pin `P8.6`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public const int GPIO35 = 35
```

### Field Value

Type: Int32

## Remarks

Conflicts with `MMC1 DAT3`.

## See Also

### Reference

BeagleBone Class
IO.Objects.RemoteIO.Platforms Namespace

# BeagleBoneGPIO36 Field

GPIO channel number for the remote BeagleBone GPIO pin `GPIO36` at pin `P8.23`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public const int GPIO36 = 36
```

### Field Value

Type: Int32

## Remarks

Conflicts with `MMC1 DAT4`.

## See Also

### Reference

BeagleBone Class
IO.Objects.RemoteIO.Platforms Namespace

# BeagleBoneGPIO37 Field

GPIO channel number for the remote BeagleBone GPIO pin `GPIO37` at pin `P8.22`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public const int GPIO37 = 37
```

### Field Value

Type: Int32

## Remarks

Conflicts with `MMC1 DAT5`.

## See Also

### Reference

BeagleBone Class
IO.Objects.RemoteIO.Platforms Namespace

# BeagleBoneGPIO38 Field

GPIO channel number for the remote BeagleBone GPIO pin `GPIO38` at pin `P8.3`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public const int GPIO38 = 38
```

### Field Value

Type: Int32

## Remarks

Conflicts with `MMC1 DAT6`.

## See Also

### Reference

BeagleBone Class
IO.Objects.RemoteIO.Platforms Namespace

# BeagleBoneGPIO39 Field

GPIO channel number for the remote BeagleBone GPIO pin `GPIO39` at pin `P8.4`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public const int GPIO39 = 39
```

### Field Value

Type: Int32

## Remarks

Conflicts with `MMC1 DAT7`.

## See Also

### Reference

BeagleBone Class
IO.Objects.RemoteIO.Platforms Namespace

# BeagleBoneGPIO4 Field

GPIO channel number for the remote BeagleBone GPIO pin `GPIO4` at pin `P9.18`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO4 = 4
```

### Field Value

Type: Int32

## Remarks

Conflicts with `SPI0 MOSI`.

## See Also

### Reference

BeagleBone Class
IO.Objects.RemoteIO.Platforms Namespace

# BeagleBoneGPIO44 Field

GPIO channel number for the remote BeagleBone GPIO pin `GPIO44` at pin `P8.12`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO44 = 44
```

### Field Value

Type: Int32

## See Also

### Reference

BeagleBone Class
IO.Objects.RemoteIO.Platforms Namespace

# BeagleBoneGPIO45 Field

GPIO channel number for the remote BeagleBone GPIO pin `GPIO45` at pin `P8.11`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO45 = 45
```

### Field Value

Type: Int32

## See Also

### Reference

BeagleBone Class
IO.Objects.RemoteIO.Platforms Namespace

# BeagleBoneGPIO46 Field

GPIO channel number for the remote BeagleBone GPIO pin `GPIO46` at pin `P8.16`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO46 = 46
```

### Field Value

Type: Int32

## See Also

### Reference

BeagleBone Class
IO.Objects.RemoteIO.Platforms Namespace

# BeagleBoneGPIO47 Field

GPIO channel number for the remote BeagleBone GPIO pin `GPIO47` at pin `P8.15`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO47 = 47
```

### Field Value

Type: Int32

## See Also

### Reference

BeagleBone Class
IO.Objects.RemoteIO.Platforms Namespace

# BeagleBoneGPIO48 Field

GPIO channel number for the remote BeagleBone GPIO pin `GPIO48` at pin `P9.15`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO48 = 48
```

### Field Value

Type: Int32

## See Also

### Reference

BeagleBone Class
IO.Objects.RemoteIO.Platforms Namespace

# BeagleBoneGPIO49 Field

GPIO channel number for the remote BeagleBone GPIO pin `GPIO49` at pin `P9.23`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO49 = 49
```

### Field Value

Type: Int32

## See Also

### Reference

BeagleBone Class
IO.Objects.RemoteIO.Platforms Namespace

# BeagleBoneGPIO5 Field

GPIO channel number for the remote BeagleBone GPIO pin `GPIO5` at pin `P9.17`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO5 = 5
```

### Field Value

Type: Int32

## Remarks

Conflicts with `SPI0 SS0`.

## See Also

### Reference

BeagleBone Class
IO.Objects.RemoteIO.Platforms Namespace

# BeagleBoneGPIO50 Field

GPIO channel number for the remote BeagleBone GPIO pin `GPIO50` at pin `P9.14`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO50 = 50
```

### Field Value

Type: Int32

## Remarks

Conflicts with `EHRPWM1A`.

## See Also

### Reference

BeagleBone Class
IO.Objects.RemoteIO.Platforms Namespace

# BeagleBoneGPIO51 Field

GPIO channel number for the remote BeagleBone GPIO pin `GPIO51` at pin `P9.16`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public const int GPIO51 = 51
```

### Field Value

Type: Int32

## Remarks

Conflicts with `EHRPWM1B`.

## See Also

### Reference

BeagleBone Class
IO.Objects.RemoteIO.Platforms Namespace

# BeagleBoneGPIO60 Field

GPIO channel number for the remote BeagleBone GPIO pin `GPIO60` at pin `P9.12`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO60 = 60
```

### Field Value

Type: Int32

## See Also

### Reference

BeagleBone Class
IO.Objects.RemoteIO.Platforms Namespace

# BeagleBoneGPIO61 Field

GPIO channel number for the remote BeagleBone GPIO pin `GPIO61` at pin `P8.26`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO61 = 61
```

### Field Value

Type: Int32

## See Also

### Reference

BeagleBone Class
IO.Objects.RemoteIO.Platforms Namespace

# BeagleBoneGPIO62 Field

GPIO channel number for the remote BeagleBone GPIO pin `GPIO62` at pin `P8.21`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO62 = 62
```

### Field Value

Type: Int32

## Remarks

Conflicts with `MMC1 CLK`.

## See Also

### Reference

BeagleBone Class
IO.Objects.RemoteIO.Platforms Namespace

# BeagleBoneGPIO63 Field

GPIO channel number for the remote BeagleBone GPIO pin `GPIO63` at pin `P8.20`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public const int GPIO63 = 63
```

### Field Value

Type: Int32

## Remarks

Conflicts with `MMC1 CMD`.

## See Also

### Reference

BeagleBone Class
IO.Objects.RemoteIO.Platforms Namespace

# BeagleBoneGPIO65 Field

GPIO channel number for the remote BeagleBone GPIO pin `GPIO65` at pin `P8.18`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO65 = 65
```

### Field Value

Type: Int32

## See Also

### Reference

BeagleBone Class
IO.Objects.RemoteIO.Platforms Namespace

# BeagleBoneGPIO66 Field

GPIO channel number for the remote BeagleBone GPIO pin `GPIO66` at pin `P8.7`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO66 = 66
```

### Field Value

Type: Int32

## See Also

### Reference

BeagleBone Class
IO.Objects.RemoteIO.Platforms Namespace

# BeagleBoneGPIO67 Field

GPIO channel number for the remote BeagleBone GPIO pin `GPIO67` at pin `P8.8`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO67 = 67
```

### Field Value

Type: Int32

## See Also

### Reference

BeagleBone Class
IO.Objects.RemoteIO.Platforms Namespace

# BeagleBoneGPIO68 Field

GPIO channel number for the remote BeagleBone GPIO pin `GPIO68` at pin `P8.10`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO68 = 68
```

### Field Value

Type: Int32

## See Also

### Reference

BeagleBone Class
IO.Objects.RemoteIO.Platforms Namespace

# BeagleBoneGPIO69 Field

GPIO channel number for the remote BeagleBone GPIO pin `GPIO69` at pin `P8.9`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO69 = 69
```

### Field Value

Type: Int32

## See Also

### Reference

BeagleBone Class
IO.Objects.RemoteIO.Platforms Namespace

# BeagleBoneGPIO7 Field

GPIO channel number for the remote BeagleBone GPIO pin `GPIO7` at pin `P9.42`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO7 = 7
```

### Field Value

Type: Int32

## Remarks

Conflicts with `SPI1 SS1`.

## See Also

### Reference

BeagleBone Class
IO.Objects.RemoteIO.Platforms Namespace

# BeagleBoneGPIO70 Field

GPIO channel number for the remote BeagleBone GPIO pin `GPIO70` at pin `P8.45`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public const int GPIO70 = 70
```

### Field Value

Type: Int32

## Remarks

Conflicts with `EHRPWM2A`.

## See Also

### Reference

BeagleBone Class
IO.Objects.RemoteIO.Platforms Namespace

# BeagleBoneGPIO71 Field

GPIO channel number for the remote BeagleBone GPIO pin `GPIO71` at pin `P8.46`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO71 = 71
```

### Field Value

Type: Int32

## Remarks

Conflicts with `EHRPWM2B`.

## See Also

### Reference

BeagleBone Class
IO.Objects.RemoteIO.Platforms Namespace

# BeagleBoneGPIO72 Field

GPIO channel number for the remote BeagleBone GPIO pin `GPIO72` at pin `P8.43`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO72 = 72
```

### Field Value

Type: Int32

## See Also

### Reference

BeagleBone Class
IO.Objects.RemoteIO.Platforms Namespace

# BeagleBoneGPIO73 Field

GPIO channel number for the remote BeagleBone GPIO pin `GPIO73` at pin `P8.44`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO73 = 73
```

### Field Value

Type: Int32

## See Also

### Reference

BeagleBone Class
IO.Objects.RemoteIO.Platforms Namespace

# BeagleBoneGPIO74 Field

GPIO channel number for the remote BeagleBone GPIO pin `GPIO74` at pin `P8.41`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO74 = 74
```

### Field Value

Type: Int32

## See Also

### Reference

BeagleBone Class
IO.Objects.RemoteIO.Platforms Namespace

# BeagleBoneGPIO75 Field

GPIO channel number for the remote BeagleBone GPIO pin `GPIO75` at pin `P8.42`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO75 = 75
```

### Field Value

Type: Int32

## See Also

### Reference

BeagleBone Class
IO.Objects.RemoteIO.Platforms Namespace

# BeagleBoneGPIO76 Field

GPIO channel number for the remote BeagleBone GPIO pin `GPIO76` at pin `P8.39`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO76 = 76
```

### Field Value

Type: Int32

## See Also

### Reference

BeagleBone Class
IO.Objects.RemoteIO.Platforms Namespace

# BeagleBoneGPIO77 Field

GPIO channel number for the remote BeagleBone GPIO pin `GPIO77` at pin `P8.40`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO77 = 77
```

### Field Value

Type: Int32

## See Also

### Reference

BeagleBone Class
IO.Objects.RemoteIO.Platforms Namespace

# BeagleBoneGPIO78 Field

GPIO channel number for the remote BeagleBone GPIO pin `GPIO78` at pin `P8.37`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public const int GPIO78 = 78
```

### Field Value

Type: Int32

## Remarks

Conflicts with `UART5 TXD`.

## See Also

### Reference

BeagleBone Class
IO.Objects.RemoteIO.Platforms Namespace

# BeagleBoneGPIO79 Field

GPIO channel number for the remote BeagleBone GPIO pin `GPIO79` at pin `P8.38`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO79 = 79
```

### Field Value

Type: Int32

## Remarks

Conflicts with `UART5_RXD`.

## See Also

### Reference

BeagleBone Class
IO.Objects.RemoteIO.Platforms Namespace

# BeagleBoneGPIO8 Field

GPIO channel number for the remote BeagleBone GPIO pin `GPIO8` at pin `P8.35`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO8 = 8
```

### Field Value

Type: Int32

## See Also

### Reference

BeagleBone Class
IO.Objects.RemoteIO.Platforms Namespace

# BeagleBoneGPIO80 Field

GPIO channel number for the remote BeagleBone GPIO pin `GPIO80` at pin `P8.36`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO80 = 80
```

### Field Value

Type: Int32

## Remarks

Conflicts with `EHRPWM1A`.

## See Also

### Reference

BeagleBone Class
IO.Objects.RemoteIO.Platforms Namespace

# BeagleBoneGPIO81 Field

GPIO channel number for the remote BeagleBone GPIO pin `GPIO81` at pin `P8.34`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO81 = 81
```

### Field Value

Type: Int32

## Remarks

Conflicts with `EHRPWM1B`.

## See Also

### Reference

BeagleBone Class
IO.Objects.RemoteIO.Platforms Namespace

# BeagleBoneGPIO86 Field

GPIO channel number for the remote BeagleBone GPIO pin `GPIO86` at pin `P8.27`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO86 = 86
```

### Field Value

Type: Int32

## See Also

### Reference

BeagleBone Class
IO.Objects.RemoteIO.Platforms Namespace

# BeagleBoneGPIO87 Field

GPIO channel number for the remote BeagleBone GPIO pin `GPIO87` at pin `P8.29`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO87 = 87
```

### Field Value

Type: Int32

## See Also

### Reference

BeagleBone Class
IO.Objects.RemoteIO.Platforms Namespace

# BeagleBoneGPIO88 Field

GPIO channel number for the remote BeagleBone GPIO pin `GPIO88` at pin `P8.28`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO88 = 88
```

### Field Value

Type: Int32

## See Also

### Reference

BeagleBone Class
IO.Objects.RemoteIO.Platforms Namespace

# BeagleBoneGPIO89 Field

GPIO channel number for the remote BeagleBone GPIO pin `GPIO89` at pin `P8.30`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO89 = 89
```

### Field Value

Type: Int32

## See Also

### Reference

BeagleBone Class
IO.Objects.RemoteIO.Platforms Namespace

# BeagleBoneGPIO9 Field

GPIO channel number for the remote BeagleBone GPIO pin `GPIO9` at pin `P8.33`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO9 = 9
```

### Field Value

Type: Int32

## See Also

### Reference

BeagleBone Class
IO.Objects.RemoteIO.Platforms Namespace

# BeagleBoneI2C2 Field

$I^2C$ bus channel number for the remote BeagleBone $I^2C$ bus at pins `P9.19` and `P9.20`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int I2C2 = 0
```

### Field Value

Type: Int32

## Remarks

Conflicts with `GPIO12` and `GPIO13`.

## See Also

### Reference

BeagleBone Class
IO.Objects.RemoteIO.Platforms Namespace

# BeagleBoneSPI0_0 Field

SPI slave channel number for the remote BeagleBone SPI slave select `SPI0_0` at pin `P9.17`

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public const int SPI0_0 = 2
```

### Field Value

Type: Int32

## See Also

### Reference

BeagleBone Class
IO.Objects.RemoteIO.Platforms Namespace

# BeagleBoneSPI1_0 Field

SPI slave channel number for the remote BeagleBone SPI slave select `SPI1_0` at pin `P9.28`

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int SPI1_0 = 0
```

### Field Value

Type: Int32

## See Also

### Reference

BeagleBone Class
IO.Objects.RemoteIO.Platforms Namespace

# BeagleBoneSPI1_1 Field

SPI slave channel number for the remote BeagleBone SPI slave select `SPI1_1` at pin `P9.42`

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int SPI1_1 = 1
```

### Field Value

Type: Int32

## See Also

### Reference

BeagleBone Class
IO.Objects.RemoteIO.Platforms Namespace

# BeagleBoneUSERLED Field

GPIO channel number for the remote BeagleBone user LED.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int USERLED = 0
```

### Field Value

Type: Int32

## Remarks

This GPIO channel cannot be configured as an input.

## See Also

### Reference

BeagleBone Class
IO.Objects.RemoteIO.Platforms Namespace

# MUNTS_0018 Class

I/O resources (channel numbers) available on a MUNTS-0018 Tutorial I/O Board Remote I/O Protocol Server.

## Inheritance Hierarchy

SystemObject IO.Objects.RemoteIO.PlatformsMUNTS_0018

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static class MUNTS_0018
```

The MUNTS_0018 type exposes the following members.

## Fields

| | Name | Description |
|---|---|---|
| S | D1 | GPIO channel number for the on-board LED `D1`. Cannot be configured as an input. |
| S | J10A0 | Analog input channel number for Grove Analog Input Connector `J10` pin `A0` (MCP3204 input `CH2`). |
| S | J10A1 | Analog input channel number for Grove |

| | | |
|---|---|---|
| | | Analog Input Connector `J10` pin `A1` (MCP3204 input `CH3`). |
| S | J11A0 | Analog input channel number for Grove Analog Input Connector `J11` pin `A0` (MCP3204 input `CH0`). |
| S | J11A1 | Analog input channel number for Grove Analog Input Connector `J11` pin `A1` (MCP3204 input `CH1`). |
| S | J2PWM | PWM channel number for Servo Header `J2`. |
| S | J3PWM | PWM channel number for Servo Header `J3`. |
| S | J4D0 | GPIO channel number for Grove Digital I/O Connector `J4` pin `D0`. |
| S | J4D1 | GPIO channel number for Grove Digital I/O Connector `J4` pin `D1`. |
| S | J5D0 | GPIO channel number for Grove Digital I/O Connector `J5` pin `D0`. |
| S | J5D1 | GPIO channel number for Grove Digital I/O Connector `J5` pin `D1`. |
| S | J6D0 | GPIO channel number for Grove DC Motor Driver Connector `J6` pin `D0`. |
| S | J6D1 | GPIO channel number for Grove DC Motor Driver Connector `J6` pin `D1`. |
| S | J6DIR | GPIO channel number for DC Motor |

| | | |
|---|---|---|
| | | Driver Grove Connector `J6` pin `D1`. |
| | J6PWM | PWM channel number for DC Motor Driver Grove Connector `J6` pin `D0`. |
| | J7D0 | GPIO channel number for Grove DC Motor Driver Connector `J7` pin `D0`. |
| | J7D1 | GPIO channel number for Grove DC Motor Driver Connector `J7` pin `D1`. |
| | J7DIR | GPIO channel number for Grove DC Motor Driver Connector `J7` pin `D1`. |
| | J7PWM | PWM channel number for Grove DC Motor Driver Connector `J7` pin `D0`. |
| | J9I2C | $I^2C$ bus channel number for Grove $I^2C$ Connector `J9`. |
| | SW1 | GPIO channel number for the on-board momentary switch `SW1`. Cannot be configured as an output. |
| | USERLED | GPIO channel number for the remote Raspberry Pi user LED. Cannot be configured as an input. |

Top

# Remarks

The following device tree overlay commands must be added to `config.txt` on the Raspberry Pi hosting the MUNTS-0018 Tutorial I/O Board Remote I/O Protocol Server:

Copy

```
dtoverlay=anyspi,spi0-1,dev="microchip,mcp3204",sp
dtoverlay=pwm-2chan,pin=12,func=4,pin2=19,func2=2
```

## See Also

### Reference

IO.Objects.RemoteIO.Platforms Namespace

# MUNTS_0018 Fields

The MUNTS_0018 type exposes the following members.

## Fields

| | Name | Description |
|---|---|---|
| S | D1 | GPIO channel number for the on-board LED `D1`. Cannot be configured as an input. |
| S | J10A0 | Analog input channel number for Grove Analog Input Connector `J10` pin `A0` (MCP3204 input `CH2`). |
| S | J10A1 | Analog input channel number for Grove Analog Input Connector `J10` pin `A1` (MCP3204 input `CH3`). |
| S | J11A0 | Analog input channel number for Grove Analog Input Connector `J11` pin `A0` (MCP3204 input `CH0`). |
| S | J11A1 | Analog input channel number for Grove Analog Input Connector `J11` pin `A1` (MCP3204 input `CH1`). |
| S | J2PWM | PWM channel number for Servo Header `J2`. |
| S | J3PWM | PWM channel number for Servo Header |

| | | |
|---|---|---|
| | | `J3`. |
| S | J4D0 | GPIO channel number for Grove Digital I/O Connector `J4` pin `D0`. |
| S | J4D1 | GPIO channel number for Grove Digital I/O Connector `J4` pin `D1`. |
| S | J5D0 | GPIO channel number for Grove Digital I/O Connector `J5` pin `D0`. |
| S | J5D1 | GPIO channel number for Grove Digital I/O Connector `J5` pin `D1`. |
| S | J6D0 | GPIO channel number for Grove DC Motor Driver Connector `J6` pin `D0`. |
| S | J6D1 | GPIO channel number for Grove DC Motor Driver Connector `J6` pin `D1`. |
| S | J6DIR | GPIO channel number for DC Motor Driver Grove Connector `J6` pin `D1`. |
| S | J6PWM | PWM channel number for DC Motor Driver Grove Connector `J6` pin `D0`. |
| S | J7D0 | GPIO channel number for Grove DC Motor Driver Connector `J7` pin `D0`. |
| S | J7D1 | GPIO channel number for Grove DC Motor Driver Connector `J7` pin `D1`. |
| S | J7DIR | GPIO channel number for Grove DC Motor Driver Connector `J7` pin `D1`. |
| S | J7PWM | PWM channel number for Grove DC |

| | | |
|---|---|---|
| | | Motor Driver Connector `J7` pin `D0`. |
| S | J9I2C | $I^2C$ bus channel number for Grove $I^2C$ Connector `J9`. |
| S | SW1 | GPIO channel number for the on-board momentary switch `SW1`. Cannot be configured as an output. |
| S | USERLED | GPIO channel number for the remote Raspberry Pi user LED. Cannot be configured as an input. |

Top

## See Also

**Reference**

MUNTS_0018 Class

IO.Objects.RemoteIO.Platforms Namespace

# MUNTS_0018D1 Field

GPIO channel number for the on-board LED `D1`. Cannot be configured as an input.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int D1 = 26
```

### Field Value

Type: Int32

## See Also

### Reference

MUNTS_0018 Class
IO.Objects.RemoteIO.Platforms Namespace

# MUNTS_0018J10A0 Field

Analog input channel number for Grove Analog Input Connector `J10` pin `A0` (MCP3204 input `CH2`).

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int J10A0 = 2
```

### Field Value

Type: Int32

## See Also

### Reference

MUNTS_0018 Class
IO.Objects.RemoteIO.Platforms Namespace

# MUNTS_0018J10A1 Field

Analog input channel number for Grove Analog Input Connector `J10` pin `A1` (MCP3204 input `CH3`).

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public const int J10A1 = 3
```

### Field Value

Type: Int32

## See Also

### Reference

MUNTS_0018 Class
IO.Objects.RemoteIO.Platforms Namespace

# MUNTS_0018J11A0 Field

Analog input channel number for Grove Analog Input Connector `J11` pin `A0` (MCP3204 input `CH0`).

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int J11A0 = 0
```

### Field Value

Type: Int32

## See Also

### Reference

MUNTS_0018 Class
IO.Objects.RemoteIO.Platforms Namespace

# MUNTS_0018J11A1 Field

Analog input channel number for Grove Analog Input Connector `J11` pin `A1` (MCP3204 input `CH1`).

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int J11A1 = 1
```

### Field Value

Type: Int32

## See Also

### Reference

MUNTS_0018 Class
IO.Objects.RemoteIO.Platforms Namespace

# MUNTS_0018J2PWM Field

PWM channel number for Servo Header `J2`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int J2PWM = 0
```

### Field Value

Type: Int32

## See Also

### Reference

MUNTS_0018 Class
IO.Objects.RemoteIO.Platforms Namespace

# MUNTS_0018J3PWM Field

PWM channel number for Servo Header `J3`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int J3PWM = 1
```

### Field Value

Type: Int32

## See Also

### Reference

MUNTS_0018 Class
IO.Objects.RemoteIO.Platforms Namespace

# MUNTS_0018J4D0 Field

GPIO channel number for Grove Digital I/O Connector `J4` pin `D0`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int J4D0 = 23
```

### Field Value

Type: Int32

## See Also

### Reference

MUNTS_0018 Class
IO.Objects.RemoteIO.Platforms Namespace

# MUNTS_0018J4D1 Field

GPIO channel number for Grove Digital I/O Connector `J4` pin `D1`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int J4D1 = 24
```

### Field Value

Type: Int32

## See Also

### Reference

MUNTS_0018 Class
IO.Objects.RemoteIO.Platforms Namespace

# MUNTS_0018J5D0 Field

GPIO channel number for Grove Digital I/O Connector `J5` pin `D0`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int J5D0 = 5
```

### Field Value

Type: Int32

## See Also

### Reference

MUNTS_0018 Class
IO.Objects.RemoteIO.Platforms Namespace

# MUNTS_0018J5D1 Field

GPIO channel number for Grove Digital I/O Connector `J5` pin `D1`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int J5D1 = 4
```

### Field Value

Type: Int32

## See Also

### Reference

MUNTS_0018 Class
IO.Objects.RemoteIO.Platforms Namespace

# MUNTS_0018J6D0 Field

GPIO channel number for Grove DC Motor Driver Connector `J6` pin `D0`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int J6D0 = 12
```

### Field Value

Type: Int32

## Remarks

This GPIO channel is normally unusable because `GPIO12` is mapped to `PWM0`.

## See Also

### Reference

MUNTS_0018 Class
IO.Objects.RemoteIO.Platforms Namespace

# MUNTS_0018J6D1 Field

GPIO channel number for Grove DC Motor Driver Connector `J6` pin `D1`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int J6D1 = 13
```

### Field Value

Type: Int32

## See Also

### Reference

MUNTS_0018 Class
IO.Objects.RemoteIO.Platforms Namespace

# MUNTS_0018J6DIR Field

GPIO channel number for DC Motor Driver Grove Connector `J6` pin `D1`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int J6DIR = 13
```

### Field Value

Type: Int32

## Remarks

`J6DIR` controls the motor **direction**.

## See Also

### Reference

MUNTS_0018 Class
IO.Objects.RemoteIO.Platforms Namespace

# MUNTS_0018J6PWM Field

PWM channel number for DC Motor Driver Grove Connector `J6` pin `D0`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public const int J6PWM = 0
```

### Field Value

Type: Int32

## Remarks

`J6PWM` controls the motor **speed**.

## See Also

### Reference

MUNTS_0018 Class
IO.Objects.RemoteIO.Platforms Namespace

# MUNTS_0018J7D0 Field

GPIO channel number for Grove DC Motor Driver Connector `J7` pin `D0`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int J7D0 = 19
```

### Field Value

Type: Int32

## Remarks

This GPIO channel is normally unusable because `GPIO19` is mapped to `PWM1`.

## See Also

### Reference

MUNTS_0018 Class
IO.Objects.RemoteIO.Platforms Namespace

# MUNTS_0018J7D1 Field

GPIO channel number for Grove DC Motor Driver Connector `J7` pin `D1`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int J7D1 = 18
```

### Field Value

Type: Int32

## See Also

### Reference

MUNTS_0018 Class
IO.Objects.RemoteIO.Platforms Namespace

# MUNTS_0018J7DIR Field

GPIO channel number for Grove DC Motor Driver Connector `J7` pin `D1`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int J7DIR = 18
```

### Field Value

Type: Int32

## Remarks

`J7DIR` controls the motor **direction**.

## See Also

### Reference

MUNTS_0018 Class
IO.Objects.RemoteIO.Platforms Namespace

# MUNTS_0018J7PWM Field

PWM channel number for Grove DC Motor Driver Connector `J7` pin `D0`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int J7PWM = 1
```

### Field Value

Type: Int32

## Remarks

`J7PWM` controls the motor **speed**.

## See Also

### Reference

MUNTS_0018 Class
IO.Objects.RemoteIO.Platforms Namespace

# MUNTS_0018J9I2C Field

$I^2C$ bus channel number for Grove $I^2C$ Connector `J9`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int J9I2C = 0
```

### Field Value

Type: Int32

## See Also

### Reference

MUNTS_0018 Class
IO.Objects.RemoteIO.Platforms Namespace

# MUNTS_0018SW1 Field

GPIO channel number for the on-board momentary switch `SW1`. Cannot be configured as an output.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int SW1 = 6
```

### Field Value

Type: Int32

## Remarks

This GPIO pin cannot be configured as an output.

## See Also

### Reference

MUNTS_0018 Class
IO.Objects.RemoteIO.Platforms Namespace

# MUNTS_0018USERLED Field

GPIO channel number for the remote Raspberry Pi user LED. Cannot be configured as an input.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int USERLED = 0
```

### Field Value

Type: Int32

## See Also

### Reference

MUNTS_0018 Class
IO.Objects.RemoteIO.Platforms Namespace

# PocketBeagle Class

I/O resources (channel numbers) available on a BeagleBone Remote I/O Protocol Server

## Inheritance Hierarchy

SystemObject IO.Objects.RemoteIO.PlatformsPocketBeagle

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static class PocketBeagle
```

The PocketBeagle type exposes the following members.

## Fields

| | Name | Description |
|---|---|---|
| | AIN0 | Analog input channel number for the remote PocketBeagle 1.8V analog input `AIN0` at pin `P1.19`. |
| | AIN1 | Analog input channel number for the remote PocketBeagle 1.8V analog input `AIN1` at pin `P1.21`. |
| | AIN2 | Analog input channel number for the |

| | | |
|---|---|---|
| | | remote PocketBeagle 1.8V analog input `AIN2` at pin `P1.23`. |
| S | AIN3 | Analog input channel number for the remote PocketBeagle 1.8V analog input `AIN3` at pin `P1.24`. |
| S | AIN4 | Analog input channel number for the remote PocketBeagle 1.8V analog input `AIN4` at pin `P1.25`. |
| S | AIN5 | Analog input channel number for the remote PocketBeagle 3.6V analog input `AIN5` at pin `P2.35`. |
| S | AIN6 | Analog input channel number for the remote PocketBeagle 3.6V analog input `AIN6` at pin `P1.2`. |
| S | AIN7 | Analog input channel number for the remote PocketBeagle 1.8V analog input `AIN7` at pin `P2.36`. |
| S | GPIO110 | GPIO channel number for the remote PocketBeagle GPIO pin `GPIO110` at pin `P1.36`. |
| S | GPIO111 | GPIO channel number for the remote PocketBeagle GPIO pin `GPIO111` at pin `P1.33`. |
| S | GPIO112 | GPIO channel number for the remote PocketBeagle GPIO pin `GPIO112` at pin `P2.32`. |
| S | | |

| | | |
|---|---|---|
| | GPIO113 | GPIO channel number for the remote PocketBeagle GPIO pin `GPIO113` at pin `P2.30`. |
| S | GPIO114 | GPIO channel number for the remote PocketBeagle GPIO pin `GPIO114` at pin `P1.31`. |
| S | GPIO115 | GPIO channel number for the remote PocketBeagle GPIO pin `GPIO115` at pin `P2.34`. |
| S | GPIO116 | GPIO channel number for the remote PocketBeagle GPIO pin `GPIO116` at pin `P2.28`. |
| S | GPIO117 | GPIO channel number for the remote PocketBeagle GPIO pin `GPIO117` at pin `P1.29`. |
| S | GPIO12 | GPIO channel number for the remote PocketBeagle GPIO pin `GPIO12` at pin `P1.26`. |
| S | GPIO13 | GPIO channel number for the remote PocketBeagle GPIO pin `GPIO13` at pin `P1.28`. |
| S | GPIO14 | GPIO channel number for the remote PocketBeagle GPIO pin `GPIO14` at pin `P2.11`. |
| S | GPIO15 | GPIO channel number for the remote PocketBeagle GPIO pin `GPIO15` at pin `P2.9`. |

| | | |
|---|---|---|
| S | GPIO19 | GPIO channel number for the remote PocketBeagle GPIO pin `GPIO19` at pin `P2.31`. |
| S | GPIO2 | GPIO channel number for the remote PocketBeagle GPIO pin `GPIO2` at pin `P1.8`. |
| S | GPIO20 | GPIO channel number for the remote PocketBeagle GPIO pin `GPIO20` at pin `P1.20`. |
| S | GPIO23 | GPIO channel number for the remote PocketBeagle GPIO pin `GPIO23` at pin `P2.3`. |
| S | GPIO26 | GPIO channel number for the remote PocketBeagle GPIO pin `GPIO26` at pin `P1.34`. |
| S | GPIO27 | GPIO channel number for the remote PocketBeagle GPIO pin `GPIO27` at pin `P2.19`. |
| S | GPIO3 | GPIO channel number for the remote PocketBeagle GPIO pin `GPIO3` at pin `P1.10`. |
| S | GPIO30 | GPIO channel number for the remote PocketBeagle GPIO pin `GPIO30` at pin `P2.5`. |
| S | GPIO31 | GPIO channel number for the remote PocketBeagle GPIO pin `GPIO31` at pin `P2.7`. |

| | | |
|---|---|---|
| S | GPIO4 | GPIO channel number for the remote PocketBeagle GPIO pin `GPIO4` at pin `P1.12`. |
| S | GPIO40 | GPIO channel number for the remote PocketBeagle GPIO pin `GPIO40` at pin `P2.27`. |
| S | GPIO41 | GPIO channel number for the remote PocketBeagle GPIO pin `GPIO41` at pin `P2.25`. |
| S | GPIO42 | GPIO channel number for the remote PocketBeagle GPIO pin `GPIO42` at pin `P1.32`. |
| S | GPIO43 | GPIO channel number for the remote PocketBeagle GPIO pin `GPIO43` at pin `P1.30`. |
| S | GPIO44 | GPIO channel number for the remote PocketBeagle GPIO pin `GPIO44` at pin `P2.24`. |
| S | GPIO45 | GPIO channel number for the remote PocketBeagle GPIO pin `GPIO45` at pin `P2.33`. |
| S | GPIO46 | GPIO channel number for the remote PocketBeagle GPIO pin `GPIO46` at pin `P2.22`. |
| S | GPIO47 | GPIO channel number for the remote PocketBeagle GPIO pin `GPIO47` at pin `P2.18`. |

| | | |
|---|---|---|
| S | GPIO5 | GPIO channel number for the remote PocketBeagle GPIO pin `GPIO5` at pin `P1.6`. |
| S | GPIO50 | GPIO channel number for the remote PocketBeagle GPIO pin `GPIO50` at pin `P2.1`. |
| S | GPIO52 | GPIO channel number for the remote PocketBeagle GPIO pin `GPIO52` at pin `P2.10`. |
| S | GPIO57 | GPIO channel number for the remote PocketBeagle GPIO pin `GPIO57` at pin `P2.6`. |
| S | GPIO58 | GPIO channel number for the remote PocketBeagle GPIO pin `GPIO58` at pin `P2.4`. |
| S | GPIO59 | GPIO channel number for the remote PocketBeagle GPIO pin `GPIO59` at pin `P2.2`. |
| S | GPIO60 | GPIO channel number for the remote PocketBeagle GPIO pin `GPIO60` at pin `P2.8`. |
| S | GPIO64 | GPIO channel number for the remote PocketBeagle GPIO pin `GPIO64` at pin `P2.20`. |
| S | GPIO65 | GPIO channel number for the remote PocketBeagle GPIO pin `GPIO65` at pin |

| | | |
|---|---|---|
| | | `P2.17`. |
| s | GPIO7 | GPIO channel number for the remote PocketBeagle GPIO pin `GPIO7` at pin `P2.29`. |
| s | GPIO86 | GPIO channel number for the remote PocketBeagle GPIO pin `GPIO86` at pin `P2.35`. |
| s | GPIO87 | GPIO channel number for the remote PocketBeagle GPIO pin `GPIO87` at pin `P1.2`. |
| s | GPIO88 | GPIO channel number for the remote PocketBeagle GPIO pin `GPIO88` at pin `P1.35`. |
| s | GPIO89 | GPIO channel number for the remote PocketBeagle GPIO pin `GPIO89` at pin `P1.4`. |
| s | I2C1 | $I^2C$ bus channel number for the remote BeagleBone $I^2C$ bus `I²C` at pins `P2.9` and `P2.11`. |
| s | I2C2 | $I^2C$ bus channel number for the remote BeagleBone $I^2C$ bus `I²C` at pins `P1.26` and `P1.28`. |
| s | PWM0_0 | PWM output channel number for the remote PocketBeagle PWM output `PWM0_0` at pin `P1.36` |
| s | PWM2_0 | PWM output channel number for the |

| | | |
|---|---|---|
| | | remote PocketBeagle PWM output `PWM2_0` at pin `P2.1` |
| | SPI0_0 | SPI slave channel number for the remote PocketBeagle SPI slave select `SPI0_0` at pin `P1.6` |
| | SPI1_1 | SPI slave channel number for the remote PocketBeagle SPI slave select `SPI1_1` at pin `P2.31` |
| | USERLED | GPIO channel number for the remote PocketBeagle user LED. |

Top

## See Also

### Reference

IO.Objects.RemoteIO.Platforms Namespace

# PocketBeagle Fields

The PocketBeagle type exposes the following members.

## Fields

| | Name | Description |
|---|---|---|
| S | AIN0 | Analog input channel number for the remote PocketBeagle 1.8V analog input `AIN0` at pin `P1.19`. |
| S | AIN1 | Analog input channel number for the remote PocketBeagle 1.8V analog input `AIN1` at pin `P1.21`. |
| S | AIN2 | Analog input channel number for the remote PocketBeagle 1.8V analog input `AIN2` at pin `P1.23`. |
| S | AIN3 | Analog input channel number for the remote PocketBeagle 1.8V analog input `AIN3` at pin `P1.24`. |
| S | AIN4 | Analog input channel number for the remote PocketBeagle 1.8V analog input `AIN4` at pin `P1.25`. |
| S | AIN5 | Analog input channel number for the remote PocketBeagle 3.6V analog input `AIN5` at pin `P2.35`. |

| | | |
|---|---|---|
| S | AIN6 | Analog input channel number for the remote PocketBeagle 3.6V analog input `AIN6` at pin `P1.2`. |
| S | AIN7 | Analog input channel number for the remote PocketBeagle 1.8V analog input `AIN7` at pin `P2.36`. |
| S | GPIO110 | GPIO channel number for the remote PocketBeagle GPIO pin `GPIO110` at pin `P1.36`. |
| S | GPIO111 | GPIO channel number for the remote PocketBeagle GPIO pin `GPIO111` at pin `P1.33`. |
| S | GPIO112 | GPIO channel number for the remote PocketBeagle GPIO pin `GPIO112` at pin `P2.32`. |
| S | GPIO113 | GPIO channel number for the remote PocketBeagle GPIO pin `GPIO113` at pin `P2.30`. |
| S | GPIO114 | GPIO channel number for the remote PocketBeagle GPIO pin `GPIO114` at pin `P1.31`. |
| S | GPIO115 | GPIO channel number for the remote PocketBeagle GPIO pin `GPIO115` at pin `P2.34`. |
| S | GPIO116 | GPIO channel number for the remote PocketBeagle GPIO pin `GPIO116` at pin `P2.28`. |

| | | |
|---|---|---|
| S | GPIO117 | GPIO channel number for the remote PocketBeagle GPIO pin `GPIO117` at pin `P1.29`. |
| S | GPIO12 | GPIO channel number for the remote PocketBeagle GPIO pin `GPIO12` at pin `P1.26`. |
| S | GPIO13 | GPIO channel number for the remote PocketBeagle GPIO pin `GPIO13` at pin `P1.28`. |
| S | GPIO14 | GPIO channel number for the remote PocketBeagle GPIO pin `GPIO14` at pin `P2.11`. |
| S | GPIO15 | GPIO channel number for the remote PocketBeagle GPIO pin `GPIO15` at pin `P2.9`. |
| S | GPIO19 | GPIO channel number for the remote PocketBeagle GPIO pin `GPIO19` at pin `P2.31`. |
| S | GPIO2 | GPIO channel number for the remote PocketBeagle GPIO pin `GPIO2` at pin `P1.8`. |
| S | GPIO20 | GPIO channel number for the remote PocketBeagle GPIO pin `GPIO20` at pin `P1.20`. |
| S | GPIO23 | GPIO channel number for the remote PocketBeagle GPIO pin `GPIO23` at pin `P2.3`. |

| | | |
|---|---|---|
| S | GPIO26 | GPIO channel number for the remote PocketBeagle GPIO pin `GPIO26` at pin `P1.34`. |
| S | GPIO27 | GPIO channel number for the remote PocketBeagle GPIO pin `GPIO27` at pin `P2.19`. |
| S | GPIO3 | GPIO channel number for the remote PocketBeagle GPIO pin `GPIO3` at pin `P1.10`. |
| S | GPIO30 | GPIO channel number for the remote PocketBeagle GPIO pin `GPIO30` at pin `P2.5`. |
| S | GPIO31 | GPIO channel number for the remote PocketBeagle GPIO pin `GPIO31` at pin `P2.7`. |
| S | GPIO4 | GPIO channel number for the remote PocketBeagle GPIO pin `GPIO4` at pin `P1.12`. |
| S | GPIO40 | GPIO channel number for the remote PocketBeagle GPIO pin `GPIO40` at pin `P2.27`. |
| S | GPIO41 | GPIO channel number for the remote PocketBeagle GPIO pin `GPIO41` at pin `P2.25`. |
| S | GPIO42 | GPIO channel number for the remote PocketBeagle GPIO pin `GPIO42` at pin `P1.32`. |

| | | |
|---|---|---|
| S | GPIO43 | GPIO channel number for the remote PocketBeagle GPIO pin `GPIO43` at pin `P1.30`. |
| S | GPIO44 | GPIO channel number for the remote PocketBeagle GPIO pin `GPIO44` at pin `P2.24`. |
| S | GPIO45 | GPIO channel number for the remote PocketBeagle GPIO pin `GPIO45` at pin `P2.33`. |
| S | GPIO46 | GPIO channel number for the remote PocketBeagle GPIO pin `GPIO46` at pin `P2.22`. |
| S | GPIO47 | GPIO channel number for the remote PocketBeagle GPIO pin `GPIO47` at pin `P2.18`. |
| S | GPIO5 | GPIO channel number for the remote PocketBeagle GPIO pin `GPIO5` at pin `P1.6`. |
| S | GPIO50 | GPIO channel number for the remote PocketBeagle GPIO pin `GPIO50` at pin `P2.1`. |
| S | GPIO52 | GPIO channel number for the remote PocketBeagle GPIO pin `GPIO52` at pin `P2.10`. |
| S | GPIO57 | GPIO channel number for the remote PocketBeagle GPIO pin `GPIO57` at pin |

| | | |
|---|---|---|
| | | `P2.6`. |
| S | GPIO58 | GPIO channel number for the remote PocketBeagle GPIO pin `GPIO58` at pin `P2.4`. |
| S | GPIO59 | GPIO channel number for the remote PocketBeagle GPIO pin `GPIO59` at pin `P2.2`. |
| S | GPIO60 | GPIO channel number for the remote PocketBeagle GPIO pin `GPIO60` at pin `P2.8`. |
| S | GPIO64 | GPIO channel number for the remote PocketBeagle GPIO pin `GPIO64` at pin `P2.20`. |
| S | GPIO65 | GPIO channel number for the remote PocketBeagle GPIO pin `GPIO65` at pin `P2.17`. |
| S | GPIO7 | GPIO channel number for the remote PocketBeagle GPIO pin `GPIO7` at pin `P2.29`. |
| S | GPIO86 | GPIO channel number for the remote PocketBeagle GPIO pin `GPIO86` at pin `P2.35`. |
| S | GPIO87 | GPIO channel number for the remote PocketBeagle GPIO pin `GPIO87` at pin `P1.2`. |
| S | GPIO88 | GPIO channel number for the remote |

| | | |
|---|---|---|
| | | PocketBeagle GPIO pin `GPIO88` at pin `P1.35`. |
| s | GPIO89 | GPIO channel number for the remote PocketBeagle GPIO pin `GPIO89` at pin `P1.4`. |
| s | I2C1 | $I^2C$ bus channel number for the remote BeagleBone $I^2C$ bus `I²C` at pins `P2.9` and `P2.11`. |
| s | I2C2 | $I^2C$ bus channel number for the remote BeagleBone $I^2C$ bus `I²C` at pins `P1.26` and `P1.28`. |
| s | PWM0_0 | PWM output channel number for the remote PocketBeagle PWM output `PWM0_0` at pin `P1.36` |
| s | PWM2_0 | PWM output channel number for the remote PocketBeagle PWM output `PWM2_0` at pin `P2.1` |
| s | SPI0_0 | SPI slave channel number for the remote PocketBeagle SPI slave select `SPI0_0` at pin `P1.6` |
| s | SPI1_1 | SPI slave channel number for the remote PocketBeagle SPI slave select `SPI1_1` at pin `P2.31` |
| s | USERLED | GPIO channel number for the remote PocketBeagle user LED. |

Top

## See Also

### Reference

PocketBeagle Class

IO.Objects.RemoteIO.Platforms Namespace

# PocketBeagleAIN0 Field

Analog input channel number for the remote PocketBeagle 1.8V analog input `AIN0` at pin `P1.19`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int AIN0 = 0
```

### Field Value

Type: Int32

## See Also

### Reference

PocketBeagle Class
IO.Objects.RemoteIO.Platforms Namespace

# PocketBeagleAIN1 Field

Analog input channel number for the remote PocketBeagle 1.8V analog input `AIN1` at pin `P1.21`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public const int AIN1 = 1
```

### Field Value

Type: Int32

## See Also

### Reference

PocketBeagle Class
IO.Objects.RemoteIO.Platforms Namespace

# PocketBeagleAIN2 Field

Analog input channel number for the remote PocketBeagle 1.8V analog input `AIN2` at pin `P1.23`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public const int AIN2 = 2
```

### Field Value

Type: Int32

## See Also

### Reference

PocketBeagle Class
IO.Objects.RemoteIO.Platforms Namespace

# PocketBeagleAIN3 Field

Analog input channel number for the remote PocketBeagle 1.8V analog input `AIN3` at pin `P1.24`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int AIN3 = 3
```

### Field Value

Type: Int32

## See Also

### Reference

PocketBeagle Class
IO.Objects.RemoteIO.Platforms Namespace

# PocketBeagleAIN4 Field

Analog input channel number for the remote PocketBeagle 1.8V analog input `AIN4` at pin `P1.25`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int AIN4 = 4
```

### Field Value

Type: Int32

## See Also

### Reference

PocketBeagle Class
IO.Objects.RemoteIO.Platforms Namespace

# PocketBeagleAIN5 Field

Analog input channel number for the remote PocketBeagle 3.6V analog input `AIN5` at pin `P2.35`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int AIN5 = 5
```

### Field Value

Type: Int32

## Remarks

Conflicts with `GPIO86`.

## See Also

### Reference

PocketBeagle Class
IO.Objects.RemoteIO.Platforms Namespace

# PocketBeagleAIN6 Field

Analog input channel number for the remote PocketBeagle 3.6V analog input `AIN6` at pin `P1.2`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public const int AIN6 = 6
```

### Field Value

Type: Int32

## Remarks

Conflicts with `GPIO87`.

## See Also

### Reference

PocketBeagle Class
IO.Objects.RemoteIO.Platforms Namespace

# PocketBeagleAIN7 Field

Analog input channel number for the remote PocketBeagle 1.8V analog input `AIN7` at pin `P2.36`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public const int AIN7 = 7
```

### Field Value

Type: Int32

## See Also

### Reference

PocketBeagle Class
IO.Objects.RemoteIO.Platforms Namespace

# PocketBeagleGPIO110 Field

GPIO channel number for the remote PocketBeagle GPIO pin `GPIO110` at pin `P1.36`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO110 = 110
```

### Field Value

Type: Int32

## Remarks

Conflicts with `PWM0:0`.

## See Also

### Reference

PocketBeagle Class
IO.Objects.RemoteIO.Platforms Namespace

# PocketBeagleGPIO111 Field

GPIO channel number for the remote PocketBeagle GPIO pin `GPIO111` at pin `P1.33`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO111 = 111
```

### Field Value

Type: Int32

## See Also

### Reference

PocketBeagle Class
IO.Objects.RemoteIO.Platforms Namespace

# PocketBeagleGPIO112 Field

GPIO channel number for the remote PocketBeagle GPIO pin `GPIO112` at pin `P2.32`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO112 = 112
```

### Field Value

Type: Int32

## See Also

### Reference

PocketBeagle Class
IO.Objects.RemoteIO.Platforms Namespace

# PocketBeagleGPIO113 Field

GPIO channel number for the remote PocketBeagle GPIO pin `GPIO113` at pin `P2.30`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO113 = 113
```

### Field Value

Type: Int32

## See Also

### Reference

PocketBeagle Class
IO.Objects.RemoteIO.Platforms Namespace

# PocketBeagleGPIO114 Field

GPIO channel number for the remote PocketBeagle GPIO pin `GPIO114` at pin `P1.31`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO114 = 114
```

### Field Value

Type: Int32

## See Also

### Reference

PocketBeagle Class
IO.Objects.RemoteIO.Platforms Namespace

# PocketBeagleGPIO115 Field

GPIO channel number for the remote PocketBeagle GPIO pin `GPIO115` at pin `P2.34`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO115 = 115
```

### Field Value

Type: Int32

## See Also

### Reference

PocketBeagle Class
IO.Objects.RemoteIO.Platforms Namespace

# PocketBeagleGPIO116 Field

GPIO channel number for the remote PocketBeagle GPIO pin `GPIO116` at pin `P2.28`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO116 = 116
```

### Field Value

Type: Int32

## See Also

### Reference

PocketBeagle Class
IO.Objects.RemoteIO.Platforms Namespace

# PocketBeagleGPIO117 Field

GPIO channel number for the remote PocketBeagle GPIO pin `GPIO117` at pin `P1.29`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO117 = 117
```

### Field Value

Type: Int32

## See Also

### Reference

PocketBeagle Class
IO.Objects.RemoteIO.Platforms Namespace

# PocketBeagleGPIO12 Field

GPIO channel number for the remote PocketBeagle GPIO pin `GPIO12` at pin `P1.26`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO12 = 12
```

### Field Value

Type: Int32

## Remarks

Conflicts with `I2C2 SDA`.

## See Also

### Reference

PocketBeagle Class
IO.Objects.RemoteIO.Platforms Namespace

# PocketBeagleGPIO13 Field

GPIO channel number for the remote PocketBeagle GPIO pin `GPIO13` at pin `P1.28`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO13 = 13
```

### Field Value

Type: Int32

## Remarks

Conflicts with `I2C2 SCL`.

## See Also

### Reference

PocketBeagle Class
IO.Objects.RemoteIO.Platforms Namespace

# PocketBeagleGPIO14 Field

GPIO channel number for the remote PocketBeagle GPIO pin `GPIO14` at pin `P2.11`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO14 = 14
```

### Field Value

Type: Int32

## Remarks

Conflicts with `I2C1 SDA`.

## See Also

### Reference

PocketBeagle Class
IO.Objects.RemoteIO.Platforms Namespace

# PocketBeagleGPIO15 Field

GPIO channel number for the remote PocketBeagle GPIO pin `GPIO15` at pin `P2.9`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO15 = 15
```

### Field Value

Type: Int32

## Remarks

Conflicts with `I2C1 SCL`.

## See Also

### Reference

PocketBeagle Class
IO.Objects.RemoteIO.Platforms Namespace

# PocketBeagleGPIO19 Field

GPIO channel number for the remote PocketBeagle GPIO pin `GPIO19` at pin `P2.31`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO19 = 19
```

### Field Value

Type: Int32

## Remarks

Conflicts with `SPI1 CS1`.

## See Also

### Reference

PocketBeagle Class
IO.Objects.RemoteIO.Platforms Namespace

# PocketBeagleGPIO2 Field

GPIO channel number for the remote PocketBeagle GPIO pin `GPIO2` at pin `P1.8`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO2 = 2
```

### Field Value

Type: Int32

## Remarks

Conflicts with `SPI0 SCLK`.

## See Also

### Reference

PocketBeagle Class
IO.Objects.RemoteIO.Platforms Namespace

# PocketBeagleGPIO20 Field

GPIO channel number for the remote PocketBeagle GPIO pin `GPIO20` at pin `P1.20`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO20 = 20
```

### Field Value

Type: Int32

## See Also

### Reference

PocketBeagle Class
IO.Objects.RemoteIO.Platforms Namespace

# PocketBeagleGPIO23 Field

GPIO channel number for the remote PocketBeagle GPIO pin `GPIO23` at pin `P2.3`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO23 = 23
```

### Field Value

Type: Int32

## See Also

### Reference

PocketBeagle Class
IO.Objects.RemoteIO.Platforms Namespace

# PocketBeagleGPIO26 Field

GPIO channel number for the remote PocketBeagle GPIO pin `GPIO26` at pin `P1.34`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO26 = 26
```

### Field Value

Type: Int32

## See Also

### Reference

PocketBeagle Class
IO.Objects.RemoteIO.Platforms Namespace

# PocketBeagleGPIO27 Field

GPIO channel number for the remote PocketBeagle GPIO pin `GPIO27` at pin `P2.19`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO27 = 27
```

### Field Value

Type: Int32

## See Also

### Reference

PocketBeagle Class
IO.Objects.RemoteIO.Platforms Namespace

# PocketBeagleGPIO3 Field

GPIO channel number for the remote PocketBeagle GPIO pin `GPIO3` at pin `P1.10`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO3 = 3
```

### Field Value

Type: Int32

## Remarks

Conflicts with `SPI0 MISO`.

## See Also

### Reference

PocketBeagle Class
IO.Objects.RemoteIO.Platforms Namespace

# PocketBeagleGPIO30 Field

GPIO channel number for the remote PocketBeagle GPIO pin `GPIO30` at pin `P2.5`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO30 = 30
```

### Field Value

Type: Int32

## Remarks

Conflicts with `RXD4`.

## See Also

### Reference

PocketBeagle Class
IO.Objects.RemoteIO.Platforms Namespace

# PocketBeagleGPIO31 Field

GPIO channel number for the remote PocketBeagle GPIO pin `GPIO31` at pin `P2.7`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO31 = 31
```

### Field Value

Type: Int32

## Remarks

Conflicts with `TXD4`.

## See Also

### Reference

PocketBeagle Class
IO.Objects.RemoteIO.Platforms Namespace

# PocketBeagleGPIO4 Field

GPIO channel number for the remote PocketBeagle GPIO pin `GPIO4` at pin `P1.12`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO4 = 4
```

### Field Value

Type: Int32

## Remarks

Conflicts with `SPI0 MOSI`.

## See Also

### Reference

PocketBeagle Class
IO.Objects.RemoteIO.Platforms Namespace

# PocketBeagleGPIO40 Field

GPIO channel number for the remote PocketBeagle GPIO pin `GPIO40` at pin `P2.27`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO40 = 40
```

### Field Value

Type: Int32

## Remarks

Conflicts with `SPI1 MISO`.

## See Also

### Reference

PocketBeagle Class
IO.Objects.RemoteIO.Platforms Namespace

# PocketBeagleGPIO41 Field

GPIO channel number for the remote PocketBeagle GPIO pin `GPIO41` at pin `P2.25`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO41 = 41
```

### Field Value

Type: Int32

## Remarks

Conflicts with `SPI1 MOSI`.

## See Also

### Reference

PocketBeagle Class
IO.Objects.RemoteIO.Platforms Namespace

# PocketBeagleGPIO42 Field

GPIO channel number for the remote PocketBeagle GPIO pin `GPIO42` at pin `P1.32`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO42 = 42
```

### Field Value

Type: Int32

## Remarks

Conflicts with `RXD0`.

## See Also

### Reference

PocketBeagle Class
IO.Objects.RemoteIO.Platforms Namespace

# PocketBeagleGPIO43 Field

GPIO channel number for the remote PocketBeagle GPIO pin `GPIO43` at pin `P1.30`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO43 = 43
```

### Field Value

Type: Int32

## Remarks

Conflicts with `TXD0`.

## See Also

### Reference

PocketBeagle Class
IO.Objects.RemoteIO.Platforms Namespace

# PocketBeagleGPIO44 Field

GPIO channel number for the remote PocketBeagle GPIO pin `GPIO44` at pin `P2.24`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO44 = 44
```

### Field Value

Type: Int32

## See Also

### Reference

PocketBeagle Class
IO.Objects.RemoteIO.Platforms Namespace

# PocketBeagleGPIO45 Field

GPIO channel number for the remote PocketBeagle GPIO pin `GPIO45` at pin `P2.33`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO45 = 45
```

### Field Value

Type: Int32

## See Also

### Reference

PocketBeagle Class
IO.Objects.RemoteIO.Platforms Namespace

# PocketBeagleGPIO46 Field

GPIO channel number for the remote PocketBeagle GPIO pin `GPIO46` at pin `P2.22`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO46 = 46
```

### Field Value

Type: Int32

## See Also

### Reference

PocketBeagle Class
IO.Objects.RemoteIO.Platforms Namespace

# PocketBeagleGPIO47 Field

GPIO channel number for the remote PocketBeagle GPIO pin `GPIO47` at pin `P2.18`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO47 = 47
```

### Field Value

Type: Int32

## See Also

### Reference

PocketBeagle Class
IO.Objects.RemoteIO.Platforms Namespace

# PocketBeagleGPIO5 Field

GPIO channel number for the remote PocketBeagle GPIO pin `GPIO5` at pin `P1.6`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO5 = 5
```

### Field Value

Type: Int32

## Remarks

Conflicts with `SPI0 CS0`.

## See Also

### Reference

PocketBeagle Class
IO.Objects.RemoteIO.Platforms Namespace

# PocketBeagleGPIO50 Field

GPIO channel number for the remote PocketBeagle GPIO pin `GPIO50` at pin `P2.1`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO50 = 50
```

### Field Value

Type: Int32

## Remarks

Conflicts with `PWM2:0`.

## See Also

### Reference

PocketBeagle Class
IO.Objects.RemoteIO.Platforms Namespace

# PocketBeagleGPIO52 Field

GPIO channel number for the remote PocketBeagle GPIO pin `GPIO52` at pin `P2.10`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO52 = 52
```

### Field Value

Type: Int32

## See Also

### Reference

PocketBeagle Class
IO.Objects.RemoteIO.Platforms Namespace

# PocketBeagleGPIO57 Field

GPIO channel number for the remote PocketBeagle GPIO pin `GPIO57` at pin `P2.6`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO57 = 57
```

### Field Value

Type: Int32

## See Also

### Reference

PocketBeagle Class
IO.Objects.RemoteIO.Platforms Namespace

# PocketBeagleGPIO58 Field

GPIO channel number for the remote PocketBeagle GPIO pin `GPIO58` at pin `P2.4`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO58 = 58
```

### Field Value

Type: Int32

## See Also

### Reference

PocketBeagle Class
IO.Objects.RemoteIO.Platforms Namespace

# PocketBeagleGPIO59 Field

GPIO channel number for the remote PocketBeagle GPIO pin `GPIO59` at pin `P2.2`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO59 = 59
```

### Field Value

Type: Int32

## See Also

### Reference

PocketBeagle Class
IO.Objects.RemoteIO.Platforms Namespace

# PocketBeagleGPIO60 Field

GPIO channel number for the remote PocketBeagle GPIO pin `GPIO60` at pin `P2.8`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO60 = 60
```

### Field Value

Type: Int32

## See Also

### Reference

PocketBeagle Class
IO.Objects.RemoteIO.Platforms Namespace

# PocketBeagleGPIO64 Field

GPIO channel number for the remote PocketBeagle GPIO pin `GPIO64` at pin `P2.20`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO64 = 64
```

### Field Value

Type: Int32

## See Also

### Reference

PocketBeagle Class
IO.Objects.RemoteIO.Platforms Namespace

# PocketBeagleGPIO65 Field

GPIO channel number for the remote PocketBeagle GPIO pin `GPIO65` at pin `P2.17`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO65 = 65
```

### Field Value

Type: Int32

## See Also

### Reference

PocketBeagle Class
IO.Objects.RemoteIO.Platforms Namespace

# PocketBeagleGPIO7 Field

GPIO channel number for the remote PocketBeagle GPIO pin `GPIO7` at pin `P2.29`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO7 = 7
```

### Field Value

Type: Int32

## Remarks

Conflicts with `SPI1 SCLK`.

## See Also

### Reference

PocketBeagle Class
IO.Objects.RemoteIO.Platforms Namespace

# PocketBeagleGPIO86 Field

GPIO channel number for the remote PocketBeagle GPIO pin `GPIO86` at pin `P2.35`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO86 = 86
```

### Field Value

Type: Int32

## Remarks

Conflicts with `AIN5`.

## See Also

### Reference

PocketBeagle Class
IO.Objects.RemoteIO.Platforms Namespace

# PocketBeagleGPIO87 Field

GPIO channel number for the remote PocketBeagle GPIO pin `GPIO87` at pin `P1.2`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO87 = 87
```

### Field Value

Type: Int32

## Remarks

Conflicts with `AIN6`.

## See Also

### Reference

PocketBeagle Class
IO.Objects.RemoteIO.Platforms Namespace

# PocketBeagleGPIO88 Field

GPIO channel number for the remote PocketBeagle GPIO pin `GPIO88` at pin `P1.35`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO88 = 88
```

### Field Value

Type: Int32

## See Also

### Reference

PocketBeagle Class
IO.Objects.RemoteIO.Platforms Namespace

# PocketBeagleGPIO89 Field

GPIO channel number for the remote PocketBeagle GPIO pin `GPIO89` at pin `P1.4`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO89 = 89
```

### Field Value

Type: Int32

## See Also

### Reference

PocketBeagle Class
IO.Objects.RemoteIO.Platforms Namespace

# PocketBeagleI2C1 Field

$I^2C$ bus channel number for the remote BeagleBone $I^2C$ bus `I²C` at pins `P2.9` and `P2.11`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int I2C1 = 1
```

### Field Value

Type: Int32

## Remarks

Conflicts with `GPIO14` and `GPIO15`.

## See Also

### Reference

PocketBeagle Class
IO.Objects.RemoteIO.Platforms Namespace

# PocketBeagleI2C2 Field

$I^2C$ bus channel number for the remote BeagleBone $I^2C$ bus `I²C` at pins `P1.26` and `P1.28`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int I2C2 = 2
```

### Field Value

Type: Int32

## Remarks

Conflicts with `GPIO12` and `GPIO13`.

## See Also

### Reference

PocketBeagle Class
IO.Objects.RemoteIO.Platforms Namespace

# PocketBeaglePWM0_0 Field

PWM output channel number for the remote PocketBeagle PWM output `PWM0_0` at pin `P1.36`

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int PWM0_0 = 0
```

### Field Value

Type: Int32

## Remarks

Conflicts with `GPIO110`.

## See Also

### Reference

PocketBeagle Class
IO.Objects.RemoteIO.Platforms Namespace

# PocketBeaglePWM2_0 Field

PWM output channel number for the remote PocketBeagle PWM output `PWM2_0` at pin `P2.1`

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int PWM2_0 = 1
```

### Field Value

Type: Int32

## Remarks

Conflicts with `GPIO50`.

## See Also

### Reference

PocketBeagle Class
IO.Objects.RemoteIO.Platforms Namespace

# PocketBeagleSPI0_0 Field

SPI slave channel number for the remote PocketBeagle SPI slave select `SPI0_0` at pin `P1.6`

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int SPI0_0 = 0
```

### Field Value

Type: Int32

## Remarks

Conflicts with `GPIO5`.

## See Also

### Reference

PocketBeagle Class
IO.Objects.RemoteIO.Platforms Namespace

# PocketBeagleSPI1_1 Field

SPI slave channel number for the remote PocketBeagle SPI slave select `SPI1_1` at pin `P2.31`

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int SPI1_1 = 1
```

### Field Value

Type: Int32

## Remarks

Conflicts with `GPIO19`.

## See Also

### Reference

PocketBeagle Class
IO.Objects.RemoteIO.Platforms Namespace

# PocketBeagleUSERLED Field

GPIO channel number for the remote PocketBeagle user LED.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int USERLED = 0
```

### Field Value

Type: Int32

## Remarks

This GPIO channel cannot be configured as an input.

## See Also

### Reference

PocketBeagle Class
IO.Objects.RemoteIO.Platforms Namespace

# RaspberryPi Class

I/O resources (channel numbers) available on a Raspberry Pi Remote I/O Protocol Server

## Inheritance Hierarchy

SystemObject IO.Objects.RemoteIO.PlatformsRaspberryPi

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static class RaspberryPi
```

The RaspberryPi type exposes the following members.

## Fields

| | Name | Description |
|---|---|---|
| S | AIN0 | Analog input channel number for the remote Raspberry Pi analog input `AIN0`. |
| S | AIN1 | Analog input channel number for the remote Raspberry Pi analog input `AIN1`. |
| S | GPIO10 | GPIO channel number for the remote Raspberry Pi GPIO pin `GPIO10`. |

| | | |
|---|---|---|
| S | GPIO11 | GPIO channel number for the remote Raspberry Pi GPIO pin `GPIO11`. |
| S | GPIO12 | GPIO channel number for the remote Raspberry Pi GPIO pin `GPIO12`. |
| S | GPIO13 | GPIO channel number for the remote Raspberry Pi GPIO pin `GPIO13`. |
| S | GPIO14 | GPIO channel number for the remote Raspberry Pi GPIO pin `GPIO14`. |
| S | GPIO15 | GPIO channel number for the remote Raspberry Pi GPIO pin `GPIO15`. |
| S | GPIO16 | GPIO channel number for the remote Raspberry Pi GPIO pin `GPIO16`. |
| S | GPIO17 | GPIO channel number for the remote Raspberry Pi GPIO pin `GPIO17`. |
| S | GPIO18 | GPIO channel number for the remote Raspberry Pi GPIO pin `GPIO18`. |
| S | GPIO19 | GPIO channel number for the remote Raspberry Pi GPIO pin `GPIO19`. |
| S | GPIO2 | GPIO channel number for the remote Raspberry Pi GPIO pin `GPIO2`. |
| S | GPIO20 | GPIO channel number for the remote Raspberry Pi GPIO pin `GPIO20`. |
| S | GPIO21 | GPIO channel number for the remote Raspberry Pi GPIO pin `GPIO21`. |

| | | |
|---|---|---|
| S | GPIO22 | GPIO channel number for the remote Raspberry Pi GPIO pin `GPIO22`. |
| S | GPIO23 | GPIO channel number for the remote Raspberry Pi GPIO pin `GPIO23`. |
| S | GPIO24 | GPIO channel number for the remote Raspberry Pi GPIO pin `GPIO24`. |
| S | GPIO25 | GPIO channel number for the remote Raspberry Pi GPIO pin `GPIO25`. |
| S | GPIO26 | GPIO channel number for the remote Raspberry Pi GPIO pin `GPIO26`. |
| S | GPIO27 | GPIO channel number for the remote Raspberry Pi GPIO pin `GPIO27`. |
| S | GPIO3 | GPIO channel number for the remote Raspberry Pi GPIO pin `GPIO3`. |
| S | GPIO4 | GPIO channel number for the remote Raspberry Pi GPIO pin `GPIO4`. |
| S | GPIO5 | GPIO channel number for the remote Raspberry Pi GPIO pin `GPIO5`. |
| S | GPIO6 | GPIO channel number for the remote Raspberry Pi GPIO pin `GPIO6`. |
| S | GPIO7 | GPIO channel number for the remote Raspberry Pi GPIO pin `GPIO7`. |
| S | GPIO8 | GPIO channel number for the remote Raspberry Pi GPIO pin `GPIO8`. |

| | | |
|---|---|---|
| S | GPIO9 | GPIO channel number for the remote Raspberry Pi GPIO pin `GPIO9`. |
| S | I2C1 | $I^2C$ bus channel number for the remote Raspberry Pi $I^2C$ bus controller `I2C1`. |
| S | PWM0 | PWM output channel number for the remote Raspberry Pi PWM output `PWM0`. |
| S | PWM1 | PWM output channel number for the remote Raspberry Pi PWM output `PWM1`. |
| S | SPI0 | SPI slave device channel number for the remote Raspberry Pi SPI slave device `/dev/spidev0.0`. |
| S | SPI1 | SPI slave device channel number for the remote Raspberry Pi SPI slave device `/dev/spidev0.1`. |
| S | USERLED | GPIO channel number for the remote Raspberry Pi user LED. |

Top

## See Also

### Reference

IO.Objects.RemoteIO.Platforms Namespace

# RaspberryPi Fields

The RaspberryPi type exposes the following members.

## Fields

| | Name | Description |
|---|---|---|
| s | AIN0 | Analog input channel number for the remote Raspberry Pi analog input `AIN0`. |
| s | AIN1 | Analog input channel number for the remote Raspberry Pi analog input `AIN1`. |
| s | GPIO10 | GPIO channel number for the remote Raspberry Pi GPIO pin `GPIO10`. |
| s | GPIO11 | GPIO channel number for the remote Raspberry Pi GPIO pin `GPIO11`. |
| s | GPIO12 | GPIO channel number for the remote Raspberry Pi GPIO pin `GPIO12`. |
| s | GPIO13 | GPIO channel number for the remote Raspberry Pi GPIO pin `GPIO13`. |
| s | GPIO14 | GPIO channel number for the remote Raspberry Pi GPIO pin `GPIO14`. |
| s | GPIO15 | GPIO channel number for the remote Raspberry Pi GPIO pin `GPIO15`. |
| s | | |

| | | |
|---|---|---|
| | GPIO16 | GPIO channel number for the remote Raspberry Pi GPIO pin `GPIO16`. |
| S | GPIO17 | GPIO channel number for the remote Raspberry Pi GPIO pin `GPIO17`. |
| S | GPIO18 | GPIO channel number for the remote Raspberry Pi GPIO pin `GPIO18`. |
| S | GPIO19 | GPIO channel number for the remote Raspberry Pi GPIO pin `GPIO19`. |
| S | GPIO2 | GPIO channel number for the remote Raspberry Pi GPIO pin `GPIO2`. |
| S | GPIO20 | GPIO channel number for the remote Raspberry Pi GPIO pin `GPIO20`. |
| S | GPIO21 | GPIO channel number for the remote Raspberry Pi GPIO pin `GPIO21`. |
| S | GPIO22 | GPIO channel number for the remote Raspberry Pi GPIO pin `GPIO22`. |
| S | GPIO23 | GPIO channel number for the remote Raspberry Pi GPIO pin `GPIO23`. |
| S | GPIO24 | GPIO channel number for the remote Raspberry Pi GPIO pin `GPIO24`. |
| S | GPIO25 | GPIO channel number for the remote Raspberry Pi GPIO pin `GPIO25`. |
| S | GPIO26 | GPIO channel number for the remote Raspberry Pi GPIO pin `GPIO26`. |

| | | |
|---|---|---|
| S | GPIO27 | GPIO channel number for the remote Raspberry Pi GPIO pin `GPIO27`. |
| S | GPIO3 | GPIO channel number for the remote Raspberry Pi GPIO pin `GPIO3`. |
| S | GPIO4 | GPIO channel number for the remote Raspberry Pi GPIO pin `GPIO4`. |
| S | GPIO5 | GPIO channel number for the remote Raspberry Pi GPIO pin `GPIO5`. |
| S | GPIO6 | GPIO channel number for the remote Raspberry Pi GPIO pin `GPIO6`. |
| S | GPIO7 | GPIO channel number for the remote Raspberry Pi GPIO pin `GPIO7`. |
| S | GPIO8 | GPIO channel number for the remote Raspberry Pi GPIO pin `GPIO8`. |
| S | GPIO9 | GPIO channel number for the remote Raspberry Pi GPIO pin `GPIO9`. |
| S | I2C1 | $I^2C$ bus channel number for the remote Raspberry Pi $I^2C$ bus controller `I2C1`. |
| S | PWM0 | PWM output channel number for the remote Raspberry Pi PWM output `PWM0`. |
| S | PWM1 | PWM output channel number for the remote Raspberry Pi PWM output `PWM1`. |
| S | SPI0 | SPI slave device channel number for the remote Raspberry Pi SPI slave device `/dev/spidev0.0`. |

| | | |
|---|---|---|
| | SPI1 | SPI slave device channel number for the remote Raspberry Pi SPI slave device `/dev/spidev0.1`. |
| | USERLED | GPIO channel number for the remote Raspberry Pi user LED. |

Top

## See Also

### Reference

RaspberryPi Class

IO.Objects.RemoteIO.Platforms Namespace

# RaspberryPiAIN0 Field

Analog input channel number for the remote Raspberry Pi analog input `AIN0`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int AIN0 = 0
```

### Field Value

Type: Int32

## Remarks

This analog input is only available if a Mikroelektronika Pi 3 Click Shield (MIKROE-2756) is installed and the following device tree overlay command added to config.txt:

Copy

```
dtoverlay=Pi3ClickShield
```

## See Also

### Reference

RaspberryPi Class
IO.Objects.RemoteIO.Platforms Namespace

# RaspberryPiAIN1 Field

Analog input channel number for the remote Raspberry Pi analog input `AIN1`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int AIN1 = 1
```

### Field Value

Type: Int32

## Remarks

This analog input is only available if a Mikroelektronika Pi 3 Click Shield (MIKROE-2756) is installed and the following device tree overlay command added to config.txt:

Copy

```
dtoverlay=Pi3ClickShield
```

## See Also

### Reference

RaspberryPi Class
IO.Objects.RemoteIO.Platforms Namespace

# RaspberryPiGPIO10 Field

GPIO channel number for the remote Raspberry Pi GPIO pin `GPIO10`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO10 = 10
```

### Field Value

Type: Int32

## Remarks

This GPIO channel is normally unusable because `GPIO10` is mapped to SPI signal `SPI0_MOSI`.

## See Also

### Reference

RaspberryPi Class
IO.Objects.RemoteIO.Platforms Namespace

# RaspberryPiGPIO11 Field

GPIO channel number for the remote Raspberry Pi GPIO pin `GPIO11`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO11 = 11
```

### Field Value

Type: Int32

## Remarks

This GPIO channel is normally unusable because `GPIO11` is mapped to SPI signal `SPI0_SCLK`.

## See Also

### Reference

RaspberryPi Class
IO.Objects.RemoteIO.Platforms Namespace

# RaspberryPiGPIO12 Field

GPIO channel number for the remote Raspberry Pi GPIO pin `GPIO12`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO12 = 12
```

### Field Value

Type: Int32

## Remarks

This GPIO channel is not usable when the Remote I/O server is running on an original Raspberry Pi 1 Model A or B with a 26-pin expansion header.

## See Also

### Reference

RaspberryPi Class
IO.Objects.RemoteIO.Platforms Namespace

# RaspberryPiGPIO13 Field

GPIO channel number for the remote Raspberry Pi GPIO pin `GPIO13`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO13 = 13
```

### Field Value

Type: Int32

## Remarks

This GPIO channel is not usable when the Remote I/O server is running on an original Raspberry Pi 1 Model A or B with a 26-pin expansion header.

## See Also

### Reference

RaspberryPi Class
IO.Objects.RemoteIO.Platforms Namespace

# RaspberryPiGPIO14 Field

GPIO channel number for the remote Raspberry Pi GPIO pin `GPIO14`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO14 = 14
```

### Field Value

Type: Int32

## Remarks

This GPIO channel is normally unusable because `GPIO14` is mapped to UART signal `UART0_TXD`.

## See Also

### Reference

RaspberryPi Class
IO.Objects.RemoteIO.Platforms Namespace

# RaspberryPiGPIO15 Field

GPIO channel number for the remote Raspberry Pi GPIO pin `GPIO15`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO15 = 15
```

### Field Value

Type: Int32

## Remarks

This GPIO channel is normally unusable because `GPIO15` is mapped to UART signal `UART0_RXD`.

## See Also

### Reference

RaspberryPi Class
IO.Objects.RemoteIO.Platforms Namespace

# RaspberryPiGPIO16 Field

GPIO channel number for the remote Raspberry Pi GPIO pin `GPIO16`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO16 = 16
```

### Field Value

Type: Int32

## Remarks

This GPIO channel is not usable when the Remote I/O server is running on an original Raspberry Pi 1 Model A or B with a 26-pin expansion header.

## See Also

### Reference

RaspberryPi Class
IO.Objects.RemoteIO.Platforms Namespace

# RaspberryPiGPIO17 Field

GPIO channel number for the remote Raspberry Pi GPIO pin `GPIO17`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO17 = 17
```

### Field Value

Type: Int32

## See Also

### Reference

RaspberryPi Class
IO.Objects.RemoteIO.Platforms Namespace

# RaspberryPiGPIO18 Field

GPIO channel number for the remote Raspberry Pi GPIO pin `GPIO18`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO18 = 18
```

### Field Value

Type: Int32

## Remarks

This GPIO channel is normally unusable because `GPIO18` is mapped to PWM output `PWM0`.

## See Also

### Reference

RaspberryPi Class
IO.Objects.RemoteIO.Platforms Namespace

# RaspberryPiGPIO19 Field

GPIO channel number for the remote Raspberry Pi GPIO pin `GPIO19`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO19 = 19
```

### Field Value

Type: Int32

## Remarks

This GPIO channel is not usable when the Remote I/O server is running on an original Raspberry Pi 1 Model A or B with a 26-pin expansion header.

## See Also

### Reference

RaspberryPi Class
IO.Objects.RemoteIO.Platforms Namespace

# RaspberryPiGPIO2 Field

GPIO channel number for the remote Raspberry Pi GPIO pin `GPIO2`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO2 = 2
```

### Field Value

Type: Int32

## Remarks

This GPIO channel is normally unusable because `GPIO2` is mapped to $I^2C$ signal `SDA1`.

## See Also

### Reference

RaspberryPi Class
IO.Objects.RemoteIO.Platforms Namespace

# RaspberryPiGPIO20 Field

GPIO channel number for the remote Raspberry Pi GPIO pin `GPIO20`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO20 = 20
```

### Field Value

Type: Int32

## Remarks

This GPIO channel is not usable when the Remote I/O server is running on an original Raspberry Pi 1 Model A or B with a 26-pin expansion header.

## See Also

### Reference

RaspberryPi Class
IO.Objects.RemoteIO.Platforms Namespace

# RaspberryPiGPIO21 Field

GPIO channel number for the remote Raspberry Pi GPIO pin `GPIO21`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO21 = 21
```

### Field Value

Type: Int32

## Remarks

This GPIO channel is not usable when the Remote I/O server is running on an original Raspberry Pi 1 Model A or B with a 26-pin expansion header.

## See Also

### Reference

RaspberryPi Class
IO.Objects.RemoteIO.Platforms Namespace

# RaspberryPiGPIO22 Field

GPIO channel number for the remote Raspberry Pi GPIO pin `GPIO22`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO22 = 22
```

### Field Value

Type: Int32

## See Also

### Reference

RaspberryPi Class
IO.Objects.RemoteIO.Platforms Namespace

# RaspberryPiGPIO23 Field

GPIO channel number for the remote Raspberry Pi GPIO pin `GPIO23`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO23 = 23
```

### Field Value

Type: Int32

## See Also

### Reference

RaspberryPi Class
IO.Objects.RemoteIO.Platforms Namespace

# RaspberryPiGPIO24 Field

GPIO channel number for the remote Raspberry Pi GPIO pin `GPIO24`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO24 = 24
```

### Field Value

Type: Int32

## See Also

### Reference

RaspberryPi Class
IO.Objects.RemoteIO.Platforms Namespace

# RaspberryPiGPIO25 Field

GPIO channel number for the remote Raspberry Pi GPIO pin `GPIO25`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO25 = 25
```

### Field Value

Type: Int32

## See Also

### Reference

RaspberryPi Class
IO.Objects.RemoteIO.Platforms Namespace

# RaspberryPiGPIO26 Field

GPIO channel number for the remote Raspberry Pi GPIO pin `GPIO26`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO26 = 26
```

### Field Value

Type: Int32

## Remarks

This GPIO channel is not usable when the Remote I/O server is running on an original Raspberry Pi 1 Model A or B with a 26-pin expansion header.

## See Also

### Reference

RaspberryPi Class
IO.Objects.RemoteIO.Platforms Namespace

# RaspberryPiGPIO27 Field

GPIO channel number for the remote Raspberry Pi GPIO pin `GPIO27`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO27 = 27
```

### Field Value

Type: Int32

## See Also

### Reference

RaspberryPi Class
IO.Objects.RemoteIO.Platforms Namespace

# RaspberryPiGPIO3 Field

GPIO channel number for the remote Raspberry Pi GPIO pin `GPIO3`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO3 = 3
```

### Field Value

Type: Int32

## Remarks

This GPIO channel is normally unusable because `GPIO3` is mapped to to $I^2C$ signal `SCL1`.

## See Also

### Reference

RaspberryPi Class
IO.Objects.RemoteIO.Platforms Namespace

# RaspberryPiGPIO4 Field

GPIO channel number for the remote Raspberry Pi GPIO pin `GPIO4`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO4 = 4
```

### Field Value

Type: Int32

## See Also

### Reference

RaspberryPi Class
IO.Objects.RemoteIO.Platforms Namespace

# RaspberryPiGPIO5 Field

GPIO channel number for the remote Raspberry Pi GPIO pin `GPIO5`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO5 = 5
```

### Field Value

Type: Int32

## Remarks

This GPIO channel is not usable when the Remote I/O server is running on an original Raspberry Pi 1 Model A or B with a 26-pin expansion header.

## See Also

### Reference

RaspberryPi Class
IO.Objects.RemoteIO.Platforms Namespace

# RaspberryPiGPIO6 Field

GPIO channel number for the remote Raspberry Pi GPIO pin `GPIO6`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO6 = 6
```

### Field Value

Type: Int32

## Remarks

This GPIO channel is not usable when the Remote I/O server is running on an original Raspberry Pi 1 Model A or B with a 26-pin expansion header.

## See Also

### Reference

RaspberryPi Class
IO.Objects.RemoteIO.Platforms Namespace

# RaspberryPiGPIO7 Field

GPIO channel number for the remote Raspberry Pi GPIO pin `GPIO7`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO7 = 7
```

### Field Value

Type: Int32

## Remarks

This GPIO channel is normally unusable because `GPIO7` is mapped to SPI signal `SPI0_CE1`.

## See Also

### Reference

RaspberryPi Class
IO.Objects.RemoteIO.Platforms Namespace

# RaspberryPiGPIO8 Field

GPIO channel number for the remote Raspberry Pi GPIO pin `GPIO8`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO8 = 8
```

### Field Value

Type: Int32

## Remarks

This GPIO channel is normally unusable because `GPIO8` is mapped to SPI signal `SPI0_CE0`.

## See Also

### Reference

RaspberryPi Class
IO.Objects.RemoteIO.Platforms Namespace

# RaspberryPiGPIO9 Field

GPIO channel number for the remote Raspberry Pi GPIO pin `GPIO9`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int GPIO9 = 9
```

### Field Value

Type: Int32

## Remarks

This GPIO channel is normally unusable because `GPIO9` is mapped to SPI signal `SPI0_MISO`.

## See Also

### Reference

RaspberryPi Class
IO.Objects.RemoteIO.Platforms Namespace

# RaspberryPiI2C1 Field

I²C bus channel number for the remote Raspberry Pi I²C bus controller `I2C1`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int I2C1 = 0
```

### Field Value

Type: Int32

## Remarks

The following device tree overlay command must be added to `config.txt` on the Raspberry Pi Remote I/O server to enable I²C bus controller `I2C1`:

Copy

```
dtparam=i2c=on
```

## See Also

### Reference

RaspberryPi Class
IO.Objects.RemoteIO.Platforms Namespace

# RaspberryPiPWM0 Field

PWM output channel number for the remote Raspberry Pi PWM output `PWM0`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int PWM0 = 0
```

### Field Value

Type: Int32

## Remarks

The following device tree overlay command must be added to `config.txt` on the Raspberry Pi Remote I/O server to enable PWM output `PWM0`:

Copy

```
dtoverlay=pwm
```

## See Also

### Reference

RaspberryPi Class
IO.Objects.RemoteIO.Platforms Namespace

# RaspberryPiPWM1 Field

PWM output channel number for the remote Raspberry Pi PWM output `PWM1`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int PWM1 = 1
```

### Field Value

Type: Int32

## Remarks

`PWM1` is not normally enabled. You can enable it with one of the device tree overlay command variationss described in https://github.com/raspberrypi/firmware/blob/master/boot/overlays/README.

## See Also

### Reference

RaspberryPi Class
IO.Objects.RemoteIO.Platforms Namespace

# RaspberryPiSPI0 Field

SPI slave device channel number for the remote Raspberry Pi SPI slave device `/dev/spidev0.0`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int SPI0 = 0
```

### Field Value

Type: Int32

## Remarks

`/dev/spidev0.0` uses `GPIO8` for slave select. The following device tree overlay command must be added to `config.txt` on the Raspberry Pi Remote I/O server to enable SPI bus controller `SPI0`:

Copy

```
dtparam=spi=on
```

## See Also

### Reference

RaspberryPi Class
IO.Objects.RemoteIO.Platforms Namespace

# RaspberryPiSPI1 Field

SPI slave device channel number for the remote Raspberry Pi SPI slave device `/dev/spidev0.1`.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int SPI1 = 1
```

### Field Value

Type: Int32

## Remarks

`/dev/spidev0.1` uses `GPIO7` for slave select. The following device tree overlay command must be added to `config.txt` on the Raspberry Pi Remote I/O server to enable SPI bus controller `SPI0`:

Copy

```
dtparam=spi=on
```

## See Also

### Reference

RaspberryPi Class
IO.Objects.RemoteIO.Platforms Namespace

# RaspberryPiUSERLED Field

GPIO channel number for the remote Raspberry Pi user LED.

**Namespace:** IO.Objects.RemoteIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public const int USERLED = 0
```

### Field Value

Type: Int32

## Remarks

This GPIO channel cannot be configured as an input.

## See Also

### Reference

RaspberryPi Class
IO.Objects.RemoteIO.Platforms Namespace

# IO.Objects.Servo.PWM Namespace

PWM Controlled Servo Services

## Classes

| | Class | Description |
|---|---|---|
| | Output | Encapsulates servo outputs using PWM outputs. |

# Output Class

Encapsulates servo outputs using PWM outputs.

## Inheritance Hierarchy

SystemObject IO.Objects.Servo.PWMOutput

**Namespace:** IO.Objects.Servo.PWM
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public class Output : Output
```

The Output type exposes the following members.

## Constructors

| | Name | Description |
|---|---|---|
| | Output | Constructor for a single servo output. |

Top

## Properties

| | Name | Description |
|---|---|---|
| | position | Write-only property for setting the servo position. Allowed values are -1.0 to +1.0. |

Top

## See Also

### Reference

IO.Objects.Servo.PWM Namespace

# Output Constructor

Constructor for a single servo output.

**Namespace:** IO.Objects.Servo.PWM
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public Output(
        Output pwm,
        int freq = 50,
        double position = 0
)
```

### Parameters

*pwm*
Type: IO.Interfaces.PWMOutput
PWM output instance.

*freq* **(Optional)**
Type: SystemInt32
PWM pulse frequency.

*position* **(Optional)**
Type: SystemDouble
Initial servo position.

## See Also

### Reference

Output Class

# IO.Objects.Servo.PWM Namespace

# Output Properties

The Output type exposes the following members.

## Properties

| | Name | Description |
|---|---|---|
| | position | Write-only property for setting the servo position. Allowed values are -1.0 to +1.0. |

Top

## See Also

**Reference**

Output Class

IO.Objects.Servo.PWM Namespace

# Outputposition Property

Write-only property for setting the servo position. Allowed values are -1.0 to +1.0.

**Namespace:** IO.Objects.Servo.PWM
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public double position { set; }
```

### Property Value

Type: Double

### Implements

Outputposition

## See Also

### Reference

Output Class
IO.Objects.Servo.PWM Namespace

# IO.Objects.SimpleIO.ADC Namespace

ADC (Analog to Digital Converter) Input Services

## Classes

| | Class | Description |
|---|---|---|
| | Sample | Encapsulates Linux Industrial I/O Subsystem ADC inputs usingi `libsimpleio`. |

# Sample Class

Encapsulates Linux Industrial I/O Subsystem ADC inputs usingi `libsimpleio`.

## Inheritance Hierarchy

SystemObject IO.Objects.SimpleIO.ADCSample

**Namespace:** IO.Objects.SimpleIO.ADC
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public class Sample : Sample
```

The Sample type exposes the following members.

## Constructors

| | Name | Description |
|---|---|---|
| | Sample | Constructor for a single ADC input. |

Top

## Properties

| | Name | Description |
|---|---|---|
| | fd | Read-only property returning the Linux |

| | | |
|---|---|---|
| | | file descriptor for the ADC input. |
| | resolution | Read-only property returning the number of bits of resolution. |
| | sample | Read-only property returning an integer analog sample from an ADC input. |

Top

# Methods

| | Name | Description |
|---|---|---|
| | name | Retrieve the subsystem name string for a Linux Industrial I/O Subsystem ADC device. |

Top

# See Also

**Reference**

IO.Objects.SimpleIO.ADC Namespace

# Sample Constructor

Constructor for a single ADC input.

**Namespace:** IO.Objects.SimpleIO.ADC
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public Sample(
        Designator desg,
        int resolution
)
```

### Parameters

*desg*
Type: IO.Objects.SimpleIO.DeviceDesignator
ADC input designator.

*resolution*
Type: SystemInt32
Bits of resolution.

## See Also

### Reference

Sample Class
IO.Objects.SimpleIO.ADC Namespace

# Sample Properties

The Sample type exposes the following members.

## Properties

| | Name | Description |
|---|---|---|
| | fd | Read-only property returning the Linux file descriptor for the ADC input. |
| | resolution | Read-only property returning the number of bits of resolution. |
| | sample | Read-only property returning an integer analog sample from an ADC input. |

Top

## See Also

**Reference**

Sample Class

IO.Objects.SimpleIO.ADC Namespace

# Samplefd Property

Read-only property returning the Linux file descriptor for the ADC input.

**Namespace:** IO.Objects.SimpleIO.ADC
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public int fd { get; }
```

### Property Value

Type: Int32

## See Also

### Reference

Sample Class
IO.Objects.SimpleIO.ADC Namespace

# Sampleresolution Property

Read-only property returning the number of bits of resolution.

**Namespace:** IO.Objects.SimpleIO.ADC
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public int resolution { get; }
```

### Property Value

Type: Int32

### Implements

Sampleresolution

## See Also

### Reference

Sample Class
IO.Objects.SimpleIO.ADC Namespace

# Samplesample Property

Read-only property returning an integer analog sample from an ADC input.

**Namespace:** IO.Objects.SimpleIO.ADC
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public int sample { get; }
```

### Property Value

Type: Int32

### Implements

Samplesample

## See Also

### Reference

Sample Class
IO.Objects.SimpleIO.ADC Namespace

# Sample Methods

The Sample type exposes the following members.

## Methods

| | Name | Description |
|---|---|---|
| | name | Retrieve the subsystem name string for a Linux Industrial I/O Subsystem ADC device. |

Top

## See Also

**Reference**

Sample Class

IO.Objects.SimpleIO.ADC Namespace

# Samplename Method

Retrieve the subsystem name string for a Linux Industrial I/O Subsystem ADC device.

**Namespace:** IO.Objects.SimpleIO.ADC
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static string name(
        int chip
)
```

### Parameters

*chip*
Type: SystemInt32
ADC chip number.

### Return Value

Type: String
Subsystem name.

## See Also

### Reference

Sample Class
IO.Objects.SimpleIO.ADC Namespace

# IO.Objects.SimpleIO.DAC Namespace

DAC (Digital to Analog Converter) Output Services

## Classes

| | Class | Description |
|---|---|---|
| | Sample | Encapsulates Linux Industrial I/O Subsystem DAC outputs using `libsimpleio`. |

# Sample Class

Encapsulates Linux Industrial I/O Subsystem DAC outputs using `libsimpleio`.

## Inheritance Hierarchy

SystemObject IO.Objects.SimpleIO.DACSample

**Namespace:** IO.Objects.SimpleIO.DAC
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public class Sample : Sample
```

The Sample type exposes the following members.

## Constructors

| | Name | Description |
|---|---|---|
| | Sample | Constructor for a single DAC output. |

Top

## Properties

| | Name | Description |
|---|---|---|
| | fd | Read-only property returning the Linux |

| | | |
|---|---|---|
| | | file descriptor for the DAC output. |
| | resolution | Read-only property returning the number of bits of resolution. |
| | sample | Write-only property for writing an integer analog sample to a DAC output. |

Top

# Methods

| | Name | Description |
|---|---|---|
| | name | Retrieve the subsystem name string for a Linux Industrial I/O Subsystem DAC device. |

Top

# See Also

**Reference**

IO.Objects.SimpleIO.DAC Namespace

# Sample Constructor

Constructor for a single DAC output.

**Namespace:** IO.Objects.SimpleIO.DAC
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public Sample(
        Designator desg,
        int resolution,
        int sample = 0
)
```

### Parameters

*desg*
Type: IO.Objects.SimpleIO.DeviceDesignator
DAC output designator.

*resolution*
Type: SystemInt32
Bits of resolution.

*sample* **(Optional)**
Type: SystemInt32
Initial DAC output sample.

## See Also

### Reference

Sample Class

# IO.Objects.SimpleIO.DAC Namespace

# Sample Properties

The Sample type exposes the following members.

## Properties

| | Name | Description |
|---|---|---|
| | fd | Read-only property returning the Linux file descriptor for the DAC output. |
| | resolution | Read-only property returning the number of bits of resolution. |
| | sample | Write-only property for writing an integer analog sample to a DAC output. |

Top

## See Also

**Reference**

Sample Class

IO.Objects.SimpleIO.DAC Namespace

# Samplefd Property

Read-only property returning the Linux file descriptor for the DAC output.

**Namespace:** IO.Objects.SimpleIO.DAC
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public int fd { get; }
```

### Property Value

Type: Int32

## See Also

### Reference

Sample Class
IO.Objects.SimpleIO.DAC Namespace

# Sampleresolution Property

Read-only property returning the number of bits of resolution.

**Namespace:** IO.Objects.SimpleIO.DAC
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public int resolution { get; }
```

### Property Value

Type: Int32

### Implements

Sampleresolution

## See Also

### Reference

Sample Class
IO.Objects.SimpleIO.DAC Namespace

# Samplesample Property

Write-only property for writing an integer analog sample to a DAC output.

**Namespace:** IO.Objects.SimpleIO.DAC
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public int sample { set; }
```

### Property Value

Type: Int32

### Implements

Samplesample

## See Also

### Reference

Sample Class
IO.Objects.SimpleIO.DAC Namespace

# Sample Methods

The Sample type exposes the following members.

## Methods

| | Name | Description |
|---|---|---|
| | name | Retrieve the subsystem name string for a Linux Industrial I/O Subsystem DAC device. |

Top

## See Also

**Reference**

Sample Class

IO.Objects.SimpleIO.DAC Namespace

# Samplename Method

Retrieve the subsystem name string for a Linux Industrial I/O Subsystem DAC device.

**Namespace:** IO.Objects.SimpleIO.DAC
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static string name(
        int chip
)
```

### Parameters

*chip*
Type: SystemInt32
DAC chip number.

### Return Value

Type: String
Subsystem name.

## See Also

### Reference

Sample Class
IO.Objects.SimpleIO.DAC Namespace

# IO.Objects.SimpleIO.Device Namespace

Common device declarations

## Structures

| | Structure | Description |
|---|---|---|
| | Designator | Linux kernel I/O device designator. |

# Designator Structure

Linux kernel I/O device designator.

**Namespace:** IO.Objects.SimpleIO.Device
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public struct Designator
```

The Designator type exposes the following members.

## Constructors

| | Name | Description |
|---|---|---|
| | Designator | device pin designator constructor. |

Top

## Properties

| | Name | Description |
|---|---|---|
| | available | Returns `true` if this device designator is not equal to `Unavailable`. |

Top

## Fields

| | Name | Description |
|---|---|---|
| | chan | Linux kernel I/O device channel number. |
| | chip | Linux kernel I/O device chip number. |
| S | Unavailable | Linux kernel I/O device designator for an unavailable device. |

Top

## Remarks

Many Linux kernel I/O devices, including ADC inputs, DAC outputs, GPIO pins, I2C buses, PWM outputs, and SPI devices, are selected by a tuple of integers: chip and channel.

## See Also

### Reference

IO.Objects.SimpleIO.Device Namespace

# Designator Constructor

device pin designator constructor.

**Namespace:** IO.Objects.SimpleIO.Device
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public Designator(
        uint chip,
        uint chan
)
```

### Parameters

*chip*
Type: SystemUInt32
Linux kernel I/O device chip number.

*chan*
Type: SystemUInt32
Linux kernel I/O device channel number.

## See Also

### Reference

Designator Structure
IO.Objects.SimpleIO.Device Namespace

# Designator Properties

The Designator type exposes the following members.

## Properties

| | Name | Description |
|---|---|---|
| | available | Returns `true` if this device designator is not equal to `Unavailable`. |

Top

## See Also

**Reference**

Designator Structure
IO.Objects.SimpleIO.Device Namespace

# Designatoravailable Property

Returns `true` if this device designator is not equal to `Unavailable`.

**Namespace:** IO.Objects.SimpleIO.Device
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public bool available { get; }
```

### Property Value

Type: Boolean

## See Also

### Reference

Designator Structure
IO.Objects.SimpleIO.Device Namespace

# Designator Fields

The Designator type exposes the following members.

## Fields

| | Name | Description |
|---|---|---|
| | chan | Linux kernel I/O device channel number. |
| | chip | Linux kernel I/O device chip number. |
| S | Unavailable | Linux kernel I/O device designator for an unavailable device. |

Top

## See Also

**Reference**

Designator Structure

IO.Objects.SimpleIO.Device Namespace

# Designatorchan Field

Linux kernel I/O device channel number.

**Namespace:** IO.Objects.SimpleIO.Device
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public uint chan
```

### Field Value

Type: UInt32

## See Also

### Reference

Designator Structure
IO.Objects.SimpleIO.Device Namespace

# Designatorchip Field

Linux kernel I/O device chip number.

**Namespace:** IO.Objects.SimpleIO.Device
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public uint chip
```

### Field Value

Type: UInt32

## See Also

### Reference

Designator Structure
IO.Objects.SimpleIO.Device Namespace

# DesignatorUnavailable Field

Linux kernel I/O device designator for an unavailable device.

**Namespace:** IO.Objects.SimpleIO.Device
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator Unavailable
```

### Field Value

Type: Designator

## See Also

### Reference

Designator Structure
IO.Objects.SimpleIO.Device Namespace

# IO.Objects.SimpleIO.GPIO Namespace

GPIO (General Purpose Input/Output) Pin Services

## Classes

| | Class | Description |
|---|---|---|
| | Pin | Encapsulates Linux GPIO pins using `libsimpleio`. |

## Enumerations

| | Enumeration | Description |
|---|---|---|
| | Driver | GPIO output driver settings. |
| | Edge | GPIO input interrupt edge settings. |
| | Polarity | GPIO polarity settings |

# Driver Enumeration

GPIO output driver settings.

**Namespace:** IO.Objects.SimpleIO.GPIO
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public enum Driver
```

## Members

| Member name | Value | Description |
|---|---|---|
| PushPull | 0 | Push Pull (current source/sink) output driver. |
| OpenDrain | 1 | Open Drain (current sink) output driver. |
| OpenSource | 2 | Open Source (current source) output driver. |

## See Also

**Reference**
IO.Objects.SimpleIO.GPIO Namespace

# Edge Enumeration

GPIO input interrupt edge settings.

**Namespace:** IO.Objects.SimpleIO.GPIO
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public enum Edge
```

## Members

| Member name | Value | Description |
|---|---|---|
| None | 0 | Configure GPIO input pin with interrupt disabled. |
| Rising | 1 | Configure GPIO input pin to interrupt on rising edge. |
| Falling | 2 | Configure GPIO pin to interrupt on falling edge. |
| Both | 3 | Configure GPIO pin to interrupt on both edges. |

## See Also

**Reference**

IO.Objects.SimpleIO.GPIO Namespace

# Pin Class

Encapsulates Linux GPIO pins using `libsimpleio`.

## Inheritance Hierarchy

SystemObject IO.Objects.SimpleIO.GPIOPin

**Namespace:** IO.Objects.SimpleIO.GPIO
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public class Pin : Pin
```

The Pin type exposes the following members.

## Constructors

| | Name | Description |
|---|---|---|
| | Pin | Constructor for a single GPIO pin. |

Top

## Properties

| | Name | Description |
|---|---|---|
| | fd | Read-only property returning the Linux file descriptor for the GPIO pin. |

| | | |
|---|---|---|
| | state | Read/Write GPIO state property. |

Top

## See Also

### Reference

IO.Objects.SimpleIO.GPIO Namespace

# Pin Constructor

Constructor for a single GPIO pin.

**Namespace:** IO.Objects.SimpleIO.GPIO
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public Pin(
        Designator desg,
        Direction dir,
        bool state = false,
        Driver driver = Driver.PushPull,
        Edge edge = Edge.None,
        Polarity polarity = Polarity.ActiveHigh
)
```

### Parameters

*desg*
Type: IO.Objects.SimpleIO.DeviceDesignator
GPIO pin designator.

*dir*
Type: IO.Interfaces.GPIODirection
Data direction.

*state* **(Optional)**
Type: SystemBoolean
Initial GPIO output state.

*driver* **(Optional)**

Type: IO.Objects.SimpleIO.GPIODriver
Output driver setting.

*edge* **(Optional)**
Type: IO.Objects.SimpleIO.GPIOEdge
Interrupt edge setting.

*polarity* **(Optional)**
Type: IO.Objects.SimpleIO.GPIOPolarity
Polarity setting.

## See Also

### Reference

Pin Class
IO.Objects.SimpleIO.GPIO Namespace

# Pin Properties

The Pin type exposes the following members.

## Properties

| | Name | Description |
|---|---|---|
| | fd | Read-only property returning the Linux file descriptor for the GPIO pin. |
| | state | Read/Write GPIO state property. |

Top

## See Also

**Reference**

Pin Class

IO.Objects.SimpleIO.GPIO Namespace

# Pinfd Property

Read-only property returning the Linux file descriptor for the GPIO pin.

**Namespace:** IO.Objects.SimpleIO.GPIO
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public int fd { get; }
```

### Property Value

Type: Int32

## See Also

### Reference

Pin Class
IO.Objects.SimpleIO.GPIO Namespace

# Pinstate Property

Read/Write GPIO state property.

**Namespace:** IO.Objects.SimpleIO.GPIO
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public bool state { get; set; }
```

### Property Value

Type: Boolean

### Implements

Pinstate

## See Also

### Reference

Pin Class
IO.Objects.SimpleIO.GPIO Namespace

# Polarity Enumeration

GPIO polarity settings

**Namespace:** IO.Objects.SimpleIO.GPIO
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public enum Polarity
```

## Members

| Member name | Value | Description |
|---|---|---|
| ActiveLow | 0 | Configure GPIO pin as active low (inverted logic). |
| ActiveHigh | 1 | Configure GPIO pin as active high (normal logic). |

## See Also

### Reference

IO.Objects.SimpleIO.GPIO Namespace

# IO.Objects.SimpleIO.HID Namespace

Raw HID (Human Interface Device) Services

## Classes

| | Class | Description |
|---|---|---|
| | Messenger | Encapsulates Linux raw HID devices using `libsimpleio`. |

# Messenger Class

Encapsulates Linux raw HID devices using `libsimpleio`.

## Inheritance Hierarchy

SystemObject IO.Objects.SimpleIO.HIDMessenger

**Namespace:** IO.Objects.SimpleIO.HID
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public class Messenger : Messenger
```

The Messenger type exposes the following members.

## Constructors

| | Name | Description |
|---|---|---|
| | Messenger(String, Int32) | Constructor for a single raw HID device. |
| | Messenger(Int32, Int32, String, Int32) | Constructor for a single raw HID device. |

Top

## Properties

| | Name | Description |
|---|---|---|
| | bustype | Read-only property returning the bus type identifier for a raw HID device. |
| | fd | Read-only property returning the Linux file descriptor for a raw HID device. |
| | name | Read-only property returning the device information string for a raw HID device. |
| | product | Read-only property returning the product identifier for a raw HID device. |
| | vendor | Read-only property returning the vendor identifier for a raw HID device. |

Top

# Methods

| | Name | Description |
|---|---|---|
| | Receive | Receive a 64-byte response message from a raw HID device. |
| | Send | Send a 64-byte command message to a raw HID device. |
| | Transaction | Send a 64-byte command message and receive a 64-byte response message. |

Top

# See Also

## Reference

IO.Objects.SimpleIO.HID Namespace

# Messenger Constructor

## Overload List

| | Name | Description |
|---|---|---|
| | Messenger(String, Int32) | Constructor for a single raw HID device. |
| | Messenger(Int32, Int32, String, Int32) | Constructor for a single raw HID device. |

Top

## See Also

**Reference**

Messenger Class

IO.Objects.SimpleIO.HID Namespace

# Messenger Constructor (String, Int32)

Constructor for a single raw HID device.

**Namespace:** IO.Objects.SimpleIO.HID
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

**C#** **VB** **F#** Copy

```
public Messenger(
        string devname,
        int timeoutms = 1000
)
```

### Parameters

*devname*
Type: SystemString
Device node name.

*timeoutms* **(Optional)**
Type: SystemInt32
Time in milliseconds to wait for read and write operations to complete. Zero means wait forever.

## See Also

### Reference

Messenger Class
Messenger Overload

# IO.Objects.SimpleIO.HID Namespace

# Messenger Constructor (Int32, Int32, String, Int32)

Constructor for a single raw HID device.

**Namespace:** IO.Objects.SimpleIO.HID
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public Messenger(
        int VID = 5840,
        int PID = 2810,
        string serial = null,
        int timeoutms = 1000
)
```

### Parameters

*VID* **(Optional)**
Type: SystemInt32
Vendor ID.

*PID* **(Optional)**
Type: SystemInt32
Product ID.

*serial* **(Optional)**
Type: SystemString
Serial Number.

*timeoutms* **(Optional)**

Type: SystemInt32
Time in milliseconds to wait for read and write operations to complete. Zero means wait forever.

## See Also

### Reference

Messenger Class
Messenger Overload
IO.Objects.SimpleIO.HID Namespace

# Messenger Properties

The Messenger type exposes the following members.

## Properties

| | Name | Description |
|---|---|---|
| | bustype | Read-only property returning the bus type identifier for a raw HID device. |
| | fd | Read-only property returning the Linux file descriptor for a raw HID device. |
| | name | Read-only property returning the device information string for a raw HID device. |
| | product | Read-only property returning the product identifier for a raw HID device. |
| | vendor | Read-only property returning the vendor identifier for a raw HID device. |

Top

## See Also

### Reference

Messenger Class
IO.Objects.SimpleIO.HID Namespace

# Messengerbustype Property

Read-only property returning the bus type identifier for a raw HID device.

**Namespace:** IO.Objects.SimpleIO.HID
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public int bustype { get; }
```

### Property Value

Type: Int32

## See Also

### Reference

Messenger Class
IO.Objects.SimpleIO.HID Namespace

# Messengerfd Property

Read-only property returning the Linux file descriptor for a raw HID device.

**Namespace:** IO.Objects.SimpleIO.HID
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public int fd { get; }
```

### Property Value

Type: Int32

## See Also

### Reference

Messenger Class
IO.Objects.SimpleIO.HID Namespace

# Messengername Property

Read-only property returning the device information string for a raw HID device.

**Namespace:** IO.Objects.SimpleIO.HID
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public string name { get; }
```

### Property Value

Type: String

## See Also

### Reference

Messenger Class
IO.Objects.SimpleIO.HID Namespace

# Messengerproduct Property

Read-only property returning the product identifier for a raw HID device.

**Namespace:** IO.Objects.SimpleIO.HID
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public int product { get; }
```

### Property Value

Type: Int32

## See Also

### Reference

Messenger Class
IO.Objects.SimpleIO.HID Namespace

# Messengervendor Property

Read-only property returning the vendor identifier for a raw HID device.

**Namespace:** IO.Objects.SimpleIO.HID
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public int vendor { get; }
```

### Property Value

Type: Int32

## See Also

### Reference

Messenger Class
IO.Objects.SimpleIO.HID Namespace

# Messenger Methods

The Messenger type exposes the following members.

## Methods

| | Name | Description |
|---|---|---|
| | Receive | Receive a 64-byte response message from a raw HID device. |
| | Send | Send a 64-byte command message to a raw HID device. |
| | Transaction | Send a 64-byte command message and receive a 64-byte response message. |

Top

## See Also

**Reference**

Messenger Class

IO.Objects.SimpleIO.HID Namespace

# MessengerReceive Method

Receive a 64-byte response message from a raw HID device.

**Namespace:** IO.Objects.SimpleIO.HID
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public void Receive(
        Message resp
)
```

### Parameters

*resp*
Type: IO.Interfaces.Message64Message
64-byte response message.

### Implements

MessengerReceive(Message)

## See Also

### Reference

Messenger Class
IO.Objects.SimpleIO.HID Namespace

# MessengerSend Method

Send a 64-byte command message to a raw HID device.

**Namespace:** IO.Objects.SimpleIO.HID
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public void Send(
        Message cmd
)
```

### Parameters

*cmd*
Type: IO.Interfaces.Message64Message
64-byte command message.

### Implements

MessengerSend(Message)

## See Also

### Reference

Messenger Class
IO.Objects.SimpleIO.HID Namespace

# MessengerTransaction Method

Send a 64-byte command message and receive a 64-byte response message.

**Namespace:** IO.Objects.SimpleIO.HID
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public void Transaction(
        Message cmd,
        Message resp
)
```

### Parameters

*cmd*
Type: IO.Interfaces.Message64Message
64-byte command message.

*resp*
Type: IO.Interfaces.Message64Message
64-byte response message.

### Implements

MessengerTransaction(Message, Message)

## See Also

### Reference

Messenger Class

# IO.Objects.SimpleIO.HID Namespace

# IO.Objects.SimpleIO.I2C Namespace

$I^2C$ (Inter-Integrated Circuit) Bus Controller Services

## Classes

| | Class | Description |
|---|---|---|
| | Bus | Encapsulates Linux $I^2C$ bus controllers using `libsimpleio`. |

# Bus Class

Encapsulates Linux I²C bus controllers using `libsimpleio`.

## Inheritance Hierarchy

SystemObject IO.Objects.SimpleIO.I2CBus

**Namespace:** IO.Objects.SimpleIO.I2C
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public class Bus : Bus
```

The Bus type exposes the following members.

## Constructors

| | Name | Description |
|---|---|---|
| | Bus(Designator) | Constructor for a single I²C bus controller. |
| | Bus(String) | Constructor for a single I²C bus controller. |

Top

## Properties

| | Name | Description |
|---|---|---|
| | fd | Read-only property returning the Linux file descriptor for the $I^2C$ bus controller. |

Top

## Methods

| | Name | Description |
|---|---|---|
| | Read | Read bytes from an $I^2C$ device. |
| | Transaction | Write and receive bytes to/from an $I^2C$ device. |
| | Write | Write bytes to an $I^2C$ device. |

Top

## See Also

**Reference**

IO.Objects.SimpleIO.I2C Namespace

# Bus Constructor

## Overload List

| | Name | Description |
|---|---|---|
| | Bus(Designator) | Constructor for a single I$^2$C bus controller. |
| | Bus(String) | Constructor for a single I$^2$C bus controller. |

Top

## See Also

**Reference**

Bus Class

IO.Objects.SimpleIO.I2C Namespace

# Bus Constructor (Designator)

Constructor for a single $I^2C$ bus controller.

**Namespace:** IO.Objects.SimpleIO.I2C
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public Bus(
        Designator desg
)
```

### Parameters

*desg*
Type: IO.Objects.SimpleIO.DeviceDesignator
$I^2$ bus designator.

## See Also

### Reference

Bus Class
Bus Overload
IO.Objects.SimpleIO.I2C Namespace

# Bus Constructor (String)

Constructor for a single I²C bus controller.

**Namespace:** IO.Objects.SimpleIO.I2C
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public Bus(
        string devname
)
```

### Parameters

*devname*
Type: SystemString
Device node name.

## See Also

### Reference

Bus Class
Bus Overload
IO.Objects.SimpleIO.I2C Namespace

# Bus Properties

The Bus type exposes the following members.

## Properties

| | Name | Description |
|---|---|---|
| | fd | Read-only property returning the Linux file descriptor for the $I^2C$ bus controller. |

Top

## See Also

**Reference**

Bus Class

IO.Objects.SimpleIO.I2C Namespace

# Busfd Property

Read-only property returning the Linux file descriptor for the $I^2C$ bus controller.

**Namespace:** IO.Objects.SimpleIO.I2C
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public int fd { get; }
```

### Property Value

Type: Int32

## See Also

### Reference

Bus Class
IO.Objects.SimpleIO.I2C Namespace

# Bus Methods

The Bus type exposes the following members.

## Methods

| | Name | Description |
|---|---|---|
| | Read | Read bytes from an I$^2$C device. |
| | Transaction | Write and receive bytes to/from an I$^2$C device. |
| | Write | Write bytes to an I$^2$C device. |

Top

## See Also

**Reference**

Bus Class

IO.Objects.SimpleIO.I2C Namespace

# BusRead Method

Read bytes from an I²C device.

**Namespace:** IO.Objects.SimpleIO.I2C
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public void Read(
        int slaveaddr,
        byte[] resp,
        int resplen
)
```

### Parameters

*slaveaddr*
Type: SystemInt32
Slave device address.

*resp*
Type: SystemByte
Response buffer.

*resplen*
Type: SystemInt32
Number of bytes to read.

### Implements

BusRead(Int32, Byte, Int32)

## See Also

**Reference**

Bus Class

IO.Objects.SimpleIO.I2C Namespace

# BusTransaction Method

Write and receive bytes to/from an I$^2$C device.

**Namespace:** IO.Objects.SimpleIO.I2C
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public void Transaction(
        int slaveaddr,
        byte[] cmd,
        int cmdlen,
        byte[] resp,
        int resplen,
        int delayus = 0
)
```

### Parameters

*slaveaddr*
Type: SystemInt32
Device slave address.

*cmd*
Type: SystemByte
Command buffer.

*cmdlen*
Type: SystemInt32
Number of bytes to write.

*resp*

Type: SystemByte
Response buffer.

*resplen*

Type: SystemInt32
Number of bytes to read.

*delayus* **(Optional)**

Type: SystemInt32
Delay in microseconds between the $I^2C$ write and read cycles.
Allowed values are 0 to 65535 microseconds.

### Implements

BusTransaction(Int32, Byte, Int32, Byte, Int32, Int32)

## See Also

### Reference

Bus Class
IO.Objects.SimpleIO.I2C Namespace

# BusWrite Method

Write bytes to an I²C device.

**Namespace:** IO.Objects.SimpleIO.I2C
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public void Write(
        int slaveaddr,
        byte[] cmd,
        int cmdlen
)
```

### Parameters

*slaveaddr*
Type: SystemInt32
Slave device address.

*cmd*
Type: SystemByte
Command buffer.

*cmdlen*
Type: SystemInt32
Number of bytes to write.

### Implements

BusWrite(Int32, Byte, Int32)

## See Also

**Reference**

Bus Class

IO.Objects.SimpleIO.I2C Namespace

# IO.Objects.SimpleIO.mikroBUS Namespace

Mikroelektronika mikroBUS (https://www.mikroe.com/mikrobus) Shield and Socket Services

## Classes

| | Class | Description |
|---|---|---|
| | Shield | Encapsulates mikroBUS shields (add-on boards providing mikroBUS sockets). |
| | Socket | Encapsulates mikroBUS sockets. |

## Enumerations

| | Enumeration | Description |
|---|---|---|
| | ShieldKinds | Supported mikroBUS shields. |

# Shield Class

Encapsulates mikroBUS shields (add-on boards providing mikroBUS sockets).

## Inheritance Hierarchy

SystemObject IO.Objects.SimpleIO.mikroBUSShield

**Namespace:** IO.Objects.SimpleIO.mikroBUS
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static class Shield
```

The Shield type exposes the following members.

## Properties

| | Name | Description |
|---|---|---|
| | kind | Returns the kind of mikroBUS shield that is installed on the target board, as obtained from the `SHIELDNAME` environment variable or guessed from the `BOARDNAME` environment variable. The guessed value for BeagleBone family target boards (`BOARDNAME` == "BeagleBone*") is `BeagleBoneClick4`. |

| | |
|---|---|
| | The guessed value for Raspberry Pi family target boards (`BOARDNAME` == "RaspberryPi*") is `PiClick3`. |

Top

## Fields

| | Name | Description |
|---|---|---|
| | I2CBus | Shared $I^2C$ bus that is common to all sockets on this shield. |

Top

## See Also

**Reference**

IO.Objects.SimpleIO.mikroBUS Namespace

# Shield Properties

The Shield type exposes the following members.

## Properties

| | Name | Description |
|---|---|---|
| | kind | Returns the kind of mikroBUS shield that is installed on the target board, as obtained from the `SHIELDNAME` environment variable or guessed from the `BOARDNAME` environment variable. The guessed value for BeagleBone family target boards (`BOARDNAME` == "BeagleBone*") is `BeagleBoneClick4`. The guessed value for Raspberry Pi family target boards (`BOARDNAME` == "RaspberryPi*") is `PiClick3`. |

Top

## See Also

**Reference**

Shield Class

IO.Objects.SimpleIO.mikroBUS Namespace

# Shieldkind Property

Returns the kind of mikroBUS shield that is installed on the target board, as obtained from the `SHIELDNAME` environment variable or guessed from the `BOARDNAME` environment variable. The guessed value for BeagleBone family target boards (`BOARDNAME` == "BeagleBone*") is `BeagleBoneClick4`. The guessed value for Raspberry Pi family target boards (`BOARDNAME` == "RaspberryPi*") is `PiClick3`.

**Namespace:** IO.Objects.SimpleIO.mikroBUS
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

**C#** **VB** **F#** Copy

```
public static ShieldKinds kind { get; }
```

### Property Value

Type: ShieldKinds

## See Also

### Reference

Shield Class
IO.Objects.SimpleIO.mikroBUS Namespace

# Shield Fields

The Shield type exposes the following members.

## Fields

| | Name | Description |
|---|---|---|
| | I2CBus | Shared $I^2C$ bus that is common to all sockets on this shield. |

Top

## See Also

**Reference**

Shield Class

IO.Objects.SimpleIO.mikroBUS Namespace

# ShieldI2CBus Field

Shared $I^2C$ bus that is common to all sockets on this shield.

**Namespace:** IO.Objects.SimpleIO.mikroBUS
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static Bus I2CBus
```

### Field Value

Type: Bus

## See Also

### Reference

Shield Class
IO.Objects.SimpleIO.mikroBUS Namespace

# ShieldKinds Enumeration

Supported mikroBUS shields.

**Namespace:** IO.Objects.SimpleIO.mikroBUS
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public enum Kinds
```

## Members

| Member name | Value | Description |
|---|---|---|
| BeagleBoneClick2 | 0 | Mikroelektronika BeagleBone Click Shield MIKROE-1596, with two mikroBUS sockets. (Obsolete, but still useful.) |
| BeagleBoneClick4 | 1 | Mikroelektronika mikroBUS Cape MIKROE-1857 with four mikroBUS sockets. |

| | | | |
|---|---|---|---|
| | PiClick1 | 2 | Mikroelektronika Pi Click Shield MIKROE-1512/1513 for 26-pin expansion header, with one mikroBUS socket (Obsolete.) |
| | PiClick2 | 3 | Mikroelektronika Pi 2 Click Shield MIKROE-1879 for 40-pin expansion header, with two mikroBUS sockets. |
| | PiClick3 | 4 | Mikroelektronika Pi 3 Click Shield MIKROE-2756 for 40-pin expansion header, with selectable on-board A/D converter and two mikroBUS sockets. |
| | PocketBeagle | 5 | PocketBeagle with female headers on top, with two |

| | | | |
|---|---|---|---|
| | | | mikroBUS sockets. |
| | Unknown | 2147483647 | No known mikroBUS shield installed. |

## See Also

**Reference**

IO.Objects.SimpleIO.mikroBUS Namespace

# Socket Class

Encapsulates mikroBUS sockets.

## Inheritance Hierarchy

SystemObject IO.Objects.SimpleIO.mikroBUSSocket

**Namespace:** IO.Objects.SimpleIO.mikroBUS
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public class Socket
```

The Socket type exposes the following members.

## Constructors

| | Name | Description |
|---|---|---|
| | Socket | Constructor for a single mikroBUS socket. |

Top

## Properties

| | Name | Description |
|---|---|---|
| | AIN | Returns the ADC input designator for AN. |

| | | |
|---|---|---|
| | AN | Returns the GPIO pin designator for AN. |
| | CS | Returns the GPIO pin designator for CS. |
| | I2CBus | Returns the $I^2C$ bus designator for this socket. |
| | INT | Returns the GPIO pin designator for INT. |
| | MISO | Returns the GPIO pin designator for MISO. |
| | MOSI | Returns the GPIO pin designator for MOSI. |
| | PWM | Returns the GPIO pin designator for PWM. |
| | PWMOut | Returns the PWM output designator for this socket. |
| | RST | Returns the GPIO pin designator for RST. |
| | RX | Returns the GPIO pin designator for RX. |
| | SCK | Returns the GPIO pin designator for SCK. |
| | SCL | Returns the GPIO pin designator for SCL. |
| | SDA | Returns the GPIO pin designator for SDA. |
| | SPIDev | Returns the SPI device designator for |

| | | |
|---|---|---|
| | | this socket. |
| | TX | Returns the GPIO pin designator for TX. |
| | UART | Returns the UART device name for this socket. |

Top

## See Also

### Reference

IO.Objects.SimpleIO.mikroBUS Namespace

# Socket Constructor

Constructor for a single mikroBUS socket.

**Namespace:** IO.Objects.SimpleIO.mikroBUS
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public Socket(
        int num,
        ShieldKinds shield = ShieldKinds.Unknown
)
```

### Parameters

*num*
Type: SystemInt32
Socket number.

*shield* **(Optional)**
Type: IO.Objects.SimpleIO.mikroBUSShieldKinds
mikroBUS shield kind. `Shield.Kinds.Unknown` indicates automatic detection using using the `Shield.kind` property.

## See Also

### Reference

Socket Class
IO.Objects.SimpleIO.mikroBUS Namespace

# Socket Properties

The Socket type exposes the following members.

## Properties

| | Name | Description |
|---|---|---|
| | AIN | Returns the ADC input designator for AN. |
| | AN | Returns the GPIO pin designator for AN. |
| | CS | Returns the GPIO pin designator for CS. |
| | I2CBus | Returns the $I^2C$ bus designator for this socket. |
| | INT | Returns the GPIO pin designator for INT. |
| | MISO | Returns the GPIO pin designator for MISO. |
| | MOSI | Returns the GPIO pin designator for MOSI. |
| | PWM | Returns the GPIO pin designator for PWM. |
| | PWMOut | Returns the PWM output designator for this socket. |
| | | |

| | | |
|---|---|---|
| | RST | Returns the GPIO pin designator for RST. |
| | RX | Returns the GPIO pin designator for RX. |
| | SCK | Returns the GPIO pin designator for SCK. |
| | SCL | Returns the GPIO pin designator for SCL. |
| | SDA | Returns the GPIO pin designator for SDA. |
| | SPIDev | Returns the SPI device designator for this socket. |
| | TX | Returns the GPIO pin designator for TX. |
| | UART | Returns the UART device name for this socket. |

Top

# See Also

**Reference**

Socket Class

IO.Objects.SimpleIO.mikroBUS Namespace

# SocketAIN Property

Returns the ADC input designator for AN.

**Namespace:** IO.Objects.SimpleIO.mikroBUS
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public Designator AIN { get; }
```

### Property Value

Type: Designator

## See Also

### Reference

Socket Class
IO.Objects.SimpleIO.mikroBUS Namespace

# SocketAN Property

Returns the GPIO pin designator for AN.

**Namespace:** IO.Objects.SimpleIO.mikroBUS
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public Designator AN { get; }
```

### Property Value

Type: Designator

## See Also

### Reference

Socket Class
IO.Objects.SimpleIO.mikroBUS Namespace

# SocketCS Property

Returns the GPIO pin designator for CS.

**Namespace:** IO.Objects.SimpleIO.mikroBUS
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public Designator CS { get; }
```

### Property Value

Type: Designator

## See Also

### Reference

Socket Class
IO.Objects.SimpleIO.mikroBUS Namespace

# SocketI2CBus Property

Returns the I$^2$C bus designator for this socket.

**Namespace:** IO.Objects.SimpleIO.mikroBUS
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public Designator I2CBus { get; }
```

### Property Value

Type: Designator

## See Also

### Reference

Socket Class
IO.Objects.SimpleIO.mikroBUS Namespace

# SocketINT Property

Returns the GPIO pin designator for INT.

**Namespace:** IO.Objects.SimpleIO.mikroBUS
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public Designator INT { get; }
```

### Property Value

Type: Designator

## See Also

### Reference

Socket Class
IO.Objects.SimpleIO.mikroBUS Namespace

# SocketMISO Property

Returns the GPIO pin designator for MISO.

**Namespace:** IO.Objects.SimpleIO.mikroBUS
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public Designator MISO { get; }
```

### Property Value

Type: Designator

## See Also

### Reference

Socket Class
IO.Objects.SimpleIO.mikroBUS Namespace

# SocketMOSI Property

Returns the GPIO pin designator for MOSI.

**Namespace:** IO.Objects.SimpleIO.mikroBUS
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public Designator MOSI { get; }
```

### Property Value

Type: Designator

## See Also

### Reference

Socket Class
IO.Objects.SimpleIO.mikroBUS Namespace

# SocketPWM Property

Returns the GPIO pin designator for PWM.

**Namespace:** IO.Objects.SimpleIO.mikroBUS
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public Designator PWM { get; }
```

### Property Value

Type: Designator

## See Also

### Reference

Socket Class
IO.Objects.SimpleIO.mikroBUS Namespace

# SocketPWMOut Property

Returns the PWM output designator for this socket.

**Namespace:** IO.Objects.SimpleIO.mikroBUS
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public Designator PWMOut { get; }
```

### Property Value

Type: Designator

## See Also

### Reference

Socket Class
IO.Objects.SimpleIO.mikroBUS Namespace

# SocketRST Property

Returns the GPIO pin designator for RST.

**Namespace:** IO.Objects.SimpleIO.mikroBUS
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public Designator RST { get; }
```

### Property Value

Type: Designator

## See Also

### Reference

Socket Class
IO.Objects.SimpleIO.mikroBUS Namespace

# SocketRX Property

Returns the GPIO pin designator for RX.

**Namespace:** IO.Objects.SimpleIO.mikroBUS
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public Designator RX { get; }
```

### Property Value

Type: Designator

## See Also

### Reference

Socket Class
IO.Objects.SimpleIO.mikroBUS Namespace

# SocketSCK Property

Returns the GPIO pin designator for SCK.

**Namespace:** IO.Objects.SimpleIO.mikroBUS
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public Designator SCK { get; }
```

### Property Value

Type: Designator

## See Also

### Reference

Socket Class
IO.Objects.SimpleIO.mikroBUS Namespace

# SocketSCL Property

Returns the GPIO pin designator for SCL.

**Namespace:** IO.Objects.SimpleIO.mikroBUS
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public Designator SCL { get; }
```

### Property Value

Type: Designator

## See Also

### Reference

Socket Class
IO.Objects.SimpleIO.mikroBUS Namespace

# SocketSDA Property

Returns the GPIO pin designator for SDA.

**Namespace:** IO.Objects.SimpleIO.mikroBUS
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public Designator SDA { get; }
```

### Property Value

Type: Designator

## See Also

### Reference

Socket Class
IO.Objects.SimpleIO.mikroBUS Namespace

# SocketSPIDev Property

Returns the SPI device designator for this socket.

**Namespace:** IO.Objects.SimpleIO.mikroBUS
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public Designator SPIDev { get; }
```

### Property Value

Type: Designator

## See Also

### Reference

Socket Class
IO.Objects.SimpleIO.mikroBUS Namespace

# SocketTX Property

Returns the GPIO pin designator for TX.

**Namespace:** IO.Objects.SimpleIO.mikroBUS
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public Designator TX { get; }
```

### Property Value

Type: Designator

## See Also

### Reference

Socket Class
IO.Objects.SimpleIO.mikroBUS Namespace

# SocketUART Property

Returns the UART device name for this socket.

**Namespace:** IO.Objects.SimpleIO.mikroBUS
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public string UART { get; }
```

### Property Value

Type: String

## See Also

### Reference

Socket Class
IO.Objects.SimpleIO.mikroBUS Namespace

# IO.Objects.SimpleIO.Platforms Namespace

Platform Definition Classes

## Classes

| | Class | Description |
|---|---|---|
| | BeagleBone | This class defines identifiers for the devices provided by the BeagleBone hardware platform. |
| | MUNTS_0018 | This class provides `Designator`s for I/O resources available on the MUNTS-0018 Tutorial I/O Board. |
| | PocketBeagle | This class defines identifiers for the devices provided by the PocketBeagle hardware platform. . |
| | RaspberryPi | This class defines identifiers for the devices provided by the Raspberry Pi hardware platform. |

# BeagleBone Class

This class defines identifiers for the devices provided by the BeagleBone hardware platform.

## Inheritance Hierarchy

SystemObject IO.Objects.SimpleIO.PlatformsBeagleBone

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static class BeagleBone
```

The BeagleBone type exposes the following members.

## Fields

| | Name | Description |
|---|---|---|
| s | AIN0 | ADC input designator (0,0) for P9.39 (1.8V). |
| s | AIN1 | ADC input designator (0,1) for P9.40 (1.8V). |
| s | AIN2 | ADC input designator (0,2) for P9.37 (1.8V). |

| | | |
|---|---|---|
| S | AIN3 | ADC input designator (0,3) for P9.38 (1.8V). |
| S | AIN4 | ADC input designator (0,4) for P9.33 (1.8V). |
| S | AIN5 | ADC input designator (0,5) for P9.36 (1.8V). |
| S | AIN6 | ADC input designator (0,6) for P9.35 (1.8V). |
| S | EHRPWM0A | PWM output designator (1,0) for P9.22 or P9.31. Conflicts with `GPIO2` or `GPIO110`. |
| S | EHRPWM0B | PWM output designator (1,1) for P9.21 or P9.29. Conflicts with `GPIO3` or `GPIO111`. |
| S | EHRPWM1A | PWM output designator (3,0) for P8.36 or P9.14. Conflicts with `GPIO80` or `GPIO50`. |
| S | EHRPWM1B | PWM output designator (3,1) for P8.34 or P9.16. Conflicts with `GPIO81` or `GPIO51`. |
| S | EHRPWM2A | PWM output designator (6,0) for P8.19 or P8.45. Conflicts with `GPIO22` or `GPIO70`. |
| S | EHRPWM2B | PWM output designator (6,1) for P8.13 or P8.46. Conflicts with `GPIO23` or `GPIO71`. |

| | | |
|---|---|---|
| S | GPIO10 | GPIO pin designator (0,10) for P8.31. |
| S | GPIO11 | GPIO pin designator (0,11) for P8.32. |
| S | GPIO110 | GPIO pin designator (3,14) for P9.31. Conflicts with `SPI1 SCLK` or `EHRPWM0A`. |
| S | GPIO111 | GPIO pin designator (3,15) for P9.29. Conflicts with `SPI1 MISO`. |
| S | GPIO112 | GPIO pin designator (3,16) for P9.30. Conflicts with `SPI1 MOSI`. |
| S | GPIO113 | GPIO pin designator (3,17) for P9.28. Conflicts with `SPI1 SS0`. |
| S | GPIO115 | GPIO pin designator (3,19) for P9.27. |
| S | GPIO117 | GPIO pin designator (3,21) for P9.25. |
| S | GPIO12 | GPIO pin designator (0,12) for P9.20. Conflicts with `I2C2 SDA`. |
| S | GPIO13 | GPIO pin designator (0,13) for P9.19. Conflicts with `I2C2 SCL`. |
| S | GPIO14 | GPIO pin designator (0,14) for P9.26. Conflicts with `UART1 RXD`. |
| S | GPIO15 | GPIO pin designator (0,15) for P9.24. Conflicts with `UART1 TXD`. |
| S | GPIO2 | GPIO pin designator (0,2) for P9.22. Conflicts with `SPI0 SCK`, `UART 2RXD` or `EHRPWM0A`. |

| | | |
|---|---|---|
| S | GPIO20 | GPIO pin designator (0,20) for P9.41. |
| S | GPIO22 | GPIO pin designator (0,22) for P8.19. Conflicts with `EHRPWM2A`. |
| S | GPIO23 | GPIO pin designator (0,23) for P8.13. Conflicts with `EHRPWM2B`. |
| S | GPIO26 | GPIO pin designator (0,26) for P8.14. |
| S | GPIO27 | GPIO pin designator (0,27) for P8.17. |
| S | GPIO3 | GPIO pin designator (0,3) for P9.21. Conflicts with `SPI0_MISO` or `UART2_TXD`. |
| S | GPIO30 | GPIO pin designator (0,30) for P9.11. Conflicts with `UART4_RXD`. |
| S | GPIO31 | GPIO pin designator (0,31) for P9.13. Conflicts with `UART4_TXD`. |
| S | GPIO32 | GPIO pin designator (1,0) for P8.25. Conflicts with `MMC_DAT0`. |
| S | GPIO33 | GPIO pin designator (1,1) for P8.24. Conflicts with `MMC1_DAT1`. |
| S | GPIO34 | GPIO pin designator (1,2) for P8.5. Conflicts with `MMC1_DAT2`. |
| S | GPIO35 | GPIO pin designator (1,3) for P8.6. Conflicts with `MMC1_DAT3`. |
| S | GPIO36 | GPIO pin designator (1,4) for P8.23. Conflicts with `MMC_DAT4`. |

| | | |
|---|---|---|
| S | GPIO37 | GPIO pin designator (1,5) for P8.22. Conflicts with `MMC1 DAT5`. |
| S | GPIO38 | GPIO pin designator (1,6) for P8.3. Conflicts with `MMC1 DAT6`. |
| S | GPIO39 | GPIO pin designator (1,7) for P8.4. Conflicts with `MMC1 DAT7`. |
| S | GPIO4 | GPIO pin designator (0,4) for P9.18. Conflicts with `SPI0 MOSI`. |
| S | GPIO44 | GPIO pin designator (1,12) for P8.12. |
| S | GPIO45 | GPIO pin designator (1,13) for P8.11. |
| S | GPIO46 | GPIO pin designator (1,14) for P8.16. |
| S | GPIO47 | GPIO pin designator (1,15) for P8.15. |
| S | GPIO48 | GPIO pin designator (1,16) for P9.15. |
| S | GPIO49 | GPIO pin designator (1,17) for P9.23. |
| S | GPIO5 | GPIO pin designator (0,5) for P9.17. Conflicts with `SPI0 SS0`. |
| S | GPIO50 | GPIO pin designator (1,18) for P9.14. Conflicts with `EHRPWM1A`. |
| S | GPIO51 | GPIO pin designator (1,19) for P9.16. Conflicts with `EHRPWM1B`. |
| S | GPIO60 | GPIO pin designator (1,28) for P9.12. |
| S | GPIO61 | GPIO pin designator (1,29) for P8.26. |

| | | |
|---|---|---|
| S | GPIO62 | GPIO pin designator (1,30) for P8.21. Conflicts with `MMC1_CLK`. |
| S | GPIO63 | GPIO pin designator (1,31) for P8.20. Conflicts with `MMC1_CMD`. |
| S | GPIO65 | GPIO pin designator (2,1) for P8.18. |
| S | GPIO66 | GPIO pin designator (2,2) for P8.7. |
| S | GPIO67 | GPIO pin designator (2,3) for P8.8. |
| S | GPIO68 | GPIO pin designator (2,4) for P8.10. |
| S | GPIO69 | GPIO pin designator (2,5) for P8.9. |
| S | GPIO7 | GPIO pin designator (0,7) for P9.42. Conflicts with `SPI1_SS1`. |
| S | GPIO70 | GPIO pin designator (2,6) for P8.45. Conflicts with `EHRPWM2A`. |
| S | GPIO71 | GPIO pin designator (2,7) for P8.46. Conflicts with `EHRPWM2B`. |
| S | GPIO72 | GPIO pin designator (2,8) for P8.43. |
| S | GPIO73 | GPIO pin designator (2,9) for P8.44. |
| S | GPIO74 | GPIO pin designator (2,10) for P8.41. |
| S | GPIO75 | GPIO pin designator (2,11) for P8.42. |
| S | GPIO76 | GPIO pin designator (2,12) for P8.39. |
| S | GPIO77 | GPIO pin designator (2,13) for P8.40. |

| | | |
|---|---|---|
| S | GPIO78 | GPIO pin designator (2,14) for P8.37. Conflicts with `UART5_TXD`. |
| S | GPIO79 | GPIO pin designator (2,15) for P8.38. Conflicts with `UART5_RXD`. |
| S | GPIO8 | GPIO pin designator (0,8) for P8.35. |
| S | GPIO80 | GPIO pin designator (2,16) for P8.36. Conflicts with `EHRPWM1A`. |
| S | GPIO81 | GPIO pin designator (2,17) for P8.34. Conflicts with `EHRPWM1B`. |
| S | GPIO86 | GPIO pin designator (2,22) for P8.27. |
| S | GPIO87 | GPIO pin designator (2,23) for P8.29. |
| S | GPIO88 | GPIO pin designator (2,24) for P8.28. |
| S | GPIO89 | GPIO pin designator (2,25) for P8.30. |
| S | GPIO9 | GPIO pin designator (0,9) for P8.33. |
| S | I2C2 | I2C bus designator for P9.19 and P9.20. Conflicts with `GPIO13` or `GPIO12`. |
| S | SPI0_0 | SPI slave designator (0,0) for P9.17. Conflicts with `SPI0_SS0` or `GPIO5`. |
| S | SPI1_0 | SPI slave designator (1,0) for P9.28. Conflicts with `SP1_SS0` or `GPIO113`. |
| S | SPI1_1 | SPI slave designator (1,1) for P9.42. Conflicts with `SPI1_SS1` or `GPIO7`. |

Top

## See Also

### Reference

IO.Objects.SimpleIO.Platforms Namespace

# BeagleBone Fields

The BeagleBone type exposes the following members.

## Fields

| | Name | Description |
|---|---|---|
| s | AIN0 | ADC input designator (0,0) for P9.39 (1.8V). |
| s | AIN1 | ADC input designator (0,1) for P9.40 (1.8V). |
| s | AIN2 | ADC input designator (0,2) for P9.37 (1.8V). |
| s | AIN3 | ADC input designator (0,3) for P9.38 (1.8V). |
| s | AIN4 | ADC input designator (0,4) for P9.33 (1.8V). |
| s | AIN5 | ADC input designator (0,5) for P9.36 (1.8V). |
| s | AIN6 | ADC input designator (0,6) for P9.35 (1.8V). |
| s | EHRPWM0A | PWM output designator (1,0) for P9.22 or P9.31. Conflicts with `GPIO2` or `GPIO110`. |

| | | |
|---|---|---|
| S | EHRPWM0B | PWM output designator (1,1) for P9.21 or P9.29. Conflicts with `GPIO3` or `GPIO111`. |
| S | EHRPWM1A | PWM output designator (3,0) for P8.36 or P9.14. Conflicts with `GPIO80` or `GPIO50`. |
| S | EHRPWM1B | PWM output designator (3,1) for P8.34 or P9.16. Conflicts with `GPIO81` or `GPIO51`. |
| S | EHRPWM2A | PWM output designator (6,0) for P8.19 or P8.45. Conflicts with `GPIO22` or `GPIO70`. |
| S | EHRPWM2B | PWM output designator (6,1) for P8.13 or P8.46. Conflicts with `GPIO23` or `GPIO71`. |
| S | GPIO10 | GPIO pin designator (0,10) for P8.31. |
| S | GPIO11 | GPIO pin designator (0,11) for P8.32. |
| S | GPIO110 | GPIO pin designator (3,14) for P9.31. Conflicts with `SPI1 SCLK` or `EHRPWM0A`. |
| S | GPIO111 | GPIO pin designator (3,15) for P9.29. Conflicts with `SPI1 MISO`. |
| S | GPIO112 | GPIO pin designator (3,16) for P9.30. Conflicts with `SPI1 MOSI`. |
| S | GPIO113 | GPIO pin designator (3,17) for P9.28. |

| | | |
|---|---|---|
| | | Conflicts with `SPI1 SS0`. |
| S | GPIO115 | GPIO pin designator (3,19) for P9.27. |
| S | GPIO117 | GPIO pin designator (3,21) for P9.25. |
| S | GPIO12 | GPIO pin designator (0,12) for P9.20. Conflicts with `I2C2 SDA`. |
| S | GPIO13 | GPIO pin designator (0,13) for P9.19. Conflicts with `I2C2 SCL`. |
| S | GPIO14 | GPIO pin designator (0,14) for P9.26. Conflicts with `UART1 RXD`. |
| S | GPIO15 | GPIO pin designator (0,15) for P9.24. Conflicts with `UART1 TXD`. |
| S | GPIO2 | GPIO pin designator (0,2) for P9.22. Conflicts with `SPI0 SCK`, `UART 2RXD` or `EHRPWM0A`. |
| S | GPIO20 | GPIO pin designator (0,20) for P9.41. |
| S | GPIO22 | GPIO pin designator (0,22) for P8.19. Conflicts with `EHRPWM2A`. |
| S | GPIO23 | GPIO pin designator (0,23) for P8.13. Conflicts with `EHRPWM2B`. |
| S | GPIO26 | GPIO pin designator (0,26) for P8.14. |
| S | GPIO27 | GPIO pin designator (0,27) for P8.17. |
| S | GPIO3 | GPIO pin designator (0,3) for P9.21. Conflicts with `SPI0 MISO` or `UART2` |

| | | |
|---|---|---|
| | | `TXD`. |
| S | GPIO30 | GPIO pin designator (0,30) for P9.11. Conflicts with `UART4_RXD`. |
| S | GPIO31 | GPIO pin designator (0,31) for P9.13. Conflicts with `UART4_TXD`. |
| S | GPIO32 | GPIO pin designator (1,0) for P8.25. Conflicts with `MMC_DAT0`. |
| S | GPIO33 | GPIO pin designator (1,1) for P8.24. Conflicts with `MMC1_DAT1`. |
| S | GPIO34 | GPIO pin designator (1,2) for P8.5. Conflicts with `MMC1_DAT2`. |
| S | GPIO35 | GPIO pin designator (1,3) for P8.6. Conflicts with `MMC1_DAT3`. |
| S | GPIO36 | GPIO pin designator (1,4) for P8.23. Conflicts with `MMC_DAT4`. |
| S | GPIO37 | GPIO pin designator (1,5) for P8.22. Conflicts with `MMC1_DAT5`. |
| S | GPIO38 | GPIO pin designator (1,6) for P8.3. Conflicts with `MMC1_DAT6`. |
| S | GPIO39 | GPIO pin designator (1,7) for P8.4. Conflicts with `MMC1_DAT7`. |
| S | GPIO4 | GPIO pin designator (0,4) for P9.18. Conflicts with `SPI0_MOSI`. |
| S | GPIO44 | GPIO pin designator (1,12) for P8.12. |

| | | |
|---|---|---|
| S | GPIO45 | GPIO pin designator (1,13) for P8.11. |
| S | GPIO46 | GPIO pin designator (1,14) for P8.16. |
| S | GPIO47 | GPIO pin designator (1,15) for P8.15. |
| S | GPIO48 | GPIO pin designator (1,16) for P9.15. |
| S | GPIO49 | GPIO pin designator (1,17) for P9.23. |
| S | GPIO5 | GPIO pin designator (0,5) for P9.17. Conflicts with `SPI0 SS0`. |
| S | GPIO50 | GPIO pin designator (1,18) for P9.14. Conflicts with `EHRPWM1A`. |
| S | GPIO51 | GPIO pin designator (1,19) for P9.16. Conflicts with `EHRPWM1B`. |
| S | GPIO60 | GPIO pin designator (1,28) for P9.12. |
| S | GPIO61 | GPIO pin designator (1,29) for P8.26. |
| S | GPIO62 | GPIO pin designator (1,30) for P8.21. Conflicts with `MMC1 CLK`. |
| S | GPIO63 | GPIO pin designator (1,31) for P8.20. Conflicts with `MMC1 CMD`. |
| S | GPIO65 | GPIO pin designator (2,1) for P8.18. |
| S | GPIO66 | GPIO pin designator (2,2) for P8.7. |
| S | GPIO67 | GPIO pin designator (2,3) for P8.8. |
| S | GPIO68 | GPIO pin designator (2,4) for P8.10. |

| | | |
|---|---|---|
| S | GPIO69 | GPIO pin designator (2,5) for P8.9. |
| S | GPIO7 | GPIO pin designator (0,7) for P9.42. Conflicts with `SPI1_SS1`. |
| S | GPIO70 | GPIO pin designator (2,6) for P8.45. Conflicts with `EHRPWM2A`. |
| S | GPIO71 | GPIO pin designator (2,7) for P8.46. Conflicts with `EHRPWM2B`. |
| S | GPIO72 | GPIO pin designator (2,8) for P8.43. |
| S | GPIO73 | GPIO pin designator (2,9) for P8.44. |
| S | GPIO74 | GPIO pin designator (2,10) for P8.41. |
| S | GPIO75 | GPIO pin designator (2,11) for P8.42. |
| S | GPIO76 | GPIO pin designator (2,12) for P8.39. |
| S | GPIO77 | GPIO pin designator (2,13) for P8.40. |
| S | GPIO78 | GPIO pin designator (2,14) for P8.37. Conflicts with `UART5_TXD`. |
| S | GPIO79 | GPIO pin designator (2,15) for P8.38. Conflicts with `UART5_RXD`. |
| S | GPIO8 | GPIO pin designator (0,8) for P8.35. |
| S | GPIO80 | GPIO pin designator (2,16) for P8.36. Conflicts with `EHRPWM1A`. |
| S | GPIO81 | GPIO pin designator (2,17) for P8.34. Conflicts with `EHRPWM1B`. |

| | | |
|---|---|---|
| S | GPIO86 | GPIO pin designator (2,22) for P8.27. |
| S | GPIO87 | GPIO pin designator (2,23) for P8.29. |
| S | GPIO88 | GPIO pin designator (2,24) for P8.28. |
| S | GPIO89 | GPIO pin designator (2,25) for P8.30. |
| S | GPIO9 | GPIO pin designator (0,9) for P8.33. |
| S | I2C2 | I2C bus designator for P9.19 and P9.20. Conflicts with `GPIO13` or `GPIO12`. |
| S | SPI0_0 | SPI slave designator (0,0) for P9.17. Conflicts with `SPI0 SS0` or `GPIO5`. |
| S | SPI1_0 | SPI slave designator (1,0) for P9.28. Conflicts with `SP1 SS0` or `GPIO113`. |
| S | SPI1_1 | SPI slave designator (1,1) for P9.42. Conflicts with `SPI1 SS1` or `GPIO7`. |

Top

## See Also

### Reference

BeagleBone Class

IO.Objects.SimpleIO.Platforms Namespace

# BeagleBoneAIN0 Field

ADC input designator (0,0) for P9.39 (1.8V).

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static readonly Designator AIN0
```

### Field Value

Type: Designator

## Remarks

Requires the `BB-ADC` device tree overlay.

## See Also

### Reference

BeagleBone Class
IO.Objects.SimpleIO.Platforms Namespace

# BeagleBoneAIN1 Field

ADC input designator (0,1) for P9.40 (1.8V).

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator AIN1
```

### Field Value

Type: Designator

## Remarks

Requires the `BB-ADC` device tree overlay.

## See Also

### Reference

BeagleBone Class
IO.Objects.SimpleIO.Platforms Namespace

# BeagleBoneAIN2 Field

ADC input designator (0,2) for P9.37 (1.8V).

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator AIN2
```

### Field Value

Type: Designator

## Remarks

Requires the `BB-ADC` device tree overlay.

## See Also

### Reference

BeagleBone Class
IO.Objects.SimpleIO.Platforms Namespace

# BeagleBoneAIN3 Field

ADC input designator (0,3) for P9.38 (1.8V).

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator AIN3
```

### Field Value

Type: Designator

## Remarks

Requires the `BB-ADC` device tree overlay.

## See Also

### Reference

BeagleBone Class
IO.Objects.SimpleIO.Platforms Namespace

# BeagleBoneAIN4 Field

ADC input designator (0,4) for P9.33 (1.8V).

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator AIN4
```

### Field Value

Type: Designator

## Remarks

Requires the `BB-ADC` device tree overlay.

## See Also

### Reference

BeagleBone Class
IO.Objects.SimpleIO.Platforms Namespace

# BeagleBoneAIN5 Field

ADC input designator (0,5) for P9.36 (1.8V).

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator AIN5
```

### Field Value

Type: Designator

## Remarks

Requires the `BB-ADC` device tree overlay.

## See Also

### Reference

BeagleBone Class
IO.Objects.SimpleIO.Platforms Namespace

# BeagleBoneAIN6 Field

ADC input designator (0,6) for P9.35 (1.8V).

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator AIN6
```

### Field Value

Type: Designator

## Remarks

Requires the `BB-ADC` device tree overlay.

## See Also

### Reference

BeagleBone Class
IO.Objects.SimpleIO.Platforms Namespace

# BeagleBoneEHRPWM0A Field

PWM output designator (1,0) for P9.22 or P9.31. Conflicts with `GPIO2` or `GPIO110`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator EHRPWM0A
```

### Field Value

Type: Designator

## Remarks

Requires the `BB-GPIO` device tree overlay.

## See Also

### Reference

BeagleBone Class
IO.Objects.SimpleIO.Platforms Namespace

# BeagleBoneEHRPWM0B Field

PWM output designator (1,1) for P9.21 or P9.29. Conflicts with `GPIO3` or `GPIO111`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static readonly Designator EHRPWM0B
```

### Field Value

Type: Designator

## Remarks

Requires the `BB-GPIO` device tree overlay.

## See Also

### Reference

BeagleBone Class
IO.Objects.SimpleIO.Platforms Namespace

# BeagleBoneEHRPWM1A Field

PWM output designator (3,0) for P8.36 or P9.14. Conflicts with `GPIO80` or `GPIO50`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static readonly Designator EHRPWM1A
```

### Field Value

Type: Designator

## Remarks

Requires the `BB-GPIO` device tree overlay.

## See Also

### Reference

BeagleBone Class
IO.Objects.SimpleIO.Platforms Namespace

# BeagleBoneEHRPWM1B Field

PWM output designator (3,1) for P8.34 or P9.16. Conflicts with `GPIO81` or `GPIO51`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static readonly Designator EHRPWM1B
```

### Field Value

Type: Designator

## Remarks

Requires the `BB-GPIO` device tree overlay.

## See Also

### Reference

BeagleBone Class
IO.Objects.SimpleIO.Platforms Namespace

# BeagleBoneEHRPWM2A Field

PWM output designator (6,0) for P8.19 or P8.45. Conflicts with `GPIO22` or `GPIO70`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator EHRPWM2A
```

### Field Value

Type: Designator

## Remarks

Requires the `BB-GPIO` device tree overlay.

## See Also

### Reference

BeagleBone Class
IO.Objects.SimpleIO.Platforms Namespace

# BeagleBoneEHRPWM2B Field

PWM output designator (6,1) for P8.13 or P8.46. Conflicts with `GPIO23` or `GPIO71`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator EHRPWM2B
```

### Field Value

Type: Designator

## Remarks

Requires the `BB-GPIO` device tree overlay.

## See Also

### Reference

BeagleBone Class
IO.Objects.SimpleIO.Platforms Namespace

# BeagleBoneGPIO10 Field

GPIO pin designator (0,10) for P8.31.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO10
```

### Field Value

Type: Designator

## Remarks

Requires the `BB-GPIO` device tree overlay.

## See Also

### Reference

BeagleBone Class
IO.Objects.SimpleIO.Platforms Namespace

# BeagleBoneGPIO11 Field

GPIO pin designator (0,11) for P8.32.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static readonly Designator GPIO11
```

### Field Value

Type: Designator

## Remarks

Requires the `BB-GPIO` device tree overlay.

## See Also

### Reference

BeagleBone Class
IO.Objects.SimpleIO.Platforms Namespace

# BeagleBoneGPIO110 Field

GPIO pin designator (3,14) for P9.31. Conflicts with `SPI1 SCLK` or `EHRPWM0A`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO110
```

### Field Value

Type: Designator

## Remarks

Requires the `BB-GPIO` device tree overlay.

## See Also

### Reference

BeagleBone Class
IO.Objects.SimpleIO.Platforms Namespace

# BeagleBoneGPIO111 Field

GPIO pin designator (3,15) for P9.29. Conflicts with `SPI1 MISO`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO111
```

### Field Value

Type: Designator

## Remarks

Requires the `BB-GPIO` device tree overlay.

## See Also

### Reference

BeagleBone Class
IO.Objects.SimpleIO.Platforms Namespace

# BeagleBoneGPIO112 Field

GPIO pin designator (3,16) for P9.30. Conflicts with `SPI1 MOSI`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO112
```

### Field Value

Type: Designator

## Remarks

Requires the `BB-GPIO` device tree overlay.

## Remarks

This pin is also `SPI1 MOSI`.

## See Also

### Reference

BeagleBone Class
IO.Objects.SimpleIO.Platforms Namespace

# BeagleBoneGPIO113 Field

GPIO pin designator (3,17) for P9.28. Conflicts with `SPI1 SS0`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO113
```

### Field Value

Type: Designator

## Remarks

Requires the `BB-GPIO` device tree overlay.

## See Also

### Reference

BeagleBone Class
IO.Objects.SimpleIO.Platforms Namespace

# BeagleBoneGPIO115 Field

GPIO pin designator (3,19) for P9.27.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO115
```

### Field Value

Type: Designator

## Remarks

Requires the `BB-GPIO` device tree overlay.

## See Also

### Reference

BeagleBone Class
IO.Objects.SimpleIO.Platforms Namespace

# BeagleBoneGPIO117 Field

GPIO pin designator (3,21) for P9.25.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO117
```

### Field Value

Type: Designator

## Remarks

Requires the `BB-GPIO` device tree overlay.

## See Also

### Reference

BeagleBone Class
IO.Objects.SimpleIO.Platforms Namespace

# BeagleBoneGPIO12 Field

GPIO pin designator (0,12) for P9.20. Conflicts with `I2C2 SDA`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO12
```

### Field Value

Type: Designator

## Remarks

Requires the `BB-GPIO` device tree overlay.

## See Also

### Reference

BeagleBone Class
IO.Objects.SimpleIO.Platforms Namespace

# BeagleBoneGPIO13 Field

GPIO pin designator (0,13) for P9.19. Conflicts with `I2C2 SCL`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO13
```

### Field Value

Type: Designator

## Remarks

Requires the `BB-GPIO` device tree overlay.

## See Also

### Reference

BeagleBone Class
IO.Objects.SimpleIO.Platforms Namespace

# BeagleBoneGPIO14 Field

GPIO pin designator (0,14) for P9.26. Conflicts with `UART1 RXD`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO14
```

### Field Value

Type: Designator

## Remarks

Requires the `BB-GPIO` device tree overlay.

## See Also

### Reference

BeagleBone Class
IO.Objects.SimpleIO.Platforms Namespace

# BeagleBoneGPIO15 Field

GPIO pin designator (0,15) for P9.24. Conflicts with `UART1 TXD`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static readonly Designator GPIO15
```

### Field Value

Type: Designator

## Remarks

Requires the `BB-GPIO` device tree overlay.

## See Also

### Reference

BeagleBone Class
IO.Objects.SimpleIO.Platforms Namespace

# BeagleBoneGPIO2 Field

GPIO pin designator (0,2) for P9.22. Conflicts with `SPI0 SCK`, `UART 2RXD` or `EHRPWM0A`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO2
```

### Field Value

Type: Designator

## Remarks

Requires the `BB-GPIO` device tree overlay.

## See Also

### Reference

BeagleBone Class
IO.Objects.SimpleIO.Platforms Namespace

# BeagleBoneGPIO20 Field

GPIO pin designator (0,20) for P9.41.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static readonly Designator GPIO20
```

### Field Value

Type: Designator

## Remarks

Requires the `BB-GPIO` device tree overlay.

## See Also

### Reference

BeagleBone Class
IO.Objects.SimpleIO.Platforms Namespace

# BeagleBoneGPIO22 Field

GPIO pin designator (0,22) for P8.19. Conflicts with `EHRPWM2A`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static readonly Designator GPIO22
```

### Field Value

Type: Designator

## Remarks

Requires the `BB-GPIO` device tree overlay.

## See Also

### Reference

BeagleBone Class
IO.Objects.SimpleIO.Platforms Namespace

# BeagleBoneGPIO23 Field

GPIO pin designator (0,23) for P8.13. Conflicts with `EHRPWM2B`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO23
```

### Field Value

Type: Designator

## Remarks

Requires the `BB-GPIO` device tree overlay.

## See Also

### Reference

BeagleBone Class
IO.Objects.SimpleIO.Platforms Namespace

# BeagleBoneGPIO26 Field

GPIO pin designator (0,26) for P8.14.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO26
```

### Field Value

Type: Designator

## Remarks

Requires the `BB-GPIO` device tree overlay.

## See Also

### Reference

BeagleBone Class
IO.Objects.SimpleIO.Platforms Namespace

# BeagleBoneGPIO27 Field

GPIO pin designator (0,27) for P8.17.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO27
```

### Field Value

Type: Designator

## Remarks

Requires the `BB-GPIO` device tree overlay.

## See Also

### Reference

BeagleBone Class
IO.Objects.SimpleIO.Platforms Namespace

# BeagleBoneGPIO3 Field

GPIO pin designator (0,3) for P9.21. Conflicts with `SPI0 MISO` or `UART2 TXD`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static readonly Designator GPIO3
```

### Field Value

Type: Designator

## Remarks

Requires the `BB-GPIO` device tree overlay.

## See Also

### Reference

BeagleBone Class
IO.Objects.SimpleIO.Platforms Namespace

# BeagleBoneGPIO30 Field

GPIO pin designator (0,30) for P9.11. Conflicts with `UART4 RXD`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static readonly Designator GPIO30
```

### Field Value

Type: Designator

## Remarks

Requires the `BB-GPIO` device tree overlay.

## See Also

### Reference

BeagleBone Class
IO.Objects.SimpleIO.Platforms Namespace

# BeagleBoneGPIO31 Field

GPIO pin designator (0,31) for P9.13. Conflicts with `UART4 TXD`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO31
```

### Field Value

Type: Designator

## Remarks

Requires the `BB-GPIO` device tree overlay.

## See Also

### Reference

BeagleBone Class
IO.Objects.SimpleIO.Platforms Namespace

# BeagleBoneGPIO32 Field

GPIO pin designator (1,0) for P8.25. Conflicts with `MMC DAT0`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static readonly Designator GPIO32
```

### Field Value

Type: Designator

## Remarks

Requires the BeagleBone White and the `BB-NOEMMC` device tree overlay.

## See Also

### Reference

BeagleBone Class
IO.Objects.SimpleIO.Platforms Namespace

# BeagleBoneGPIO33 Field

GPIO pin designator (1,1) for P8.24. Conflicts with `MMC1 DAT1`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO33
```

### Field Value

Type: Designator

## Remarks

Requires the BeagleBone White and the `BB-NOEMMC` device tree overlay.

## See Also

### Reference

BeagleBone Class
IO.Objects.SimpleIO.Platforms Namespace

# BeagleBoneGPIO34 Field

GPIO pin designator (1,2) for P8.5. Conflicts with `MMC1 DAT2`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static readonly Designator GPIO34
```

### Field Value

Type: Designator

## Remarks

Requires the BeagleBone White and the `BB-NOEMMC` device tree overlay.

## See Also

### Reference

BeagleBone Class
IO.Objects.SimpleIO.Platforms Namespace

# BeagleBoneGPIO35 Field

GPIO pin designator (1,3) for P8.6. Conflicts with `MMC1 DAT3`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO35
```

### Field Value

Type: Designator

## Remarks

Requires the BeagleBone White and the `BB-NOEMMC` device tree overlay.

## See Also

### Reference

BeagleBone Class
IO.Objects.SimpleIO.Platforms Namespace

# BeagleBoneGPIO36 Field

GPIO pin designator (1,4) for P8.23. Conflicts with `MMC DAT4`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO36
```

### Field Value

Type: Designator

## Remarks

Requires the BeagleBone White and the `BB-NOEMMC` device tree overlay.

## See Also

### Reference

BeagleBone Class
IO.Objects.SimpleIO.Platforms Namespace

# BeagleBoneGPIO37 Field

GPIO pin designator (1,5) for P8.22. Conflicts with `MMC1 DAT5`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO37
```

### Field Value

Type: Designator

## Remarks

Requires the BeagleBone White and the `BB-NOEMMC` device tree overlay.

## See Also

### Reference

BeagleBone Class
IO.Objects.SimpleIO.Platforms Namespace

# BeagleBoneGPIO38 Field

GPIO pin designator (1,6) for P8.3. Conflicts with `MMC1 DAT6`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO38
```

### Field Value

Type: Designator

## Remarks

Requires the BeagleBone White and the `BB-NOEMMC` device tree overlay.

## See Also

### Reference

BeagleBone Class
IO.Objects.SimpleIO.Platforms Namespace

# BeagleBoneGPIO39 Field

GPIO pin designator (1,7) for P8.4. Conflicts with `MMC1 DAT7`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static readonly Designator GPIO39
```

### Field Value

Type: Designator

## Remarks

Requires the BeagleBone White and the `BB-NOEMMC` device tree overlay.

## See Also

### Reference

BeagleBone Class
IO.Objects.SimpleIO.Platforms Namespace

# BeagleBoneGPIO4 Field

GPIO pin designator (0,4) for P9.18. Conflicts with `SPI0 MOSI`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO4
```

### Field Value

Type: Designator

## Remarks

Requires the `BB-GPIO` device tree overlay.

## See Also

### Reference

BeagleBone Class
IO.Objects.SimpleIO.Platforms Namespace

# BeagleBoneGPIO44 Field

GPIO pin designator (1,12) for P8.12.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO44
```

### Field Value

Type: Designator

## Remarks

Requires the `BB-GPIO` device tree overlay.

## See Also

### Reference

BeagleBone Class
IO.Objects.SimpleIO.Platforms Namespace

# BeagleBoneGPIO45 Field

GPIO pin designator (1,13) for P8.11.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO45
```

### Field Value

Type: Designator

## Remarks

Requires the `BB-GPIO` device tree overlay.

## See Also

### Reference

BeagleBone Class
IO.Objects.SimpleIO.Platforms Namespace

# BeagleBoneGPIO46 Field

GPIO pin designator (1,14) for P8.16.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static readonly Designator GPIO46
```

### Field Value

Type: Designator

## Remarks

Requires the `BB-GPIO` device tree overlay.

## See Also

### Reference

BeagleBone Class
IO.Objects.SimpleIO.Platforms Namespace

# BeagleBoneGPIO47 Field

GPIO pin designator (1,15) for P8.15.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO47
```

### Field Value

Type: Designator

## Remarks

Requires the `BB-GPIO` device tree overlay.

## See Also

### Reference

BeagleBone Class
IO.Objects.SimpleIO.Platforms Namespace

# BeagleBoneGPIO48 Field

GPIO pin designator (1,16) for P9.15.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static readonly Designator GPIO48
```

### Field Value

Type: Designator

## Remarks

Requires the `BB-GPIO` device tree overlay.

## See Also

### Reference

BeagleBone Class
IO.Objects.SimpleIO.Platforms Namespace

# BeagleBoneGPIO49 Field

GPIO pin designator (1,17) for P9.23.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO49
```

### Field Value

Type: Designator

## Remarks

Requires the `BB-GPIO` device tree overlay.

## See Also

### Reference

BeagleBone Class
IO.Objects.SimpleIO.Platforms Namespace

# BeagleBoneGPIO5 Field

GPIO pin designator (0,5) for P9.17. Conflicts with `SPI0 SS0`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static readonly Designator GPIO5
```

### Field Value

Type: Designator

## Remarks

Requires the `BB-GPIO` device tree overlay.

## See Also

### Reference

BeagleBone Class
IO.Objects.SimpleIO.Platforms Namespace

# BeagleBoneGPIO50 Field

GPIO pin designator (1,18) for P9.14. Conflicts with `EHRPWM1A`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO50
```

### Field Value

Type: Designator

## Remarks

Requires the `BB-GPIO` device tree overlay.

## See Also

### Reference

BeagleBone Class
IO.Objects.SimpleIO.Platforms Namespace

# BeagleBoneGPIO51 Field

GPIO pin designator (1,19) for P9.16. Conflicts with `EHRPWM1B`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO51
```

### Field Value

Type: Designator

## Remarks

Requires the `BB-GPIO` device tree overlay.

## See Also

### Reference

BeagleBone Class
IO.Objects.SimpleIO.Platforms Namespace

# BeagleBoneGPIO60 Field

GPIO pin designator (1,28) for P9.12.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO60
```

### Field Value

Type: Designator

## Remarks

Requires the `BB-GPIO` device tree overlay.

## See Also

### Reference

BeagleBone Class
IO.Objects.SimpleIO.Platforms Namespace

# BeagleBoneGPIO61 Field

GPIO pin designator (1,29) for P8.26.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO61
```

### Field Value

Type: Designator

## Remarks

Requires the `BB-GPIO` device tree overlay.

## See Also

### Reference

BeagleBone Class
IO.Objects.SimpleIO.Platforms Namespace

# BeagleBoneGPIO62 Field

GPIO pin designator (1,30) for P8.21. Conflicts with `MMC1 CLK`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static readonly Designator GPIO62
```

### Field Value

Type: Designator

## Remarks

Requires the BeagleBone White and the `BB-NOEMMC` device tree overlay.

## See Also

### Reference

BeagleBone Class
IO.Objects.SimpleIO.Platforms Namespace

# BeagleBoneGPIO63 Field

GPIO pin designator (1,31) for P8.20. Conflicts with `MMC1 CMD`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static readonly Designator GPIO63
```

### Field Value

Type: Designator

## Remarks

Requires the BeagleBone White and the `BB-NOEMMC` device tree overlay.

## See Also

### Reference

BeagleBone Class
IO.Objects.SimpleIO.Platforms Namespace

# BeagleBoneGPIO65 Field

GPIO pin designator (2,1) for P8.18.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static readonly Designator GPIO65
```

### Field Value

Type: Designator

## Remarks

Requires the `BB-GPIO` device tree overlay.

## See Also

### Reference

BeagleBone Class
IO.Objects.SimpleIO.Platforms Namespace

# BeagleBoneGPIO66 Field

GPIO pin designator (2,2) for P8.7.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO66
```

### Field Value

Type: Designator

## Remarks

Requires the `BB-GPIO` device tree overlay.

## See Also

### Reference

BeagleBone Class
IO.Objects.SimpleIO.Platforms Namespace

# BeagleBoneGPIO67 Field

GPIO pin designator (2,3) for P8.8.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static readonly Designator GPIO67
```

### Field Value

Type: Designator

## Remarks

Requires the `BB-GPIO` device tree overlay.

## See Also

### Reference

BeagleBone Class
IO.Objects.SimpleIO.Platforms Namespace

# BeagleBoneGPIO68 Field

GPIO pin designator (2,4) for P8.10.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static readonly Designator GPIO68
```

### Field Value

Type: Designator

## Remarks

Requires the `BB-GPIO` device tree overlay.

## See Also

### Reference

BeagleBone Class
IO.Objects.SimpleIO.Platforms Namespace

# BeagleBoneGPIO69 Field

GPIO pin designator (2,5) for P8.9.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static readonly Designator GPIO69
```

### Field Value

Type: Designator

## Remarks

Requires the `BB-GPIO` device tree overlay.

## See Also

### Reference

BeagleBone Class
IO.Objects.SimpleIO.Platforms Namespace

# BeagleBoneGPIO7 Field

GPIO pin designator (0,7) for P9.42. Conflicts with `SPI1 SS1`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static readonly Designator GPIO7
```

### Field Value

Type: Designator

## Remarks

Requires the `BB-GPIO` device tree overlay.

## See Also

### Reference

BeagleBone Class
IO.Objects.SimpleIO.Platforms Namespace

# BeagleBoneGPIO70 Field

GPIO pin designator (2,6) for P8.45. Conflicts with `EHRPWM2A`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO70
```

### Field Value

Type: Designator

## Remarks

Requires the `BB-GPIO` device tree overlay.

## See Also

### Reference

BeagleBone Class
IO.Objects.SimpleIO.Platforms Namespace

# BeagleBoneGPIO71 Field

GPIO pin designator (2,7) for P8.46. Conflicts with `EHRPWM2B`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO71
```

### Field Value

Type: Designator

## Remarks

Requires the `BB-GPIO` device tree overlay.

## See Also

### Reference

BeagleBone Class
IO.Objects.SimpleIO.Platforms Namespace

# BeagleBoneGPIO72 Field

GPIO pin designator (2,8) for P8.43.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static readonly Designator GPIO72
```

### Field Value

Type: Designator

## Remarks

Requires the `BB-GPIO` device tree overlay.

## See Also

### Reference

BeagleBone Class
IO.Objects.SimpleIO.Platforms Namespace

# BeagleBoneGPIO73 Field

GPIO pin designator (2,9) for P8.44.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static readonly Designator GPIO73
```

### Field Value

Type: Designator

## Remarks

Requires the `BB-GPIO` device tree overlay.

## See Also

### Reference

BeagleBone Class
IO.Objects.SimpleIO.Platforms Namespace

# BeagleBoneGPIO74 Field

GPIO pin designator (2,10) for P8.41.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO74
```

### Field Value

Type: Designator

## Remarks

Requires the `BB-GPIO` device tree overlay.

## See Also

### Reference

BeagleBone Class
IO.Objects.SimpleIO.Platforms Namespace

# BeagleBoneGPIO75 Field

GPIO pin designator (2,11) for P8.42.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO75
```

### Field Value

Type: Designator

## Remarks

Requires the `BB-GPIO` device tree overlay.

## See Also

### Reference

BeagleBone Class
IO.Objects.SimpleIO.Platforms Namespace

# BeagleBoneGPIO76 Field

GPIO pin designator (2,12) for P8.39.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static readonly Designator GPIO76
```

### Field Value

Type: Designator

## Remarks

Requires the `BB-GPIO` device tree overlay.

## See Also

### Reference

BeagleBone Class
IO.Objects.SimpleIO.Platforms Namespace

# BeagleBoneGPIO77 Field

GPIO pin designator (2,13) for P8.40.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO77
```

### Field Value

Type: Designator

## Remarks

Requires the `BB-GPIO` device tree overlay.

## See Also

### Reference

BeagleBone Class
IO.Objects.SimpleIO.Platforms Namespace

# BeagleBoneGPIO78 Field

GPIO pin designator (2,14) for P8.37. Conflicts with `UART5 TXD`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static readonly Designator GPIO78
```

### Field Value

Type: Designator

## Remarks

Requires the `BB-GPIO` device tree overlay.

## See Also

### Reference

BeagleBone Class
IO.Objects.SimpleIO.Platforms Namespace

# BeagleBoneGPIO79 Field

GPIO pin designator (2,15) for P8.38. Conflicts with `UART5 RXD`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO79
```

### Field Value

Type: Designator

## Remarks

Requires the `BB-GPIO` device tree overlay.

## See Also

### Reference

BeagleBone Class
IO.Objects.SimpleIO.Platforms Namespace

# BeagleBoneGPIO8 Field

GPIO pin designator (0,8) for P8.35.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO8
```

### Field Value

Type: Designator

## Remarks

Requires the `BB-GPIO` device tree overlay.

## See Also

### Reference

BeagleBone Class
IO.Objects.SimpleIO.Platforms Namespace

# BeagleBoneGPIO80 Field

GPIO pin designator (2,16) for P8.36. Conflicts with `EHRPWM1A`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static readonly Designator GPIO80
```

### Field Value

Type: Designator

## Remarks

Requires the `BB-GPIO` device tree overlay.

## See Also

### Reference

BeagleBone Class
IO.Objects.SimpleIO.Platforms Namespace

# BeagleBoneGPIO81 Field

GPIO pin designator (2,17) for P8.34. Conflicts with `EHRPWM1B`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO81
```

### Field Value

Type: Designator

## Remarks

Requires the `BB-GPIO` device tree overlay.

## See Also

### Reference

BeagleBone Class
IO.Objects.SimpleIO.Platforms Namespace

# BeagleBoneGPIO86 Field

GPIO pin designator (2,22) for P8.27.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO86
```

### Field Value

Type: Designator

## Remarks

Requires the `BB-GPIO` device tree overlay.

## See Also

### Reference

BeagleBone Class
IO.Objects.SimpleIO.Platforms Namespace

# BeagleBoneGPIO87 Field

GPIO pin designator (2,23) for P8.29.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static readonly Designator GPIO87
```

### Field Value

Type: Designator

## Remarks

Requires the `BB-GPIO` device tree overlay.

## See Also

### Reference

BeagleBone Class
IO.Objects.SimpleIO.Platforms Namespace

# BeagleBoneGPIO88 Field

GPIO pin designator (2,24) for P8.28.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static readonly Designator GPIO88
```

### Field Value

Type: Designator

## Remarks

Requires the `BB-GPIO` device tree overlay.

## See Also

### Reference

BeagleBone Class
IO.Objects.SimpleIO.Platforms Namespace

# BeagleBoneGPIO89 Field

GPIO pin designator (2,25) for P8.30.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static readonly Designator GPIO89
```

### Field Value

Type: Designator

## Remarks

Requires the `BB-GPIO` device tree overlay.

## See Also

### Reference

BeagleBone Class
IO.Objects.SimpleIO.Platforms Namespace

# BeagleBoneGPIO9 Field

GPIO pin designator (0,9) for P8.33.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO9
```

### Field Value

Type: Designator

## Remarks

Requires the `BB-GPIO` device tree overlay.

## See Also

### Reference

BeagleBone Class
IO.Objects.SimpleIO.Platforms Namespace

# BeagleBoneI2C2 Field

I2C bus designator for P9.19 and P9.20. Conflicts with `GPIO13` or `GPIO12`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator I2C2
```

### Field Value

Type: Designator

## Remarks

Requires the `BB-GPIO` device tree overlay.

## See Also

### Reference

BeagleBone Class
IO.Objects.SimpleIO.Platforms Namespace

# BeagleBoneSPI0_0 Field

SPI slave designator (0,0) for P9.17. Conflicts with `SPI0 SS0` or `GPIO5`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator SPI0_0
```

### Field Value

Type: Designator

## Remarks

Requires the `BB-GPIO` device tree overlay.

## See Also

### Reference

BeagleBone Class
IO.Objects.SimpleIO.Platforms Namespace

# BeagleBoneSPI1_0 Field

SPI slave designator (1,0) for P9.28. Conflicts with `SP1 SS0` or `GPIO113`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static readonly Designator SPI1_0
```

### Field Value

Type: Designator

## Remarks

Requires the `BB-GPIO` device tree overlay.

## See Also

### Reference

BeagleBone Class
IO.Objects.SimpleIO.Platforms Namespace

# BeagleBoneSPI1_1 Field

SPI slave designator (1,1) for P9.42. Conflicts with `SPI1_SS1` or `GPIO7`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator SPI1_1
```

### Field Value

Type: Designator

## Remarks

Requires the `BB-GPIO` device tree overlay.

## See Also

### Reference

BeagleBone Class
IO.Objects.SimpleIO.Platforms Namespace

# MUNTS_0018 Class

This class provides `Designator`s for I/O resources available on the MUNTS-0018 Tutorial I/O Board.

## Inheritance Hierarchy

SystemObject IO.Objects.SimpleIO.PlatformsMUNTS_0018

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static class MUNTS_0018
```

The MUNTS_0018 type exposes the following members.

## Methods

| | Name | Description |
|---|---|---|
| S | AnalogInputFactory | Analog input object factory for the on-board analog inputs at connectors `J10` and `J11`. |
| S | ButtonInputFactory | GPIO pin object factory for the on-board button switch at `SW1`. |

| | | |
|---|---|---|
| S | GPIOPinFactory | GPIO pin object factory for GPIO pins at connectors `J4` to `J7`. |
| S | I2CBusFactory | $I^2C$ bus object factory for the on-board $I^2C$ buses at connectors `J5` and `J9`. |
| S | LEDOutputFactory | GPIO pin object factory for the on-board LED at `D1`. |
| S | MotorOutputFactory | Motor output object factory for the on-board motor outputs at connectors `J6` and `J7`. |
| S | ServoOutputFactory | Servo output object factory for the on-board servo outputs at headers `J2` and `J3`. |

Top

## Fields

| | Name | Description |
|---|---|---|
| S | D1 | GPIO pin `Designator` for the on-board LED `D1`. |
| S | J10A0 | Analog input `Designator` for Analog Input Grove Connector `J10` pin `A0` (MCP3204 input `CH2`. |
| S | J10A1 | Analog input `Designator` for Analog |

| | | |
|---|---|---|
| | | Input Grove Connector `J10` pin `A1` (MCP3204 input `CH3`. |
| S | J11A0 | Analog input `Designator` for Analog Input Grove Connector `J11` pin `A0` (MCP3204 input `CH0`. |
| S | J11A1 | Analog input `Designator` for Analog Input Grove Connector `J11` pin `A1` (MCP3204 input `CH1`. |
| S | J2PWM | PWM output `Designator` for Servo Header `J2`. |
| S | J3PWM | PWM output `Designator` for Servo Header `J3`. |
| S | J4D0 | GPIO pin `Designator` for Digital I/O Grove Connector `J4` pin `D0`. |
| S | J4D1 | GPIO pin `Designator` for Digital I/O Grove Connector `J4` pin `D1`. |
| S | J5D0 | GPIO pin `Designator` for Digital I/O Grove Connector `J5` pin `D0`. |
| S | J5D1 | GPIO pin `Designator` for Digital I/O Grove Connector `J5` pin `D1`. |
| S | J5I2C | $I^2C$ bus `Designator` for `/dev/i2c3` on Digital I/O Grove Connector `J5`. |
| S | J6D0 | GPIO pin `Designator` for DC Motor Driver Grove Connector `J6` pin `D0`. |
| S | J6D1 | GPIO pin `Designator` for DC Motor |

| | | |
|---|---|---|
| | | Driver Grove Connector `J6` pin `D1`. |
| S | J6DIR | GPIO pin `Designator` for DC Motor Driver Grove Connector `J6` pin `D1`. |
| S | J6PWM | PWM output `Designator` for DC Motor Driver Grove Connector `J6` pin `D0`. |
| S | J7D0 | GPIO pin `Designator` for DC Motor Driver Grove Connector `J7` pin `D0`. |
| S | J7D1 | GPIO pin `Designator` for DC Motor Driver Grove Connector `J7` pin `D1`. |
| S | J7DIR | GPIO pin `Designator` for DC Motor Driver Grove Connector `J7` pin `D1`. |
| S | J7PWM | PWM output `Designator` for DC Motor Driver Grove Connector `J7` pin `D0`. |
| S | J9I2C | $I^2C$ bus `Designator` for `/dev/i2c-1` on $I^2C$ Grove Connector `J9`. |
| S | SW1 | GPIO pin `Designator` for the on-board momentary switch `SW1`. |

Top

## Remarks

The following device tree overlay commands must be added to `/boot/config.txt`:

Copy

```
dtoverlay=anyspi,spi0-1,dev="microchip,mcp3204",speed=
dtoverlay=pwm-2chan,pin=12,func=4,pin2=19,func2=2
```

## See Also

### Reference

IO.Objects.SimpleIO.Platforms Namespace

# MUNTS_0018 Methods

The MUNTS_0018 type exposes the following members.

## Methods

| | Name | Description |
|---|---|---|
| | AnalogInputFactory | Analog input object factory for the on-board analog inputs at connectors `J10` and `J11`. |
| | ButtonInputFactory | GPIO pin object factory for the on-board button switch at `SW1`. |
| | GPIOPinFactory | GPIO pin object factory for GPIO pins at connectors `J4` to `J7`. |
| | I2CBusFactory | $I^2C$ bus object factory for the on-board $I^2C$ buses at connectors `J5` and `J9`. |
| | LEDOutputFactory | GPIO pin object factory for the on-board LED at `D1`. |
| | MotorOutputFactory | Motor output object factory for the on-board motor |

| | | |
|---|---|---|
| | | outputs at connectors `J6` and `J7`. |
| | ServoOutputFactory | Servo output object factory for the on-board servo outputs at headers `J2` and `J3`. |

Top

## See Also

**Reference**

MUNTS_0018 Class

IO.Objects.SimpleIO.Platforms Namespace

# MUNTS_0018AnalogInputFactory Method

Analog input object factory for the on-board analog inputs at connectors `J10` and `J11`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static Voltage AnalogInputFactory(
        Designator desg
)
```

### Parameters

*desg*
Type: IO.Objects.SimpleIO.DeviceDesignator
Device designator for one of the on-board analog inputs (`J10A0`, `J10A1`, `J11A0`, or `J11A1`).

### Return Value

Type: Voltage
Analog input object.

## See Also

### Reference

MUNTS_0018 Class

# IO.Objects.SimpleIO.Platforms Namespace

# MUNTS_0018ButtonInputFactory Method

GPIO pin object factory for the on-board button switch at `SW1`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

**C#** **VB** **F#** Copy

```
public static Pin ButtonInputFactory(
        Edge edge = Edge.None
)
```

### Parameters

*edge* **(Optional)**
Type: IO.Objects.SimpleIO.GPIOEdge
Interrupt edge setting.

### Return Value

Type: Pin
GPIO input pin object.

## See Also

### Reference

MUNTS_0018 Class
IO.Objects.SimpleIO.Platforms Namespace

# MUNTS_0018GPIOPinFactory Method

GPIO pin object factory for GPIO pins at connectors `J4` to `J7`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

**C#** **VB** **F#** Copy

```
public static Pin GPIOPinFactory(
        Designator desg,
        Direction dir,
        bool state = false,
        Driver driver = Driver.PushPull,
        Edge edge = Edge.None,
        Polarity polarity = Polarity.ActiveHigh
)
```

### Parameters

*desg*
Type: IO.Objects.SimpleIO.DeviceDesignator
Device designator for one of the on-board GPIO pins (`J4D0` to `J7D1`.

*dir*
Type: IO.Interfaces.GPIODirection
Data direction.

*state* **(Optional)**

Type: SystemBoolean
Initial GPIO output state.

*driver* **(Optional)**
Type: IO.Objects.SimpleIO.GPIODriver
Output driver setting.

*edge* **(Optional)**
Type: IO.Objects.SimpleIO.GPIOEdge
Interrupt edge setting.

*polarity* **(Optional)**
Type: IO.Objects.SimpleIO.GPIOPolarity
Polarity setting.

### Return Value

Type: Pin
GPIO pin object.

## See Also

### Reference

MUNTS_0018 Class
IO.Objects.SimpleIO.Platforms Namespace

# MUNTS_0018I2CBusFactory Method

I²C bus object factory for the on-board I²C buses at connectors `J5` and `J9`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static Bus I2CBusFactory(
        Designator desg
)
```

### Parameters

*desg*
Type: IO.Objects.SimpleIO.DeviceDesignator
Device designator for one of the on-board I²C buses (`J5I2C` or `J9I2C`).

### Return Value

Type: Bus
I²C bus object.

## See Also

### Reference

MUNTS_0018 Class

# IO.Objects.SimpleIO.Platforms Namespace

# MUNTS_0018LEDOutputFactory Method

GPIO pin object factory for the on-board LED at `D1`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static Pin LEDOutputFactory(
        bool state = false
)
```

### Parameters

*state* **(Optional)**
Type: SystemBoolean
Initial GPIO output state.

### Return Value

Type: Pin
GPIO output pin object.

## See Also

### Reference

MUNTS_0018 Class
IO.Objects.SimpleIO.Platforms Namespace

# MUNTS_0018MotorOutputFactory Method

Motor output object factory for the on-board motor outputs at connectors `J6` and `J7`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static Output MotorOutputFactory(
        Designator speed,
        Designator direction,
        int frequency,
        double velocity = 0
)
```

### Parameters

*speed*
Type: IO.Objects.SimpleIO.DeviceDesignator
Device designator for the motor speed PWM output (`J6PWM` or `J7PWM`).

*direction*
Type: IO.Objects.SimpleIO.DeviceDesignator
Device designator for the motor direction GPIO output (`J6DIR` or `J7DIR`).

*frequency*

Type: SystemInt32
PWM pulse frequency.

*velocity* **(Optional)**
Type: SystemDouble
Initial motor velocity.

### Return Value

Type: Output
Motor output object.

## See Also

### Reference

MUNTS_0018 Class
IO.Objects.SimpleIO.Platforms Namespace

# MUNTS_0018ServoOutputFactory Method

Servo output object factory for the on-board servo outputs at headers `J2` and `J3`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static Output ServoOutputFactory(
        Designator desg,
        int frequency = 50
)
```

### Parameters

*desg*
Type: IO.Objects.SimpleIO.DeviceDesignator
Device designator for one of the on-board PWM outputs (`J2PWM` or `J3PWM`).

*frequency* **(Optional)**
Type: SystemInt32
PWM pulse frequency. Ordinary analog servos operate best at 50 Hz.

### Return Value

Type: Output
Servo output object.

## See Also

### Reference

MUNTS_0018 Class

IO.Objects.SimpleIO.Platforms Namespace

# MUNTS_0018 Fields

The MUNTS_0018 type exposes the following members.

## Fields

| | Name | Description |
|---|---|---|
| S | D1 | GPIO pin `Designator` for the on-board LED `D1`. |
| S | J10A0 | Analog input `Designator` for Analog Input Grove Connector `J10` pin `A0` (MCP3204 input `CH2`. |
| S | J10A1 | Analog input `Designator` for Analog Input Grove Connector `J10` pin `A1` (MCP3204 input `CH3`. |
| S | J11A0 | Analog input `Designator` for Analog Input Grove Connector `J11` pin `A0` (MCP3204 input `CH0`. |
| S | J11A1 | Analog input `Designator` for Analog Input Grove Connector `J11` pin `A1` (MCP3204 input `CH1`. |
| S | J2PWM | PWM output `Designator` for Servo Header `J2`. |
| S | J3PWM | PWM output `Designator` for Servo Header `J3`. |

| | | |
|---|---|---|
| S | J4D0 | GPIO pin `Designator` for Digital I/O Grove Connector `J4` pin `D0`. |
| S | J4D1 | GPIO pin `Designator` for Digital I/O Grove Connector `J4` pin `D1`. |
| S | J5D0 | GPIO pin `Designator` for Digital I/O Grove Connector `J5` pin `D0`. |
| S | J5D1 | GPIO pin `Designator` for Digital I/O Grove Connector `J5` pin `D1`. |
| S | J5I2C | $I^2C$ bus `Designator` for `/dev/i2c3` on Digital I/O Grove Connector `J5`. |
| S | J6D0 | GPIO pin `Designator` for DC Motor Driver Grove Connector `J6` pin `D0`. |
| S | J6D1 | GPIO pin `Designator` for DC Motor Driver Grove Connector `J6` pin `D1`. |
| S | J6DIR | GPIO pin `Designator` for DC Motor Driver Grove Connector `J6` pin `D1`. |
| S | J6PWM | PWM output `Designator` for DC Motor Driver Grove Connector `J6` pin `D0`. |
| S | J7D0 | GPIO pin `Designator` for DC Motor Driver Grove Connector `J7` pin `D0`. |
| S | J7D1 | GPIO pin `Designator` for DC Motor Driver Grove Connector `J7` pin `D1`. |
| S | J7DIR | GPIO pin `Designator` for DC Motor Driver Grove Connector `J7` pin `D1`. |

| | | |
|---|---|---|
| | J7PWM | PWM output `Designator` for DC Motor Driver Grove Connector `J7` pin `D0`. |
| | J9I2C | $I^2C$ bus `Designator` for `/dev/i2c-1` on $I^2C$ Grove Connector `J9`. |
| | SW1 | GPIO pin `Designator` for the on-board momentary switch `SW1`. |

Top

## See Also

### Reference

MUNTS_0018 Class

IO.Objects.SimpleIO.Platforms Namespace

# MUNTS_0018D1 Field

GPIO pin `Designator` for the on-board LED `D1`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator D1
```

### Field Value

Type: Designator

## See Also

### Reference

MUNTS_0018 Class
IO.Objects.SimpleIO.Platforms Namespace

# MUNTS_0018J10A0 Field

Analog input `Designator` for Analog Input Grove Connector `J10` pin `A0` (MCP3204 input `CH2`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator J10A0
```

### Field Value

Type: Designator

## See Also

### Reference

MUNTS_0018 Class
IO.Objects.SimpleIO.Platforms Namespace

# MUNTS_0018J10A1 Field

Analog input `Designator` for Analog Input Grove Connector `J10` pin `A1` (MCP3204 input `CH3`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator J10A1
```

### Field Value

Type: Designator

## See Also

### Reference

MUNTS_0018 Class
IO.Objects.SimpleIO.Platforms Namespace

# MUNTS_0018J11A0 Field

Analog input `Designator` for Analog Input Grove Connector `J11` pin `A0` (MCP3204 input `CH0`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator J11A0
```

### Field Value

Type: Designator

## See Also

### Reference

MUNTS_0018 Class
IO.Objects.SimpleIO.Platforms Namespace

# MUNTS_0018J11A1 Field

Analog input `Designator` for Analog Input Grove Connector `J11` pin `A1` (MCP3204 input `CH1`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator J11A1
```

### Field Value

Type: Designator

## See Also

### Reference

MUNTS_0018 Class
IO.Objects.SimpleIO.Platforms Namespace

# MUNTS_0018J2PWM Field

PWM output `Designator` for Servo Header `J2`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator J2PWM
```

### Field Value

Type: Designator

## See Also

### Reference

MUNTS_0018 Class
IO.Objects.SimpleIO.Platforms Namespace

# MUNTS_0018J3PWM Field

PWM output `Designator` for Servo Header `J3`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator J3PWM
```

### Field Value

Type: Designator

## See Also

### Reference

MUNTS_0018 Class
IO.Objects.SimpleIO.Platforms Namespace

# MUNTS_0018J4D0 Field

GPIO pin `Designator` for Digital I/O Grove Connector `J4` pin `D0`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator J4D0
```

### Field Value

Type: Designator

## See Also

### Reference

MUNTS_0018 Class
IO.Objects.SimpleIO.Platforms Namespace

# MUNTS_0018J4D1 Field

GPIO pin `Designator` for Digital I/O Grove Connector `J4` pin `D1`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator J4D1
```

### Field Value

Type: Designator

## See Also

### Reference

MUNTS_0018 Class
IO.Objects.SimpleIO.Platforms Namespace

# MUNTS_0018J5D0 Field

GPIO pin `Designator` for Digital I/O Grove Connector `J5` pin `D0`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator J5D0
```

### Field Value

Type: Designator

## See Also

### Reference

MUNTS_0018 Class
IO.Objects.SimpleIO.Platforms Namespace

# MUNTS_0018J5D1 Field

GPIO pin `Designator` for Digital I/O Grove Connector `J5` pin `D1`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator J5D1
```

### Field Value

Type: Designator

## See Also

### Reference

MUNTS_0018 Class
IO.Objects.SimpleIO.Platforms Namespace

# MUNTS_0018J5I2C Field

I²C bus `Designator` for `/dev/i2c3` on Digital I/O Grove Connector `J5`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator J5I2C
```

### Field Value

Type: Designator

## Remarks

This optional I²C bus is only available on a Raspberry Pi 4, and only if the following is added to `/boot/config.txt`:

Copy

```
dtoverlay=i2c3
```

You may also need to install I²C bus 4.7K or 10K pull-up resistors on the MUNTS-0018 Tutorial I/O Board, at positions `R5` and `R6`.

## See Also

### Reference

MUNTS_0018 Class
IO.Objects.SimpleIO.Platforms Namespace

# MUNTS_0018J6D0 Field

GPIO pin `Designator` for DC Motor Driver Grove Connector `J6` pin `D0`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator J6D0
```

### Field Value

Type: Designator

## Remarks

This GPIO pin is normally unusable because `GPIO12` is mapped to PWM output `PWM0`.

## See Also

### Reference

MUNTS_0018 Class
IO.Objects.SimpleIO.Platforms Namespace

# MUNTS_0018J6D1 Field

GPIO pin `Designator` for DC Motor Driver Grove Connector `J6` pin `D1`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator J6D1
```

### Field Value

Type: Designator

## See Also

### Reference

MUNTS_0018 Class
IO.Objects.SimpleIO.Platforms Namespace

# MUNTS_0018J6DIR Field

GPIO pin `Designator` for DC Motor Driver Grove Connector `J6` pin `D1`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static readonly Designator J6DIR
```

### Field Value

Type: Designator

## Remarks

`J6DIR` controls the motor **direction**.

## See Also

### Reference

MUNTS_0018 Class
IO.Objects.SimpleIO.Platforms Namespace

# MUNTS_0018J6PWM Field

PWM output `Designator` for DC Motor Driver Grove Connector `J6` pin `D0`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static readonly Designator J6PWM
```

### Field Value

Type: Designator

## Remarks

`J6PWM` controls the motor **speed**.

## See Also

### Reference

MUNTS_0018 Class
IO.Objects.SimpleIO.Platforms Namespace

# MUNTS_0018J7D0 Field

GPIO pin `Designator` for DC Motor Driver Grove Connector `J7` pin `D0`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator J7D0
```

### Field Value

Type: Designator

## Remarks

This GPIO pin is normally unusable because `GPIO19` is mapped to PWM output `PWM1`.

## See Also

### Reference

MUNTS_0018 Class
IO.Objects.SimpleIO.Platforms Namespace

# MUNTS_0018J7D1 Field

GPIO pin `Designator` for DC Motor Driver Grove Connector `J7` pin `D1`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator J7D1
```

### Field Value

Type: Designator

## See Also

### Reference

MUNTS_0018 Class
IO.Objects.SimpleIO.Platforms Namespace

# MUNTS_0018J7DIR Field

GPIO pin `Designator` for DC Motor Driver Grove Connector `J7` pin `D1`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator J7DIR
```

### Field Value

Type: Designator

## Remarks

`J7DIR` controls the motor **direction**.

## See Also

### Reference

MUNTS_0018 Class
IO.Objects.SimpleIO.Platforms Namespace

# MUNTS_0018J7PWM Field

PWM output `Designator` for DC Motor Driver Grove Connector `J7` pin `D0`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static readonly Designator J7PWM
```

### Field Value

Type: Designator

## Remarks

`J7PWM` controls the motor **speed**.

## See Also

### Reference

MUNTS_0018 Class
IO.Objects.SimpleIO.Platforms Namespace

# MUNTS_0018J9I2C Field

I²C bus `Designator` for `/dev/i2c-1` on I²C Grove Connector `J9`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator J9I2C
```

### Field Value

Type: Designator

## See Also

### Reference

MUNTS_0018 Class
IO.Objects.SimpleIO.Platforms Namespace

# MUNTS_0018SW1 Field

GPIO pin `Designator` for the on-board momentary switch `SW1`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator SW1
```

### Field Value

Type: Designator

## See Also

### Reference

MUNTS_0018 Class
IO.Objects.SimpleIO.Platforms Namespace

# PocketBeagle Class

This class defines identifiers for the devices provided by the PocketBeagle hardware platform. .

## Inheritance Hierarchy

SystemObject IO.Objects.SimpleIO.PlatformsPocketBeagle

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static class PocketBeagle
```

The PocketBeagle type exposes the following members.

## Fields

| | Name | Description |
|---|---|---|
| | AIN0 | ADC input designator (0,0) for P1.19 (1.8V). |
| | AIN1 | ADC input designator (0,1) for P1.21 (1.8V). |
| | AIN2 | ADC input designator (0,2) for P1.23 (1.8V). |

| | | |
|---|---|---|
| S | AIN3 | ADC input designator (0,3) for P1.25 (1.8V). |
| S | AIN4 | ADC input designator (0,4) for P1.27 (1.8V). |
| S | AIN5 | ADC input designator (0,5) for P2.35 (3.6V). Conflicts with `GPIO86`. |
| S | AIN6 | ADC input designator (0,6) for P1.2 (3.6V). Conflicts with `GPIO87`. |
| S | AIN7 | ADC input designator (0,7) for P2.36 (1.8V). |
| S | GPIO110 | GPIO pin designator (3,14) for P1.36. |
| S | GPIO111 | GPIO pin designator (3,15) for P1.33. |
| S | GPIO112 | GPIO pin designator (3,16) for P2.32. |
| S | GPIO113 | GPIO pin designator (3,17) for P2.30. |
| S | GPIO114 | GPIO pin designator (3,18) for P1.31. |
| S | GPIO115 | GPIO pin designator (3,19) for P2.34. |
| S | GPIO116 | GPIO pin designator (3,20) for P2.28. |
| S | GPIO117 | GPIO pin designator (3,21) for P1.29. |
| S | GPIO12 | GPIO pin designator (0,12) for P1.26. Conflicts with `I2C2 SDA`. |
| S | GPIO13 | GPIO pin designator (0,13) for P1.28. Conflicts with `I2C2 SCL`. |

| | | |
|---|---|---|
| s | GPIO14 | GPIO pin designator (0,14) for P2.11. Conflicts with `I2C1 SDA`. |
| s | GPIO15 | GPIO pin designator (0,15) for P2.9. Conflicts with `I2C1 SCL`. |
| s | GPIO19 | GPIO pin designator (0,19) for P2.31. Conflicts with `SPI1 CS1`. |
| s | GPIO2 | GPIO pin designator (0,2) for P1.8. Conflicts with `SPI0 SCLK`. |
| s | GPIO20 | GPIO pin designator (0,20) for P1.20. |
| s | GPIO23 | GPIO pin designator (0,23) for P2.3. |
| s | GPIO26 | GPIO pin designator (0,26) for P1.34. |
| s | GPIO27 | GPIO pin designator (0,27) for P2.19. |
| s | GPIO3 | GPIO pin designator (0,3) for P1.10. Conflicts with `SPI0 MISO`. |
| s | GPIO30 | GPIO pin designator (0,30) for P2.5. Conflicts with `RXD4`. |
| s | GPIO31 | GPIO pin designator (0,31) for P2.7. Conflicts with `TXD4`. |
| s | GPIO4 | GPIO pin designator (0,4) for P1.12. Conflicts with `SPI0 MOSI`. |
| s | GPIO40 | GPIO pin designator (1,8) for P2.27. Conflicts with `SPI1 MISO`. |
| s | GPIO41 | GPIO pin designator (1,9) for P2.25. |

| | | |
|---|---|---|
| | | Conflicts with `SPI1_MOSI`. |
| S | GPIO42 | GPIO pin designator (1,10) for P1.32. Conflicts with `RXD0`. |
| S | GPIO43 | GPIO pin designator (1,11) for P1.30. Conflicts with `TXD0`. |
| S | GPIO44 | GPIO pin designator (1,12) for P2.24. |
| S | GPIO45 | GPIO pin designator (1,13) for P2.33. |
| S | GPIO46 | GPIO pin designator (1,14) for P2.22. |
| S | GPIO47 | GPIO pin designator (1,15) for P2.18. |
| S | GPIO5 | GPIO pin designator (0,5) for P1.6. Conflicts with `SPI0_CS0`. |
| S | GPIO50 | GPIO pin designator (1,18) for P2.1. |
| S | GPIO52 | GPIO pin designator (1,20) for P2.10. |
| S | GPIO57 | GPIO pin designator (1,25) for P2.6. |
| S | GPIO58 | GPIO pin designator (1,26) for P2.4. |
| S | GPIO59 | GPIO pin designator (1,27) for P2.2. |
| S | GPIO60 | GPIO pin designator (1,28) for P2.8. |
| S | GPIO64 | GPIO pin designator (2,0) for P2.20. |
| S | GPIO65 | GPIO pin designator (2,1) for P2.17. |
| S | GPIO7 | GPIO pin designator (0,7) for P2.29. Conflicts with `SPI1_SCLK`. |

| | | |
|---|---|---|
| S | GPIO86 | GPIO pin designator (2,22) for P2.35. Conflicts with `AIN5`. |
| S | GPIO87 | GPIO pin designator (2,23) for P1.2. Conflicts with `AIN6`. |
| S | GPIO88 | GPIO pin designator (2,24) for P1.35. |
| S | GPIO89 | GPIO pin designator (2,25) for P1.4. |
| S | I2C1 | I2C bus designator (0,1) for P2.9 and P2.11. Conflicts with `GPIO15` and `GPIO14`. |
| S | I2C2 | I2C bus designator (0,2) for P1.26 and P1.28. Conflicts with `GPIO12` and `GPIO13`. |
| S | PWM0_0 | PWM output designator (0,0) for P1.36. Conflicts with `GPIO110`. |
| S | PWM2_0 | PWM output designator (2,0) for P2.1. Conflicts with `GPIO50`. |
| S | SPI0_0 | SPI slave designator (0,0) for P1.6. Conflicts with `GPIO5`. |
| S | SPI1_1 | SPI slave designator (1,1) for P2.31. Conflicts with `GPIO19`. |

Top

# See Also

## Reference

# IO.Objects.SimpleIO.Platforms Namespace

# PocketBeagle Fields

The PocketBeagle type exposes the following members.

## Fields

| | Name | Description |
|---|---|---|
| s | AIN0 | ADC input designator (0,0) for P1.19 (1.8V). |
| s | AIN1 | ADC input designator (0,1) for P1.21 (1.8V). |
| s | AIN2 | ADC input designator (0,2) for P1.23 (1.8V). |
| s | AIN3 | ADC input designator (0,3) for P1.25 (1.8V). |
| s | AIN4 | ADC input designator (0,4) for P1.27 (1.8V). |
| s | AIN5 | ADC input designator (0,5) for P2.35 (3.6V). Conflicts with `GPIO86`. |
| s | AIN6 | ADC input designator (0,6) for P1.2 (3.6V). Conflicts with `GPIO87`. |
| s | AIN7 | ADC input designator (0,7) for P2.36 (1.8V). |
| s | | |

| | | |
|---|---|---|
| | GPIO110 | GPIO pin designator (3,14) for P1.36. |
| S | GPIO111 | GPIO pin designator (3,15) for P1.33. |
| S | GPIO112 | GPIO pin designator (3,16) for P2.32. |
| S | GPIO113 | GPIO pin designator (3,17) for P2.30. |
| S | GPIO114 | GPIO pin designator (3,18) for P1.31. |
| S | GPIO115 | GPIO pin designator (3,19) for P2.34. |
| S | GPIO116 | GPIO pin designator (3,20) for P2.28. |
| S | GPIO117 | GPIO pin designator (3,21) for P1.29. |
| S | GPIO12 | GPIO pin designator (0,12) for P1.26. Conflicts with `I2C2 SDA`. |
| S | GPIO13 | GPIO pin designator (0,13) for P1.28. Conflicts with `I2C2 SCL`. |
| S | GPIO14 | GPIO pin designator (0,14) for P2.11. Conflicts with `I2C1 SDA`. |
| S | GPIO15 | GPIO pin designator (0,15) for P2.9. Conflicts with `I2C1 SCL`. |
| S | GPIO19 | GPIO pin designator (0,19) for P2.31. Conflicts with `SPI1 CS1`. |
| S | GPIO2 | GPIO pin designator (0,2) for P1.8. Conflicts with `SPI0 SCLK`. |
| S | GPIO20 | GPIO pin designator (0,20) for P1.20. |
| S | GPIO23 | GPIO pin designator (0,23) for P2.3. |

| | | |
|---|---|---|
| S | GPIO26 | GPIO pin designator (0,26) for P1.34. |
| S | GPIO27 | GPIO pin designator (0,27) for P2.19. |
| S | GPIO3 | GPIO pin designator (0,3) for P1.10. Conflicts with `SPI0 MISO`. |
| S | GPIO30 | GPIO pin designator (0,30) for P2.5. Conflicts with `RXD4`. |
| S | GPIO31 | GPIO pin designator (0,31) for P2.7. Conflicts with `TXD4`. |
| S | GPIO4 | GPIO pin designator (0,4) for P1.12. Conflicts with `SPI0 MOSI`. |
| S | GPIO40 | GPIO pin designator (1,8) for P2.27. Conflicts with `SPI1 MISO`. |
| S | GPIO41 | GPIO pin designator (1,9) for P2.25. Conflicts with `SPI1 MOSI`. |
| S | GPIO42 | GPIO pin designator (1,10) for P1.32. Conflicts with `RXD0`. |
| S | GPIO43 | GPIO pin designator (1,11) for P1.30. Conflicts with `TXD0`. |
| S | GPIO44 | GPIO pin designator (1,12) for P2.24. |
| S | GPIO45 | GPIO pin designator (1,13) for P2.33. |
| S | GPIO46 | GPIO pin designator (1,14) for P2.22. |
| S | GPIO47 | GPIO pin designator (1,15) for P2.18. |

| | | |
|---|---|---|
| S | GPIO5 | GPIO pin designator (0,5) for P1.6. Conflicts with `SPI0 CS0`. |
| S | GPIO50 | GPIO pin designator (1,18) for P2.1. |
| S | GPIO52 | GPIO pin designator (1,20) for P2.10. |
| S | GPIO57 | GPIO pin designator (1,25) for P2.6. |
| S | GPIO58 | GPIO pin designator (1,26) for P2.4. |
| S | GPIO59 | GPIO pin designator (1,27) for P2.2. |
| S | GPIO60 | GPIO pin designator (1,28) for P2.8. |
| S | GPIO64 | GPIO pin designator (2,0) for P2.20. |
| S | GPIO65 | GPIO pin designator (2,1) for P2.17. |
| S | GPIO7 | GPIO pin designator (0,7) for P2.29. Conflicts with `SPI1 SCLK`. |
| S | GPIO86 | GPIO pin designator (2,22) for P2.35. Conflicts with `AIN5`. |
| S | GPIO87 | GPIO pin designator (2,23) for P1.2. Conflicts with `AIN6`. |
| S | GPIO88 | GPIO pin designator (2,24) for P1.35. |
| S | GPIO89 | GPIO pin designator (2,25) for P1.4. |
| S | I2C1 | I2C bus designator (0,1) for P2.9 and P2.11. Conflicts with `GPIO15` and `GPIO14`. |
| S | I2C2 | I2C bus designator (0,2) for P1.26 and |

| | | |
|---|---|---|
| | | P1.28. Conflicts with `GPIO12` and `GPIO13`. |
| | PWM0_0 | PWM output designator (0,0) for P1.36. Conflicts with `GPIO110`. |
| | PWM2_0 | PWM output designator (2,0) for P2.1. Conflicts with `GPIO50`. |
| | SPI0_0 | SPI slave designator (0,0) for P1.6. Conflicts with `GPIO5`. |
| | SPI1_1 | SPI slave designator (1,1) for P2.31. Conflicts with `GPIO19`. |

Top

## See Also

### Reference

PocketBeagle Class

IO.Objects.SimpleIO.Platforms Namespace

# PocketBeagleAIN0 Field

ADC input designator (0,0) for P1.19 (1.8V).

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static readonly Designator AIN0
```

### Field Value

Type: Designator

## Remarks

Requires the `PB-ADC` device tree overlay.

## See Also

### Reference

PocketBeagle Class
IO.Objects.SimpleIO.Platforms Namespace

# PocketBeagleAIN1 Field

ADC input designator (0,1) for P1.21 (1.8V).

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator AIN1
```

### Field Value

Type: Designator

## Remarks

Requires the `PB-ADC` device tree overlay.

## See Also

### Reference

PocketBeagle Class
IO.Objects.SimpleIO.Platforms Namespace

# PocketBeagleAIN2 Field

ADC input designator (0,2) for P1.23 (1.8V).

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator AIN2
```

### Field Value

Type: Designator

## Remarks

Requires the `PB-ADC` device tree overlay.

## See Also

### Reference

PocketBeagle Class
IO.Objects.SimpleIO.Platforms Namespace

# PocketBeagleAIN3 Field

ADC input designator (0,3) for P1.25 (1.8V).

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator AIN3
```

### Field Value

Type: Designator

## Remarks

Requires the `PB-ADC` device tree overlay.

## See Also

### Reference

PocketBeagle Class
IO.Objects.SimpleIO.Platforms Namespace

# PocketBeagleAIN4 Field

ADC input designator (0,4) for P1.27 (1.8V).

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator AIN4
```

### Field Value

Type: Designator

## Remarks

Requires the `PB-ADC` device tree overlay.

## See Also

### Reference

PocketBeagle Class
IO.Objects.SimpleIO.Platforms Namespace

# PocketBeagleAIN5 Field

ADC input designator (0,5) for P2.35 (3.6V). Conflicts with `GPIO86`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator AIN5
```

### Field Value

Type: Designator

## Remarks

Requires the `PB-ADC` device tree overlay.

## See Also

### Reference

PocketBeagle Class
IO.Objects.SimpleIO.Platforms Namespace

# PocketBeagleAIN6 Field

ADC input designator (0,6) for P1.2 (3.6V). Conflicts with `GPIO87`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator AIN6
```

### Field Value

Type: Designator

## Remarks

Requires the `PB-ADC` device tree overlay.

## See Also

### Reference

PocketBeagle Class
IO.Objects.SimpleIO.Platforms Namespace

# PocketBeagleAIN7 Field

ADC input designator (0,7) for P2.36 (1.8V).

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator AIN7
```

### Field Value

Type: Designator

## Remarks

Requires the `PB-ADC` device tree overlay.

## See Also

### Reference

PocketBeagle Class
IO.Objects.SimpleIO.Platforms Namespace

# PocketBeagleGPIO110 Field

GPIO pin designator (3,14) for P1.36.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO110
```

### Field Value

Type: Designator

## See Also

### Reference

PocketBeagle Class
IO.Objects.SimpleIO.Platforms Namespace

# PocketBeagleGPIO111 Field

GPIO pin designator (3,15) for P1.33.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO111
```

### Field Value

Type: Designator

## See Also

### Reference

PocketBeagle Class
IO.Objects.SimpleIO.Platforms Namespace

# PocketBeagleGPIO112 Field

GPIO pin designator (3,16) for P2.32.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO112
```

### Field Value

Type: Designator

## See Also

### Reference

PocketBeagle Class
IO.Objects.SimpleIO.Platforms Namespace

# PocketBeagleGPIO113 Field

GPIO pin designator (3,17) for P2.30.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO113
```

### Field Value

Type: Designator

## See Also

### Reference

PocketBeagle Class
IO.Objects.SimpleIO.Platforms Namespace

# PocketBeagleGPIO114 Field

GPIO pin designator (3,18) for P1.31.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO114
```

### Field Value

Type: Designator

## See Also

### Reference

PocketBeagle Class
IO.Objects.SimpleIO.Platforms Namespace

# PocketBeagleGPIO115 Field

GPIO pin designator (3,19) for P2.34.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO115
```

### Field Value

Type: Designator

## See Also

### Reference

PocketBeagle Class
IO.Objects.SimpleIO.Platforms Namespace

# PocketBeagleGPIO116 Field

GPIO pin designator (3,20) for P2.28.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static readonly Designator GPIO116
```

### Field Value

Type: Designator

## See Also

### Reference

PocketBeagle Class
IO.Objects.SimpleIO.Platforms Namespace

# PocketBeagleGPIO117 Field

GPIO pin designator (3,21) for P1.29.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static readonly Designator GPIO117
```

### Field Value

Type: Designator

## See Also

### Reference

PocketBeagle Class
IO.Objects.SimpleIO.Platforms Namespace

# PocketBeagleGPIO12 Field

GPIO pin designator (0,12) for P1.26. Conflicts with `I2C2 SDA`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO12
```

### Field Value

Type: Designator

## See Also

### Reference

PocketBeagle Class
IO.Objects.SimpleIO.Platforms Namespace

# PocketBeagleGPIO13 Field

GPIO pin designator (0,13) for P1.28. Conflicts with `I2C2 SCL`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO13
```

### Field Value

Type: Designator

## See Also

### Reference

PocketBeagle Class
IO.Objects.SimpleIO.Platforms Namespace

# PocketBeagleGPIO14 Field

GPIO pin designator (0,14) for P2.11. Conflicts with `I2C1 SDA`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO14
```

### Field Value

Type: Designator

## See Also

### Reference

PocketBeagle Class
IO.Objects.SimpleIO.Platforms Namespace

# PocketBeagleGPIO15 Field

GPIO pin designator (0,15) for P2.9. Conflicts with `I2C1 SCL`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static readonly Designator GPIO15
```

### Field Value

Type: Designator

## See Also

### Reference

PocketBeagle Class
IO.Objects.SimpleIO.Platforms Namespace

# PocketBeagleGPIO19 Field

GPIO pin designator (0,19) for P2.31. Conflicts with `SPI1 CS1`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO19
```

### Field Value

Type: Designator

## See Also

### Reference

PocketBeagle Class
IO.Objects.SimpleIO.Platforms Namespace

# PocketBeagleGPIO2 Field

GPIO pin designator (0,2) for P1.8. Conflicts with `SPI0 SCLK`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO2
```

### Field Value

Type: Designator

## See Also

### Reference

PocketBeagle Class
IO.Objects.SimpleIO.Platforms Namespace

# PocketBeagleGPIO20 Field

GPIO pin designator (0,20) for P1.20.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO20
```

### Field Value

Type: Designator

## See Also

### Reference

PocketBeagle Class
IO.Objects.SimpleIO.Platforms Namespace

# PocketBeagleGPIO23 Field

GPIO pin designator (0,23) for P2.3.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO23
```

### Field Value

Type: Designator

## See Also

### Reference

PocketBeagle Class
IO.Objects.SimpleIO.Platforms Namespace

# PocketBeagleGPIO26 Field

GPIO pin designator (0,26) for P1.34.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO26
```

### Field Value

Type: Designator

## See Also

### Reference

PocketBeagle Class
IO.Objects.SimpleIO.Platforms Namespace

# PocketBeagleGPIO27 Field

GPIO pin designator (0,27) for P2.19.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO27
```

### Field Value

Type: Designator

## See Also

### Reference

PocketBeagle Class
IO.Objects.SimpleIO.Platforms Namespace

# PocketBeagleGPIO3 Field

GPIO pin designator (0,3) for P1.10. Conflicts with `SPI0 MISO`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO3
```

### Field Value

Type: Designator

## See Also

### Reference

PocketBeagle Class
IO.Objects.SimpleIO.Platforms Namespace

# PocketBeagleGPIO30 Field

GPIO pin designator (0,30) for P2.5. Conflicts with `RXD4`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO30
```

### Field Value

Type: Designator

## See Also

### Reference

PocketBeagle Class
IO.Objects.SimpleIO.Platforms Namespace

# PocketBeagleGPIO31 Field

GPIO pin designator (0,31) for P2.7. Conflicts with `TXD4`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO31
```

### Field Value

Type: Designator

## See Also

### Reference

PocketBeagle Class
IO.Objects.SimpleIO.Platforms Namespace

# PocketBeagleGPIO4 Field

GPIO pin designator (0,4) for P1.12. Conflicts with `SPI0 MOSI`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO4
```

### Field Value

Type: Designator

## See Also

### Reference

PocketBeagle Class
IO.Objects.SimpleIO.Platforms Namespace

# PocketBeagleGPIO40 Field

GPIO pin designator (1,8) for P2.27. Conflicts with `SPI1 MISO`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO40
```

### Field Value

Type: Designator

## See Also

### Reference

PocketBeagle Class
IO.Objects.SimpleIO.Platforms Namespace

# PocketBeagleGPIO41 Field

GPIO pin designator (1,9) for P2.25. Conflicts with `SPI1 MOSI`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO41
```

### Field Value

Type: Designator

## See Also

### Reference

PocketBeagle Class
IO.Objects.SimpleIO.Platforms Namespace

# PocketBeagleGPIO42 Field

GPIO pin designator (1,10) for P1.32. Conflicts with `RXD0`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO42
```

### Field Value

Type: Designator

## See Also

### Reference

PocketBeagle Class
IO.Objects.SimpleIO.Platforms Namespace

# PocketBeagleGPIO43 Field

GPIO pin designator (1,11) for P1.30. Conflicts with `TXD0`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO43
```

### Field Value

Type: Designator

## See Also

### Reference

PocketBeagle Class
IO.Objects.SimpleIO.Platforms Namespace

# PocketBeagleGPIO44 Field

GPIO pin designator (1,12) for P2.24.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO44
```

### Field Value

Type: Designator

## See Also

### Reference

PocketBeagle Class
IO.Objects.SimpleIO.Platforms Namespace

# PocketBeagleGPIO45 Field

GPIO pin designator (1,13) for P2.33.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO45
```

### Field Value

Type: Designator

## See Also

### Reference

PocketBeagle Class
IO.Objects.SimpleIO.Platforms Namespace

# PocketBeagleGPIO46 Field

GPIO pin designator (1,14) for P2.22.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO46
```

### Field Value

Type: Designator

## See Also

### Reference

PocketBeagle Class
IO.Objects.SimpleIO.Platforms Namespace

# PocketBeagleGPIO47 Field

GPIO pin designator (1,15) for P2.18.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO47
```

### Field Value

Type: Designator

## See Also

### Reference

PocketBeagle Class
IO.Objects.SimpleIO.Platforms Namespace

# PocketBeagleGPIO5 Field

GPIO pin designator (0,5) for P1.6. Conflicts with `SPI0 CS0`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO5
```

### Field Value

Type: Designator

## See Also

### Reference

PocketBeagle Class
IO.Objects.SimpleIO.Platforms Namespace

# PocketBeagleGPIO50 Field

GPIO pin designator (1,18) for P2.1.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO50
```

### Field Value

Type: Designator

## See Also

### Reference

PocketBeagle Class
IO.Objects.SimpleIO.Platforms Namespace

# PocketBeagleGPIO52 Field

GPIO pin designator (1,20) for P2.10.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO52
```

### Field Value

Type: Designator

## See Also

### Reference

PocketBeagle Class
IO.Objects.SimpleIO.Platforms Namespace

# PocketBeagleGPIO57 Field

GPIO pin designator (1,25) for P2.6.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO57
```

### Field Value

Type: Designator

## See Also

### Reference

PocketBeagle Class
IO.Objects.SimpleIO.Platforms Namespace

# PocketBeagleGPIO58 Field

GPIO pin designator (1,26) for P2.4.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO58
```

### Field Value

Type: Designator

## See Also

### Reference

PocketBeagle Class
IO.Objects.SimpleIO.Platforms Namespace

# PocketBeagleGPIO59 Field

GPIO pin designator (1,27) for P2.2.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO59
```

### Field Value

Type: Designator

## See Also

### Reference

PocketBeagle Class
IO.Objects.SimpleIO.Platforms Namespace

# PocketBeagleGPIO60 Field

GPIO pin designator (1,28) for P2.8.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static readonly Designator GPIO60
```

### Field Value

Type: Designator

## See Also

### Reference

PocketBeagle Class
IO.Objects.SimpleIO.Platforms Namespace

# PocketBeagleGPIO64 Field

GPIO pin designator (2,0) for P2.20.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO64
```

### Field Value

Type: Designator

## See Also

### Reference

PocketBeagle Class
IO.Objects.SimpleIO.Platforms Namespace

# PocketBeagleGPIO65 Field

GPIO pin designator (2,1) for P2.17.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO65
```

### Field Value

Type: Designator

## See Also

### Reference

PocketBeagle Class
IO.Objects.SimpleIO.Platforms Namespace

# PocketBeagleGPIO7 Field

GPIO pin designator (0,7) for P2.29. Conflicts with `SPI1 SCLK`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO7
```

### Field Value

Type: Designator

## See Also

### Reference

PocketBeagle Class
IO.Objects.SimpleIO.Platforms Namespace

# PocketBeagleGPIO86 Field

GPIO pin designator (2,22) for P2.35. Conflicts with `AIN5`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO86
```

### Field Value

Type: Designator

## See Also

### Reference

PocketBeagle Class
IO.Objects.SimpleIO.Platforms Namespace

# PocketBeagleGPIO87 Field

GPIO pin designator (2,23) for P1.2. Conflicts with `AIN6`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO87
```

### Field Value

Type: Designator

## See Also

### Reference

PocketBeagle Class
IO.Objects.SimpleIO.Platforms Namespace

# PocketBeagleGPIO88 Field

GPIO pin designator (2,24) for P1.35.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO88
```

### Field Value

Type: Designator

## See Also

### Reference

PocketBeagle Class
IO.Objects.SimpleIO.Platforms Namespace

# PocketBeagleGPIO89 Field

GPIO pin designator (2,25) for P1.4.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO89
```

### Field Value

Type: Designator

## See Also

### Reference

PocketBeagle Class
IO.Objects.SimpleIO.Platforms Namespace

# PocketBeagleI2C1 Field

I2C bus designator (0,1) for P2.9 and P2.11. Conflicts with `GPIO15` and `GPIO14`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator I2C1
```

### Field Value

Type: Designator

## See Also

### Reference

PocketBeagle Class
IO.Objects.SimpleIO.Platforms Namespace

# PocketBeagleI2C2 Field

I2C bus designator (0,2) for P1.26 and P1.28. Conflicts with `GPIO12` and `GPIO13`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator I2C2
```

### Field Value

Type: Designator

## See Also

### Reference

PocketBeagle Class
IO.Objects.SimpleIO.Platforms Namespace

# PocketBeaglePWM0_0 Field

PWM output designator (0,0) for P1.36. Conflicts with `GPIO110`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator PWM0_0
```

### Field Value

Type: Designator

## See Also

### Reference

PocketBeagle Class
IO.Objects.SimpleIO.Platforms Namespace

# PocketBeaglePWM2_0 Field

PWM output designator (2,0) for P2.1. Conflicts with `GPIO50`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator PWM2_0
```

### Field Value

Type: Designator

## See Also

### Reference

PocketBeagle Class
IO.Objects.SimpleIO.Platforms Namespace

# PocketBeagleSPI0_0 Field

SPI slave designator (0,0) for P1.6. Conflicts with `GPIO5`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator SPI0_0
```

### Field Value

Type: Designator

## See Also

### Reference

PocketBeagle Class
IO.Objects.SimpleIO.Platforms Namespace

# PocketBeagleSPI1_1 Field

SPI slave designator (1,1) for P2.31. Conflicts with `GPIO19`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator SPI1_1
```

### Field Value

Type: Designator

## See Also

### Reference

PocketBeagle Class
IO.Objects.SimpleIO.Platforms Namespace

# RaspberryPi Class

This class defines identifiers for the devices provided by the Raspberry Pi hardware platform.

## Inheritance Hierarchy

SystemObject IO.Objects.SimpleIO.PlatformsRaspberryPi

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static class RaspberryPi
```

The RaspberryPi type exposes the following members.

## Fields

| | Name | Description |
|---|---|---|
| | AIN0 | Analog input designator (0,0). Requires the Mikroelektronika Pi 3 Click Shield and the `Pi3ClickShield.dtbo` device tree overlay. |
| | AIN1 | Analog input designator (0,1). Requires the Mikroelektronika Pi 3 Click Shield and the `Pi3ClickShield.dtbo` device tree overlay. |

| | | |
|---|---|---|
| S | GPIO10 | GPIO pin designator (0,10). Conflicts with `SPI0 MOSI`. |
| S | GPIO11 | GPIO pin designator (0,11). Conflicts with `SPI0 SCLK`. |
| S | GPIO12 | GPIO pin designator (0,12). Not available on early boards with 26 pin expansion headers. |
| S | GPIO13 | GPIO pin designator (0,13). Not available on early boards with 26 pin expansion headers. |
| S | GPIO14 | GPIO pin designator (0,14). Conflicts with `UART0 TXD`. |
| S | GPIO15 | GPIO pin designator (0,15). Conflicts with `UART0 RXD`. |
| S | GPIO16 | GPIO pin designator (0,16). Conflicts with `SPI1 SS2`. Not available on early boards with 26 pin expansion headers. |
| S | GPIO17 | GPIO pin designator (0,17). Conflicts with `SPI1 SS1`. |
| S | GPIO18 | GPIO pin designator (0,18). Conflicts with `PWM0` or `SPI1 SS0`. |
| S | GPIO19 | GPIO pin designator (0,19). Conflicts with `PWM1` or `SPI1 MISO`. Not available on early boards with 26 pin expansion headers. |

| | | |
|---|---|---|
| S | GPIO2 | GPIO pin designator (0,2). Conflicts with `I2C1 SDA`. |
| S | GPIO20 | GPIO pin designator (0,20). Conflicts with `SPI1 MOSI`. Not available on early boards with 26 pin expansion headers. |
| S | GPIO21 | GPIO pin designator (0,21). Conflicts with `SPI1 SCLK`. Not available on early boards with 26 pin expansion headers. |
| S | GPIO22 | GPIO pin designator (0,22). |
| S | GPIO23 | GPIO pin designator (0,23). |
| S | GPIO24 | GPIO pin designator (0,24). |
| S | GPIO25 | GPIO pin designator (0,25). |
| S | GPIO26 | GPIO pin designator (0,26). Not available on early boards with 26 pin expansion headers. |
| S | GPIO27 | GPIO pin designator (0,27). |
| S | GPIO3 | GPIO pin designator (0,3). Conflicts with `I2C1 SCL`. |
| S | GPIO4 | GPIO pin designator (0,4). |
| S | GPIO5 | GPIO pin designator (0,5). Not available on early boards with 26 pin expansion headers. |
| S | GPIO6 | GPIO pin designator (0,6). Not available on early boards with 26 pin expansion |

| | | |
|---|---|---|
| | | headers. |
| S | GPIO7 | GPIO pin designator (0,7). Conflicts with `SPI0 SS1`. |
| S | GPIO8 | GPIO pin designator (0,8). Conflicts with `SPI0 SS0`. |
| S | GPIO9 | GPIO pin designator (0,9). Conflicts with `SPI0 MISO`. |
| S | I2C1 | I2C bus designator (0,1). Conflicts with `GPIO2` and `GPIO3`. |
| S | I2C3 | I2C bus designator (0,3). Only available on a Raspberry Pi 4, and requires the `i2c3.dtbo` device tree overlay). |
| S | I2C4 | I2C bus designator (0,4). Only available on a Raspberry Pi 4, and requires the `i2c4.dtbo` device tree overlay). |
| S | I2C5 | I2C bus designator (0,5). Only available on a Raspberry Pi 4, and requires the `i2c5.dtbo` device tree overlay). |
| S | I2C6 | I2C bus designator (0,6). Only available on a Raspberry Pi 4, and requires the `i2c6.dtbo` device tree overlay). |
| S | PWM0_0 | PWM output designator (0,0). Conflicts with `GPIO12` or `GPIO18`. Requires the `pwm.dtbo` device tree overlay. |
| S | PWM0_1 | PWM output designator (0,1). Conflicts |

| | | |
|---|---|---|
| | | with `GPIO13` or `GPIO19`. Requires the `pwm.dtbo` device tree overlay. |
| S | SPI0_0 | SPI slave designator (0,0). Conflicts with `GPIO8`. |
| S | SPI0_1 | SPI slave designator (0,1). Conflicts with `GPIO7`. |
| S | SPI1_0 | SPI slave designator (1,0). Conflicts with `GPIO18` or `PWM0`. Requires one of the `spi1-1cs.dtbo`, `spi1-2cs.dtbo` or `spi1-3cs.dtbo` device tree overlays. |
| S | SPI1_1 | SPI slave designator (1,1). Conflicts with `GPIO17`. Requires one of the `spi1-1cs.dtbo`, `spi1-2cs.dtbo` or `spi1-3cs.dtbo` device tree overlays. |
| S | SPI1_2 | SPI slave designator (1,2). Conflicts with `GPIO16`. Requires one of the `spi1-1cs.dtbo`, `spi1-2cs.dtbo` or `spi1-3cs.dtbo` device tree overlays. |

Top

## See Also

### Reference

IO.Objects.SimpleIO.Platforms Namespace

# RaspberryPi Fields

The RaspberryPi type exposes the following members.

## Fields

| | Name | Description |
|---|---|---|
| S | AIN0 | Analog input designator (0,0). Requires the Mikroelektronika Pi 3 Click Shield and the `Pi3ClickShield.dtbo` device tree overlay. |
| S | AIN1 | Analog input designator (0,1). Requires the Mikroelektronika Pi 3 Click Shield and the `Pi3ClickShield.dtbo` device tree overlay. |
| S | GPIO10 | GPIO pin designator (0,10). Conflicts with `SPI0 MOSI`. |
| S | GPIO11 | GPIO pin designator (0,11). Conflicts with `SPI0 SCLK`. |
| S | GPIO12 | GPIO pin designator (0,12). Not available on early boards with 26 pin expansion headers. |
| S | GPIO13 | GPIO pin designator (0,13). Not available on early boards with 26 pin expansion headers. |

| | | |
|---|---|---|
| S | GPIO14 | GPIO pin designator (0,14). Conflicts with `UART0 TXD`. |
| S | GPIO15 | GPIO pin designator (0,15). Conflicts with `UART0 RXD`. |
| S | GPIO16 | GPIO pin designator (0,16). Conflicts with `SPI1 SS2`. Not available on early boards with 26 pin expansion headers. |
| S | GPIO17 | GPIO pin designator (0,17). Conflicts with `SPI1 SS1`. |
| S | GPIO18 | GPIO pin designator (0,18). Conflicts with `PWM0` or `SPI1 SS0`. |
| S | GPIO19 | GPIO pin designator (0,19). Conflicts with `PWM1` or `SPI1 MISO`. Not available on early boards with 26 pin expansion headers. |
| S | GPIO2 | GPIO pin designator (0,2). Conflicts with `I2C1 SDA`. |
| S | GPIO20 | GPIO pin designator (0,20). Conflicts with `SPI1 MOSI`. Not available on early boards with 26 pin expansion headers. |
| S | GPIO21 | GPIO pin designator (0,21). Conflicts with `SPI1 SCLK`. Not available on early boards with 26 pin expansion headers. |
| S | GPIO22 | GPIO pin designator (0,22). |
| S | GPIO23 | GPIO pin designator (0,23). |

| | | |
|---|---|---|
| S | GPIO24 | GPIO pin designator (0,24). |
| S | GPIO25 | GPIO pin designator (0,25). |
| S | GPIO26 | GPIO pin designator (0,26). Not available on early boards with 26 pin expansion headers. |
| S | GPIO27 | GPIO pin designator (0,27). |
| S | GPIO3 | GPIO pin designator (0,3). Conflicts with `I2C1 SCL`. |
| S | GPIO4 | GPIO pin designator (0,4). |
| S | GPIO5 | GPIO pin designator (0,5). Not available on early boards with 26 pin expansion headers. |
| S | GPIO6 | GPIO pin designator (0,6). Not available on early boards with 26 pin expansion headers. |
| S | GPIO7 | GPIO pin designator (0,7). Conflicts with `SPI0 SS1`. |
| S | GPIO8 | GPIO pin designator (0,8). Conflicts with `SPI0 SS0`. |
| S | GPIO9 | GPIO pin designator (0,9). Conflicts with `SPI0 MISO`. |
| S | I2C1 | I2C bus designator (0,1). Conflicts with `GPIO2` and `GPIO3`. |
| S | I2C3 | I2C bus designator (0,3). Only available |

| | | |
|---|---|---|
| | | on a Raspberry Pi 4, and requires the `i2c3.dtbo` device tree overlay). |
| S | I2C4 | I2C bus designator (0,4). Only available on a Raspberry Pi 4, and requires the `i2c4.dtbo` device tree overlay). |
| S | I2C5 | I2C bus designator (0,5). Only available on a Raspberry Pi 4, and requires the `i2c5.dtbo` device tree overlay). |
| S | I2C6 | I2C bus designator (0,6). Only available on a Raspberry Pi 4, and requires the `i2c6.dtbo` device tree overlay). |
| S | PWM0_0 | PWM output designator (0,0). Conflicts with `GPIO12` or `GPIO18`. Requires the `pwm.dtbo` device tree overlay. |
| S | PWM0_1 | PWM output designator (0,1). Conflicts with `GPIO13` or `GPIO19`. Requires the `pwm.dtbo` device tree overlay. |
| S | SPI0_0 | SPI slave designator (0,0). Conflicts with `GPIO8`. |
| S | SPI0_1 | SPI slave designator (0,1). Conflicts with `GPIO7`. |
| S | SPI1_0 | SPI slave designator (1,0). Conflicts with `GPIO18` or `PWM0`. Requires one of the `spi1-1cs.dtbo`, `spi1-2cs.dtbo` or `spi1-3cs.dtbo` device tree overlays. |
| S | SPI1_1 | SPI slave designator (1,1). Conflicts with |

| | | |
|---|---|---|
| | | `GPIO17`. Requires one of the `spi1-1cs.dtbo`, `spi1-2cs.dtbo` or `spi1-3cs.dtbo` device tree overlays. |
| | SPI1_2 | SPI slave designator (1,2). Conflicts with `GPIO16`. Requires one of the `spi1-1cs.dtbo`, `spi1-2cs.dtbo` or `spi1-3cs.dtbo` device tree overlays. |

Top

## See Also

### Reference

RaspberryPi Class
IO.Objects.SimpleIO.Platforms Namespace

# RaspberryPiAIN0 Field

Analog input designator (0,0). Requires the Mikroelektronika Pi 3 Click Shield and the `Pi3ClickShield.dtbo` device tree overlay.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator AIN0
```

### Field Value

Type: Designator

## See Also

### Reference

RaspberryPi Class
IO.Objects.SimpleIO.Platforms Namespace

# RaspberryPiAIN1 Field

Analog input designator (0,1). Requires the Mikroelektronika Pi 3 Click Shield and the `Pi3ClickShield.dtbo` device tree overlay.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator AIN1
```

### Field Value

Type: Designator

## See Also

### Reference

RaspberryPi Class
IO.Objects.SimpleIO.Platforms Namespace

# RaspberryPiGPIO10 Field

GPIO pin designator (0,10). Conflicts with `SPI0 MOSI`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO10
```

### Field Value

Type: Designator

## See Also

### Reference

RaspberryPi Class
IO.Objects.SimpleIO.Platforms Namespace

# RaspberryPiGPIO11 Field

GPIO pin designator (0,11). Conflicts with `SPI0 SCLK`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO11
```

### Field Value

Type: Designator

## See Also

### Reference

RaspberryPi Class
IO.Objects.SimpleIO.Platforms Namespace

# RaspberryPiGPIO12 Field

GPIO pin designator (0,12). Not available on early boards with 26 pin expansion headers.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO12
```

### Field Value

Type: Designator

## See Also

### Reference

RaspberryPi Class
IO.Objects.SimpleIO.Platforms Namespace

# RaspberryPiGPIO13 Field

GPIO pin designator (0,13). Not available on early boards with 26 pin expansion headers.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO13
```

### Field Value

Type: Designator

## See Also

### Reference

RaspberryPi Class
IO.Objects.SimpleIO.Platforms Namespace

# RaspberryPiGPIO14 Field

GPIO pin designator (0,14). Conflicts with `UART0 TXD`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO14
```

### Field Value

Type: Designator

## See Also

### Reference

RaspberryPi Class
IO.Objects.SimpleIO.Platforms Namespace

# RaspberryPiGPIO15 Field

GPIO pin designator (0,15). Conflicts with `UART0 RXD`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO15
```

### Field Value

Type: Designator

## See Also

### Reference

RaspberryPi Class
IO.Objects.SimpleIO.Platforms Namespace

# RaspberryPiGPIO16 Field

GPIO pin designator (0,16). Conflicts with `SPI1 SS2`. Not available on early boards with 26 pin expansion headers.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO16
```

### Field Value

Type: Designator

## See Also

### Reference

RaspberryPi Class
IO.Objects.SimpleIO.Platforms Namespace

# RaspberryPiGPIO17 Field

GPIO pin designator (0,17). Conflicts with `SPI1 SS1`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO17
```

### Field Value

Type: Designator

## See Also

### Reference

RaspberryPi Class
IO.Objects.SimpleIO.Platforms Namespace

# RaspberryPiGPIO18 Field

GPIO pin designator (0,18). Conflicts with `PWM0` or `SPI1 SS0`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO18
```

### Field Value

Type: Designator

## See Also

### Reference

RaspberryPi Class
IO.Objects.SimpleIO.Platforms Namespace

# RaspberryPiGPIO19 Field

GPIO pin designator (0,19). Conflicts with `PWM1` or `SPI1 MISO`. Not available on early boards with 26 pin expansion headers.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO19
```

### Field Value

Type: Designator

## See Also

### Reference

RaspberryPi Class
IO.Objects.SimpleIO.Platforms Namespace

# RaspberryPiGPIO2 Field

GPIO pin designator (0,2). Conflicts with `I2C1 SDA`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO2
```

### Field Value

Type: Designator

## See Also

### Reference

RaspberryPi Class
IO.Objects.SimpleIO.Platforms Namespace

# RaspberryPiGPIO20 Field

GPIO pin designator (0,20). Conflicts with `SPI1 MOSI`. Not available on early boards with 26 pin expansion headers.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static readonly Designator GPIO20
```

### Field Value

Type: Designator

## See Also

### Reference

RaspberryPi Class
IO.Objects.SimpleIO.Platforms Namespace

# RaspberryPiGPIO21 Field

GPIO pin designator (0,21). Conflicts with `SPI1 SCLK`. Not available on early boards with 26 pin expansion headers.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO21
```

### Field Value

Type: Designator

## See Also

### Reference

RaspberryPi Class
IO.Objects.SimpleIO.Platforms Namespace

# RaspberryPiGPIO22 Field

GPIO pin designator (0,22).

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static readonly Designator GPIO22
```

### Field Value

Type: Designator

## See Also

### Reference

RaspberryPi Class
IO.Objects.SimpleIO.Platforms Namespace

# RaspberryPiGPIO23 Field

GPIO pin designator (0,23).

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO23
```

### Field Value

Type: Designator

## See Also

### Reference

RaspberryPi Class
IO.Objects.SimpleIO.Platforms Namespace

# RaspberryPiGPIO24 Field

GPIO pin designator (0,24).

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO24
```

### Field Value

Type: Designator

## See Also

### Reference

RaspberryPi Class
IO.Objects.SimpleIO.Platforms Namespace

# RaspberryPiGPIO25 Field

GPIO pin designator (0,25).

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO25
```

### Field Value

Type: Designator

## See Also

### Reference

RaspberryPi Class
IO.Objects.SimpleIO.Platforms Namespace

# RaspberryPiGPIO26 Field

GPIO pin designator (0,26). Not available on early boards with 26 pin expansion headers.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO26
```

### Field Value

Type: Designator

## See Also

### Reference

RaspberryPi Class
IO.Objects.SimpleIO.Platforms Namespace

# RaspberryPiGPIO27 Field

GPIO pin designator (0,27).

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO27
```

### Field Value

Type: Designator

## See Also

### Reference

RaspberryPi Class
IO.Objects.SimpleIO.Platforms Namespace

# RaspberryPiGPIO3 Field

GPIO pin designator (0,3). Conflicts with `I2C1 SCL`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO3
```

### Field Value

Type: Designator

## See Also

### Reference

RaspberryPi Class
IO.Objects.SimpleIO.Platforms Namespace

# RaspberryPiGPIO4 Field

GPIO pin designator (0,4).

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO4
```

### Field Value

Type: Designator

## See Also

### Reference

RaspberryPi Class
IO.Objects.SimpleIO.Platforms Namespace

# RaspberryPiGPIO5 Field

GPIO pin designator (0,5). Not available on early boards with 26 pin expansion headers.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO5
```

### Field Value

Type: Designator

## See Also

### Reference

RaspberryPi Class
IO.Objects.SimpleIO.Platforms Namespace

# RaspberryPiGPIO6 Field

GPIO pin designator (0,6). Not available on early boards with 26 pin expansion headers.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO6
```

### Field Value

Type: Designator

## See Also

### Reference

RaspberryPi Class
IO.Objects.SimpleIO.Platforms Namespace

# RaspberryPiGPIO7 Field

GPIO pin designator (0,7). Conflicts with `SPI0 SS1`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO7
```

### Field Value

Type: Designator

## See Also

### Reference

RaspberryPi Class
IO.Objects.SimpleIO.Platforms Namespace

# RaspberryPiGPIO8 Field

GPIO pin designator (0,8). Conflicts with `SPI0 SS0`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO8
```

### Field Value

Type: Designator

## See Also

### Reference

RaspberryPi Class
IO.Objects.SimpleIO.Platforms Namespace

# RaspberryPiGPIO9 Field

GPIO pin designator (0,9). Conflicts with `SPI0 MISO`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator GPIO9
```

### Field Value

Type: Designator

## See Also

### Reference

RaspberryPi Class
IO.Objects.SimpleIO.Platforms Namespace

# RaspberryPiI2C1 Field

I2C bus designator (0,1). Conflicts with `GPIO2` and `GPIO3`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator I2C1
```

### Field Value

Type: Designator

## See Also

### Reference

RaspberryPi Class
IO.Objects.SimpleIO.Platforms Namespace

# RaspberryPiI2C3 Field

I2C bus designator (0,3). Only available on a Raspberry Pi 4, and requires the `i2c3.dtbo` device tree overlay).

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator I2C3
```

### Field Value

Type: Designator

## See Also

### Reference

RaspberryPi Class
IO.Objects.SimpleIO.Platforms Namespace

# RaspberryPiI2C4 Field

I2C bus designator (0,4). Only available on a Raspberry Pi 4, and requires the `i2c4.dtbo` device tree overlay).

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator I2C4
```

### Field Value

Type: Designator

## See Also

### Reference

RaspberryPi Class
IO.Objects.SimpleIO.Platforms Namespace

# RaspberryPiI2C5 Field

I2C bus designator (0,5). Only available on a Raspberry Pi 4, and requires the `i2c5.dtbo` device tree overlay).

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static readonly Designator I2C5
```

### Field Value

Type: Designator

## See Also

### Reference

RaspberryPi Class
IO.Objects.SimpleIO.Platforms Namespace

# RaspberryPiI2C6 Field

I2C bus designator (0,6). Only available on a Raspberry Pi 4, and requires the `i2c6.dtbo` device tree overlay).

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static readonly Designator I2C6
```

### Field Value

Type: Designator

## See Also

### Reference

RaspberryPi Class
IO.Objects.SimpleIO.Platforms Namespace

# RaspberryPiPWM0_0 Field

PWM output designator (0,0). Conflicts with `GPIO12` or `GPIO18`. Requires the `pwm.dtbo` device tree overlay.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator PWM0_0
```

### Field Value

Type: Designator

## See Also

### Reference

RaspberryPi Class
IO.Objects.SimpleIO.Platforms Namespace

# RaspberryPiPWM0_1 Field

PWM output designator (0,1). Conflicts with `GPIO13` or `GPIO19`. Requires the `pwm.dtbo` device tree overlay.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator PWM0_1
```

### Field Value

Type: Designator

## See Also

### Reference

RaspberryPi Class
IO.Objects.SimpleIO.Platforms Namespace

# RaspberryPiSPI0_0 Field

SPI slave designator (0,0). Conflicts with `GPIO8`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator SPI0_0
```

### Field Value

Type: Designator

## See Also

### Reference

RaspberryPi Class
IO.Objects.SimpleIO.Platforms Namespace

# RaspberryPiSPI0_1 Field

SPI slave designator (0,1). Conflicts with `GPIO7`.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator SPI0_1
```

### Field Value

Type: Designator

## See Also

### Reference

RaspberryPi Class
IO.Objects.SimpleIO.Platforms Namespace

# RaspberryPiSPI1_0 Field

SPI slave designator (1,0). Conflicts with `GPIO18` or `PWM0`. Requires one of the `spi1-1cs.dtbo`, `spi1-2cs.dtbo` or `spi1-3cs.dtbo` device tree overlays.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public static readonly Designator SPI1_0
```

### Field Value

Type: Designator

## See Also

### Reference

RaspberryPi Class
IO.Objects.SimpleIO.Platforms Namespace

# RaspberryPiSPI1_1 Field

SPI slave designator (1,1). Conflicts with `GPIO17`. Requires one of the `spi1-1cs.dtbo`, `spi1-2cs.dtbo` or `spi1-3cs.dtbo` device tree overlays.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static readonly Designator SPI1_1
```

### Field Value

Type: Designator

## See Also

### Reference

RaspberryPi Class
IO.Objects.SimpleIO.Platforms Namespace

# RaspberryPiSPI1_2 Field

SPI slave designator (1,2). Conflicts with `GPIO16`. Requires one of the `spi1-1cs.dtbo`, `spi1-2cs.dtbo` or `spi1-3cs.dtbo` device tree overlays.

**Namespace:** IO.Objects.SimpleIO.Platforms
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

**C#** **VB** **F#** Copy

```
public static readonly Designator SPI1_2
```

### Field Value

Type: Designator

## See Also

### Reference

RaspberryPi Class
IO.Objects.SimpleIO.Platforms Namespace

# IO.Objects.SimpleIO.PWM Namespace

PWM (Pulse Width Modulated) Output Services

## Classes

| | Class | Description |
|---|---|---|
| | Output | Encapsulates Linux PWM outputs using `libsimpleio`. |

# Output Class

Encapsulates Linux PWM outputs using `libsimpleio`.

## Inheritance Hierarchy

SystemObject IO.Objects.SimpleIO.PWMOutput

**Namespace:** IO.Objects.SimpleIO.PWM
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public class Output : Output
```

The Output type exposes the following members.

## Constructors

| | Name | Description |
|---|---|---|
| | Output | Constructor for a single PWM output. |

Top

## Properties

| | Name | Description |
|---|---|---|
| | dutycycle | Write-only property for setting the PWM output duty cycle. Allowed values |

| | | |
|---|---|---|
| | | are 0.0 to 100.0 percent. |
| | fd | Read-only property returning the Linux file descriptor for the PWM output. |

Top

## See Also

### Reference

IO.Objects.SimpleIO.PWM Namespace

# Output Constructor

Constructor for a single PWM output.

**Namespace:** IO.Objects.SimpleIO.PWM
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public Output(
        Designator desg,
        int frequency,
        double dutycycle = 0,
        int polarity = 1
)
```

### Parameters

*desg*
Type: IO.Objects.SimpleIO.DeviceDesignator
PWM output designator.

*frequency*
Type: SystemInt32
PWM pulse frequency.

*dutycycle* **(Optional)**
Type: SystemDouble
Initial PWM output duty cycle. Allowed values are 0.0 to 100.0 percent.

*polarity* **(Optional)**

Type: SystemInt32
PWM output polarity.

## See Also

### Reference

Output Class
IO.Objects.SimpleIO.PWM Namespace

# Output Properties

The Output type exposes the following members.

## Properties

| | Name | Description |
|---|---|---|
| | dutycycle | Write-only property for setting the PWM output duty cycle. Allowed values are 0.0 to 100.0 percent. |
| | fd | Read-only property returning the Linux file descriptor for the PWM output. |

Top

## See Also

**Reference**

Output Class

IO.Objects.SimpleIO.PWM Namespace

# Outputdutycycle Property

Write-only property for setting the PWM output duty cycle. Allowed values are 0.0 to 100.0 percent.

**Namespace:** IO.Objects.SimpleIO.PWM
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public double dutycycle { set; }
```

### Property Value

Type: Double

### Implements

Outputdutycycle

## See Also

### Reference

Output Class
IO.Objects.SimpleIO.PWM Namespace

# Outputfd Property

Read-only property returning the Linux file descriptor for the PWM output.

**Namespace:** IO.Objects.SimpleIO.PWM
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public int fd { get; }
```

### Property Value

Type: Int32

## See Also

### Reference

Output Class
IO.Objects.SimpleIO.PWM Namespace

# IO.Objects.SimpleIO.Servo Namespace

Servo Output Services

## Classes

| | Class | Description |
|---|---|---|
| | Output | Encapsulates Linux servo outputs using `libsimpleio`. |

# Output Class

Encapsulates Linux servo outputs using `libsimpleio`.

## Inheritance Hierarchy

SystemObject IO.Objects.SimpleIO.ServoOutput

**Namespace:** IO.Objects.SimpleIO.Servo
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public class Output : Output
```

The Output type exposes the following members.

## Constructors

| | Name | Description |
|---|---|---|
| | Output | Constructor for a single servo output. |

Top

## Properties

| | Name | Description |
|---|---|---|
| | fd | Read-only property returning the Linux file descriptor for the servo output. |

| | |
|---|---|
| position | Write-only property for setting the servo position. Allowed values are -1.0 to +1.0. |

Top

## See Also

### Reference

IO.Objects.SimpleIO.Servo Namespace

# Output Constructor

Constructor for a single servo output.

**Namespace:** IO.Objects.SimpleIO.Servo
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public Output(
        Designator desg,
        int frequency = 50,
        double position = 0
)
```

### Parameters

*desg*
Type: IO.Objects.SimpleIO.DeviceDesignator
PWM output designator.

*frequency* **(Optional)**
Type: SystemInt32
PWM pulse frequency.

*position* **(Optional)**
Type: SystemDouble
Initial servo position.

## See Also

### Reference

Output Class

# IO.Objects.SimpleIO.Servo Namespace

# Output Properties

The Output type exposes the following members.

## Properties

| | Name | Description |
|---|---|---|
| | fd | Read-only property returning the Linux file descriptor for the servo output. |
| | position | Write-only property for setting the servo position. Allowed values are -1.0 to +1.0. |

Top

## See Also

**Reference**

Output Class

IO.Objects.SimpleIO.Servo Namespace

# Outputfd Property

Read-only property returning the Linux file descriptor for the servo output.

**Namespace:** IO.Objects.SimpleIO.Servo
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public int fd { get; }
```

### Property Value

Type: Int32

## See Also

### Reference

Output Class
IO.Objects.SimpleIO.Servo Namespace

# Outputposition Property

Write-only property for setting the servo position. Allowed values are -1.0 to +1.0.

**Namespace:** IO.Objects.SimpleIO.Servo
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public double position { set; }
```

### Property Value

Type: Double

### Implements

Outputposition

## See Also

### Reference

Output Class
IO.Objects.SimpleIO.Servo Namespace

# IO.Objects.SimpleIO.SPI Namespace

SPI (Serial Peripheral Interconnect) Device Services

## Classes

| | Class | Description |
|---|---|---|
| | Device | Encapsulates Linux SPI devices using `libsimpleio`. |

# Device Class

Encapsulates Linux SPI devices using `libsimpleio`.

## Inheritance Hierarchy

SystemObject IO.Objects.SimpleIO.SPIDevice

**Namespace:** IO.Objects.SimpleIO.SPI
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public class Device : Device
```

The Device type exposes the following members.

## Constructors

| | Name | Description |
|---|---|---|
| | Device(Designator, Int32, Int32, Int32, Pin) | Constructor for a single SPI device. |
| | Device(String, Int32, Int32, Int32, Pin) | Constructor for a single SPI device. |

Top

## Properties

| | Name | Description |
|---|---|---|
| | fd | Read-only property returning the Linux file descriptor for the SPI slave device. |

Top

## Methods

| | Name | Description |
|---|---|---|
| | Read | Read bytes from an SPI slave device. |
| | Transaction | Write and read bytes to and from an SPI slave device. |
| | Write | Write bytes to an SPI slave device. |

Top

## Fields

| | Name | Description |
|---|---|---|
| S | AUTOCHIPSELECT | Use hardware slave select. |

Top

## See Also

**Reference**

IO.Objects.SimpleIO.SPI Namespace

# Device Constructor

## Overload List

| | Name | Description |
|---|---|---|
| | Device(Designator, Int32, Int32, Int32, Pin) | Constructor for a single SPI device. |
| | Device(String, Int32, Int32, Int32, Pin) | Constructor for a single SPI device. |

Top

## See Also

**Reference**

Device Class

IO.Objects.SimpleIO.SPI Namespace

# Device Constructor (Designator, Int32, Int32, Int32, Pin)

Constructor for a single SPI device.

**Namespace:** IO.Objects.SimpleIO.SPI
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public Device(
        Designator desg,
        int mode,
        int wordsize,
        int speed,
        Pin cspin = null
)
```

### Parameters

*desg*
Type: IO.Objects.SimpleIO.DeviceDesignator
SPI device designator.

*mode*
Type: SystemInt32
SPI clock mode.

*wordsize*
Type: SystemInt32
SPI transfer word size.

*speed*

Type: SystemInt32
SPI transfer speed.

*cspin* **(Optional)**

Type: IO.Objects.SimpleIO.GPIOPin
SPI slave select GPIO pin number, or `AUTOCHIPSELECT`.

## See Also

### Reference

Device Class
Device Overload
IO.Objects.SimpleIO.SPI Namespace

# Device Constructor (String, Int32, Int32, Int32, Pin)

Constructor for a single SPI device.

**Namespace:** IO.Objects.SimpleIO.SPI
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public Device(
        string devname,
        int mode,
        int wordsize,
        int speed,
        Pin cspin = null
)
```

### Parameters

*devname*
Type: SystemString
SPI device node name.

*mode*
Type: SystemInt32
SPI clock mode.

*wordsize*
Type: SystemInt32
SPI transfer word size.

*speed*

Type: SystemInt32
SPI transfer speed.

*cspin* **(Optional)**

Type: IO.Objects.SimpleIO.GPIOPin
SPI slave select GPIO pin number, or `AUTOCHIPSELECT`.

## See Also

### Reference

Device Class
Device Overload
IO.Objects.SimpleIO.SPI Namespace

# Device Properties

The Device type exposes the following members.

## Properties

| | Name | Description |
|---|---|---|
| | fd | Read-only property returning the Linux file descriptor for the SPI slave device. |

Top

## See Also

**Reference**

Device Class

IO.Objects.SimpleIO.SPI Namespace

# Devicefd Property

Read-only property returning the Linux file descriptor for the SPI slave device.

**Namespace:** IO.Objects.SimpleIO.SPI
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public int fd { get; }
```

### Property Value

Type: Int32

## See Also

### Reference

Device Class
IO.Objects.SimpleIO.SPI Namespace

# Device Methods

The Device type exposes the following members.

## Methods

| | Name | Description |
|---|---|---|
| | Read | Read bytes from an SPI slave device. |
| | Transaction | Write and read bytes to and from an SPI slave device. |
| | Write | Write bytes to an SPI slave device. |

Top

## See Also

**Reference**

Device Class

IO.Objects.SimpleIO.SPI Namespace

# DeviceRead Method

Read bytes from an SPI slave device.

**Namespace:** IO.Objects.SimpleIO.SPI
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public void Read(
        byte[] resp,
        int resplen
)
```

### Parameters

*resp*
Type: SystemByte
Response buffer.

*resplen*
Type: SystemInt32
Number of bytes to read.

### Implements

DeviceRead(Byte, Int32)

## See Also

### Reference

Device Class
IO.Objects.SimpleIO.SPI Namespace

# DeviceTransaction Method

Write and read bytes to and from an SPI slave device.

**Namespace:** IO.Objects.SimpleIO.SPI
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public void Transaction(
        byte[] cmd,
        int cmdlen,
        byte[] resp,
        int resplen,
        int delayus = 0
)
```

### Parameters

*cmd*
Type: SystemByte
Command buffer.

*cmdlen*
Type: SystemInt32
Number of bytes to write.

*resp*
Type: SystemByte
Response buffer.

*resplen*
Type: SystemInt32
Number of bytes to read.

*delayus* **(Optional)**

Type: SystemInt32

Delay in microseconds between write and read operations.

### Implements

DeviceTransaction(Byte, Int32, Byte, Int32, Int32)

## See Also

### Reference

Device Class

IO.Objects.SimpleIO.SPI Namespace

# DeviceWrite Method

Write bytes to an SPI slave device.

**Namespace:** IO.Objects.SimpleIO.SPI
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public void Write(
        byte[] cmd,
        int cmdlen
)
```

### Parameters

*cmd*
Type: SystemByte
Command buffer.

*cmdlen*
Type: SystemInt32
Number of bytes to write.

### Implements

DeviceWrite(Byte, Int32)

## See Also

### Reference

Device Class
IO.Objects.SimpleIO.SPI Namespace

# Device Fields

The Device type exposes the following members.

## Fields

| | Name | Description |
|---|---|---|
| | AUTOCHIPSELECT | Use hardware slave select. |

Top

## See Also

**Reference**

Device Class

IO.Objects.SimpleIO.SPI Namespace

# DeviceAUTOCHIPSELECT Field

Use hardware slave select.

**Namespace:** IO.Objects.SimpleIO.SPI
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const Pin AUTOCHIPSELECT = null
```

### Field Value

Type: Pin

## See Also

### Reference

Device Class
IO.Objects.SimpleIO.SPI Namespace

# IO.Objects.SimpleIO.syslog Namespace

Error Logging Services Using the Linux `syslog` Subsystem

## Classes

| | Class | Description |
|---|---|---|
| | Logger | Encapsulates system logging services using `libsimpleio`. |

## Enumerations

| | Enumeration | Description |
|---|---|---|
| | LoggerRegistration | `TraceListener` registration options. |

# Logger Class

Encapsulates system logging services using `libsimpleio`.

## Inheritance Hierarchy

System.DiagnosticsTraceListener IO.Objects.SimpleIO.syslogLogger

**Namespace:** IO.Objects.SimpleIO.syslog
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public class Logger : TraceListener
```

The Logger type exposes the following members.

## Constructors

| | Name | Description |
|---|---|---|
| | Logger | Constructor for a logging object that uses the Linux `syslog` service. |

Top

## Methods

| | Name | Description |
|---|---|---|
| | Error(String) | Log an error message. |

| | | |
|---|---|---|
| | Error(String, Int32) | Log an error message, including an `errno` error string. |
| | Note | Log a notification message. |
| S | Register | Register a new `Logger` instance as a `TraceListener`. |
| | Warning | Log a warning message. |
| | Write | Trace interface method for posting a message. This method implementation is identical to `WriteLine()`, because `syslog` is not a stream oriented service. |
| | WriteLine | Trace interface method for posting a message. |

Top

# Fields

| | Name | Description |
|---|---|---|
| S | LOG_ALERT | Alert condition message. |
| S | LOG_AUTH | Authentication facility. |
| S | LOG_CRIT | Critical condition message. |
| S | LOG_DAEMON | System daemon/background process facility. |
| S | LOG_DEBUG | Debug message. |

| | | |
|---|---|---|
| s | LOG_EMERG | Emergency condition message. |
| s | LOG_ERR | Error message. |
| s | LOG_INFO | Informational message. |
| s | LOG_LOCAL0 | Locally defined facility. |
| s | LOG_LOCAL1 | Locally defined facility. |
| s | LOG_LOCAL2 | Locally defined facility. |
| s | LOG_LOCAL3 | Locally defined facility. |
| s | LOG_LOCAL4 | Locally defined facility. |
| s | LOG_LOCAL5 | Locally defined facility. |
| s | LOG_LOCAL6 | Locally defined facility. |
| s | LOG_LOCAL7 | Locally defined facility. |
| s | LOG_MAIL | Mail subsystem facility. |
| s | LOG_NDELAY | Open the connection to the `syslog` service immediately. Recommended. |
| s | LOG_NOTICE | Normal condition message. |
| s | LOG_ODELAY | Do not open the connection to the `syslog` service before logging the first message. Not recommended. |
| s | LOG_PERROR | Write message to both `syslog` service AND `stderr`. |

| | | |
|---|---|---|
| | LOG_PID | Prepend the caller's process ID to the message. |
| | LOG_USER | User program facility. Use `LOG_LOCALx` instead. |
| | LOG_WARNING | Warning message. |

Top

## Remarks

An instance of this class can be used either standalone or as a `System.Diagnostics.TraceListener`. The static methods `Systems.Diagnostics.Trace.Write()` and `Systems.Diagnostics.Trace.WriteIf()` most naturally fit the Linux `syslog` facility.

## Examples

Copy

```
using static System.Diagnostics.Trace;

Listeners.RemoveAt(0);
Listeners.Add(new IO.Objects.SimpleIO.syslog.Logger("H

Write("Hello, World!");
```

## See Also

### Reference

IO.Objects.SimpleIO.syslog Namespace

# Logger Constructor

Constructor for a logging object that uses the Linux `syslog` service.

**Namespace:** IO.Objects.SimpleIO.syslog
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public Logger(
        string id = "@@APPNAME@@",
        int facility = 128,
        int options = 32,
        int severity = 6
)
```

### Parameters

*id* **(Optional)**
Type: SystemString
Program identifier string.

*facility* **(Optional)**
Type: SystemInt32
`syslog` facility.

*options* **(Optional)**
Type: SystemInt32
`syslog` options.

*severity* **(Optional)**
Type: SystemInt32
`syslog` severity level.

## See Also

### Reference

Logger Class

IO.Objects.SimpleIO.syslog Namespace

# Logger Methods

The Logger type exposes the following members.

## Methods

| | Name | Description |
|---|---|---|
| | Error(String) | Log an error message. |
| | Error(String, Int32) | Log an error message, including an `errno` error string. |
| | Note | Log a notification message. |
| | Register | Register a new `Logger` instance as a `TraceListener`. |
| | Warning | Log a warning message. |
| | Write | Trace interface method for posting a message. This method implementation is identical to `WriteLine()`, because `syslog` is not a stream oriented service. |
| | WriteLine | Trace interface method for posting a message. |

Top

## See Also

**Reference**

Logger Class

IO.Objects.SimpleIO.syslog Namespace

# LoggerError Method

## Overload List

| | Name | Description |
|---|---|---|
| | Error(String) | Log an error message. |
| | Error(String, Int32) | Log an error message, including an `errno` error string. |

Top

## See Also

**Reference**

Logger Class

IO.Objects.SimpleIO.syslog Namespace

# LoggerError Method (String)

Log an error message.

**Namespace:** IO.Objects.SimpleIO.syslog
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public void Error(
        string message
)
```

### Parameters

*message*
Type: SystemString
Error message.

## See Also

### Reference

Logger Class
Error Overload
IO.Objects.SimpleIO.syslog Namespace

# LoggerError Method (String, Int32)

Log an error message, including an `errno` error string.

**Namespace:** IO.Objects.SimpleIO.syslog
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public void Error(
        string message,
        int errnum
)
```

### Parameters

*message*
Type: SystemString
Error Message.

*errnum*
Type: SystemInt32
`errno` error number.

## See Also

### Reference

Logger Class
Error Overload
IO.Objects.SimpleIO.syslog Namespace

# LoggerNote Method

Log a notification message.

**Namespace:** IO.Objects.SimpleIO.syslog
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public void Note(
        string message
)
```

### Parameters

*message*
Type: SystemString
Notification message.

## See Also

### Reference

Logger Class
IO.Objects.SimpleIO.syslog Namespace

# LoggerRegister Method

Register a new `Logger` instance as a `TraceListener`.

**Namespace:** IO.Objects.SimpleIO.syslog
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public static void Register(
        string id = "@@APPNAME@@",
        int facility = 128,
        int options = 32,
        int severity = 6,
        LoggerRegistration reghow = LoggerRegistration
)
```

### Parameters

*id* **(Optional)**
Type: SystemString
Program identifier string.

*facility* **(Optional)**
Type: SystemInt32
`syslog` facility.

*options* **(Optional)**
Type: SystemInt32
`syslog` options.

*severity* **(Optional)**

Type: SystemInt32
`syslog` severity level.

*reghow* **(Optional)**
Type: IO.Objects.SimpleIO.syslogLoggerRegistration
How to register the `TraceListener` instance.

# Examples

Copy

```
using static System.Diagnostics.Trace;

IO.Objects.SimpleIO.syslog.Logger.Register("HelloWorld

Write("Hello, World!");
```

# See Also

**Reference**
Logger Class
IO.Objects.SimpleIO.syslog Namespace

# LoggerWarning Method

Log a warning message.

**Namespace:** IO.Objects.SimpleIO.syslog
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public void Warning(
        string message
)
```

### Parameters

*message*
Type: SystemString
Warning message.

## See Also

### Reference

Logger Class
IO.Objects.SimpleIO.syslog Namespace

# LoggerWrite Method

Trace interface method for posting a message. This method implementation is identical to `WriteLine()`, because `syslog` is not a stream oriented service.

**Namespace:** IO.Objects.SimpleIO.syslog
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public virtual void Write(
        string message
)
```

### Parameters

*message*
Type: SystemString
Trace message.

## See Also

### Reference

Logger Class
IO.Objects.SimpleIO.syslog Namespace

# LoggerWriteLine Method

Trace interface method for posting a message.

**Namespace:** IO.Objects.SimpleIO.syslog
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public virtual void WriteLine(
        string message
)
```

### Parameters

*message*
Type: SystemString
Trace message.

## See Also

### Reference

Logger Class
IO.Objects.SimpleIO.syslog Namespace

# Logger Fields

The Logger type exposes the following members.

## Fields

| | Name | Description |
|---|---|---|
| S | LOG_ALERT | Alert condition message. |
| S | LOG_AUTH | Authentication facility. |
| S | LOG_CRIT | Critical condition message. |
| S | LOG_DAEMON | System daemon/background process facility. |
| S | LOG_DEBUG | Debug message. |
| S | LOG_EMERG | Emergency condition message. |
| S | LOG_ERR | Error message. |
| S | LOG_INFO | Informational message. |
| S | LOG_LOCAL0 | Locally defined facility. |
| S | LOG_LOCAL1 | Locally defined facility. |
| S | LOG_LOCAL2 | Locally defined facility. |
| S | LOG_LOCAL3 | Locally defined facility. |

| | | |
|---|---|---|
| s | LOG_LOCAL4 | Locally defined facility. |
| s | LOG_LOCAL5 | Locally defined facility. |
| s | LOG_LOCAL6 | Locally defined facility. |
| s | LOG_LOCAL7 | Locally defined facility. |
| s | LOG_MAIL | Mail subsystem facility. |
| s | LOG_NDELAY | Open the connection to the `syslog` service immediately. Recommended. |
| s | LOG_NOTICE | Normal condition message. |
| s | LOG_ODELAY | Do not open the connection to the `syslog` service before logging the first message. Not recommended. |
| s | LOG_PERROR | Write message to both `syslog` service AND `stderr`. |
| s | LOG_PID | Prepend the caller's process ID to the message. |
| s | LOG_USER | User program facility. Use `LOG_LOCALx` instead. |
| s | LOG_WARNING | Warning message. |

Top

## See Also

**Reference**

Logger Class

IO.Objects.SimpleIO.syslog Namespace

# LoggerLOG_ALERT Field

Alert condition message.

**Namespace:** IO.Objects.SimpleIO.syslog
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int LOG_ALERT = 1
```

### Field Value

Type: Int32

## See Also

### Reference

Logger Class
IO.Objects.SimpleIO.syslog Namespace

# LoggerLOG_AUTH Field

Authentication facility.

**Namespace:** IO.Objects.SimpleIO.syslog
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int LOG_AUTH = 32
```

### Field Value

Type: Int32

## See Also

### Reference

Logger Class
IO.Objects.SimpleIO.syslog Namespace

# LoggerLOG_CRIT Field

Critical condition message.

**Namespace:** IO.Objects.SimpleIO.syslog
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int LOG_CRIT = 2
```

### Field Value

Type: Int32

## See Also

### Reference

Logger Class
IO.Objects.SimpleIO.syslog Namespace

# LoggerLOG_DAEMON Field

System daemon/background process facility.

**Namespace:** IO.Objects.SimpleIO.syslog
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int LOG_DAEMON = 24
```

### Field Value

Type: Int32

## See Also

### Reference

Logger Class
IO.Objects.SimpleIO.syslog Namespace

# LoggerLOG_DEBUG Field

Debug message.

**Namespace:** IO.Objects.SimpleIO.syslog
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int LOG_DEBUG = 7
```

### Field Value

Type: Int32

## See Also

### Reference

Logger Class
IO.Objects.SimpleIO.syslog Namespace

# LoggerLOG_EMERG Field

Emergency condition message.

**Namespace:** IO.Objects.SimpleIO.syslog
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int LOG_EMERG = 0
```

### Field Value

Type: Int32

## See Also

### Reference

Logger Class
IO.Objects.SimpleIO.syslog Namespace

# LoggerLOG_ERR Field

Error message.

**Namespace:** IO.Objects.SimpleIO.syslog
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int LOG_ERR = 3
```

### Field Value

Type: Int32

## See Also

### Reference

Logger Class
IO.Objects.SimpleIO.syslog Namespace

# LoggerLOG_INFO Field

Informational message.

**Namespace:** IO.Objects.SimpleIO.syslog
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int LOG_INFO = 6
```

### Field Value

Type: Int32

## See Also

### Reference

Logger Class
IO.Objects.SimpleIO.syslog Namespace

# LoggerLOG_LOCAL0 Field

Locally defined facility.

**Namespace:** IO.Objects.SimpleIO.syslog
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int LOG_LOCAL0 = 128
```

### Field Value

Type: Int32

## See Also

### Reference

Logger Class
IO.Objects.SimpleIO.syslog Namespace

# LoggerLOG_LOCAL1 Field

Locally defined facility.

**Namespace:** IO.Objects.SimpleIO.syslog
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int LOG_LOCAL1 = 136
```

### Field Value

Type: Int32

## See Also

### Reference

Logger Class
IO.Objects.SimpleIO.syslog Namespace

# LoggerLOG_LOCAL2 Field

Locally defined facility.

**Namespace:** IO.Objects.SimpleIO.syslog
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int LOG_LOCAL2 = 144
```

### Field Value

Type: Int32

## See Also

### Reference

Logger Class
IO.Objects.SimpleIO.syslog Namespace

# LoggerLOG_LOCAL3 Field

Locally defined facility.

**Namespace:** IO.Objects.SimpleIO.syslog
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int LOG_LOCAL3 = 152
```

### Field Value

Type: Int32

## See Also

### Reference

Logger Class
IO.Objects.SimpleIO.syslog Namespace

# LoggerLOG_LOCAL4 Field

Locally defined facility.

**Namespace:** IO.Objects.SimpleIO.syslog
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int LOG_LOCAL4 = 160
```

### Field Value

Type: Int32

## See Also

### Reference

Logger Class
IO.Objects.SimpleIO.syslog Namespace

# LoggerLOG_LOCAL5 Field

Locally defined facility.

**Namespace:** IO.Objects.SimpleIO.syslog
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int LOG_LOCAL5 = 168
```

### Field Value

Type: Int32

## See Also

### Reference

Logger Class
IO.Objects.SimpleIO.syslog Namespace

# LoggerLOG_LOCAL6 Field

Locally defined facility.

**Namespace:** IO.Objects.SimpleIO.syslog
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int LOG_LOCAL6 = 176
```

### Field Value

Type: Int32

## See Also

### Reference

Logger Class
IO.Objects.SimpleIO.syslog Namespace

# LoggerLOG_LOCAL7 Field

Locally defined facility.

**Namespace:** IO.Objects.SimpleIO.syslog
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int LOG_LOCAL7 = 184
```

### Field Value

Type: Int32

## See Also

### Reference

Logger Class
IO.Objects.SimpleIO.syslog Namespace

# LoggerLOG_MAIL Field

Mail subsystem facility.

**Namespace:** IO.Objects.SimpleIO.syslog
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int LOG_MAIL = 16
```

### Field Value

Type: Int32

## See Also

### Reference

Logger Class
IO.Objects.SimpleIO.syslog Namespace

# LoggerLOG_NDELAY Field

Open the connection to the `syslog` service immediately. Recommended.

**Namespace:** IO.Objects.SimpleIO.syslog
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int LOG_NDELAY = 8
```

### Field Value

Type: Int32

## See Also

### Reference

Logger Class
IO.Objects.SimpleIO.syslog Namespace

# LoggerLOG_NOTICE Field

Normal condition message.

**Namespace:** IO.Objects.SimpleIO.syslog
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int LOG_NOTICE = 5
```

### Field Value

Type: Int32

## See Also

### Reference

Logger Class
IO.Objects.SimpleIO.syslog Namespace

# LoggerLOG_ODELAY Field

Do not open the connection to the `syslog` service before logging the first message. Not recommended.

**Namespace:** IO.Objects.SimpleIO.syslog
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int LOG_ODELAY = 8
```

### Field Value

Type: Int32

## See Also

### Reference

Logger Class
IO.Objects.SimpleIO.syslog Namespace

# LoggerLOG_PERROR Field

Write message to both `syslog` service AND `stderr`.

**Namespace:** IO.Objects.SimpleIO.syslog
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int LOG_PERROR = 32
```

### Field Value

Type: Int32

## See Also

### Reference

Logger Class
IO.Objects.SimpleIO.syslog Namespace

# LoggerLOG_PID Field

Prepend the caller's process ID to the message.

**Namespace:** IO.Objects.SimpleIO.syslog
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int LOG_PID = 1
```

### Field Value

Type: Int32

## See Also

### Reference

Logger Class
IO.Objects.SimpleIO.syslog Namespace

# LoggerLOG_USER Field

User program facility. Use `LOG_LOCALx` instead.

**Namespace:** IO.Objects.SimpleIO.syslog
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int LOG_USER = 8
```

### Field Value

Type: Int32

## See Also

### Reference

Logger Class
IO.Objects.SimpleIO.syslog Namespace

# LoggerLOG_WARNING Field

Warning message.

**Namespace:** IO.Objects.SimpleIO.syslog
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int LOG_WARNING = 4
```

### Field Value

Type: Int32

## See Also

### Reference

Logger Class
IO.Objects.SimpleIO.syslog Namespace

# LoggerRegistration Enumeration

`TraceListener` registration options.

**Namespace:** IO.Objects.SimpleIO.syslog
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public enum Registration
```

## Members

| | Member name | Value | Description |
|---|---|---|---|
| | Append | 0 | Add to the `TraceListener` list. |
| | Replace | 1 | Replace all `TraceListener` list items. |

## See Also

### Reference

IO.Objects.SimpleIO.syslog Namespace

# IO.Objects.SimpleIO.Watchdog Namespace

Watchdog Timer Services

## Classes

| | Class | Description |
|---|---|---|
| | Timer | Encapsulates Linux watchdog timers using `libsimpleio`. |

# Timer Class

Encapsulates Linux watchdog timers using `libsimpleio`.

## Inheritance Hierarchy

SystemObject IO.Objects.SimpleIO.WatchdogTimer

**Namespace:** IO.Objects.SimpleIO.Watchdog
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public class Timer : Timer
```

The Timer type exposes the following members.

## Constructors

| | Name | Description |
|---|---|---|
| | Timer | Constructor for a single watchdog timer. |

Top

## Properties

| | Name | Description |
|---|---|---|
| | fd | Read-only property returning the Linux file descriptor for the watchdog timer. |

| | | |
|---|---|---|
| | timeout | Get or set the watchdog timeout. Not all platforms may support this. Even if supported, there may be constraints. For example, some platforms allow shortening the timeout but not lengthening it. |

Top

## Methods

| | Name | Description |
|---|---|---|
| | Kick | Reset the watchdog timer. |

Top

## Fields

| | Name | Description |
|---|---|---|
| S | DefaultDevice | Default watchdog timer device name. |
| S | DefaultTimeout | Default watchdog timer timeout value (disabled). |

Top

## See Also

**Reference**

IO.Objects.SimpleIO.Watchdog Namespace

# Timer Constructor

Constructor for a single watchdog timer.

**Namespace:** IO.Objects.SimpleIO.Watchdog
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public Timer(
        string devname = "/dev/watchdog",
        int timeout = 0
)
```

### Parameters

*devname* **(Optional)**
Type: SystemString
Device node name.

*timeout* **(Optional)**
Type: SystemInt32
Watchdog timeout setting in seconds, or `DefaultTimeout`.

## See Also

### Reference

Timer Class
IO.Objects.SimpleIO.Watchdog Namespace

# Timer Properties

The Timer type exposes the following members.

## Properties

| | Name | Description |
|---|---|---|
| | fd | Read-only property returning the Linux file descriptor for the watchdog timer. |
| | timeout | Get or set the watchdog timeout. Not all platforms may support this. Even if supported, there may be constraints. For example, some platforms allow shortening the timeout but not lengthening it. |

Top

## See Also

**Reference**

Timer Class

IO.Objects.SimpleIO.Watchdog Namespace

# Timerfd Property

Read-only property returning the Linux file descriptor for the watchdog timer.

**Namespace:** IO.Objects.SimpleIO.Watchdog
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public int fd { get; }
```

### Property Value

Type: Int32

## See Also

### Reference

Timer Class
IO.Objects.SimpleIO.Watchdog Namespace

# Timertimeout Property

Get or set the watchdog timeout. Not all platforms may support this. Even if supported, there may be constraints. For example, some platforms allow shortening the timeout but not lengthening it.

**Namespace:** IO.Objects.SimpleIO.Watchdog
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public int timeout { get; set; }
```

### Property Value

Type: Int32

### Implements

Timertimeout

## See Also

### Reference

Timer Class
IO.Objects.SimpleIO.Watchdog Namespace

# Timer Methods

The Timer type exposes the following members.

## Methods

| | Name | Description |
|---|---|---|
| | Kick | Reset the watchdog timer. |

Top

## See Also

**Reference**

Timer Class

IO.Objects.SimpleIO.Watchdog Namespace

# TimerKick Method

Reset the watchdog timer.

**Namespace:** IO.Objects.SimpleIO.Watchdog
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F# Copy

```
public void Kick()
```

### Implements

TimerKick

## See Also

### Reference

Timer Class
IO.Objects.SimpleIO.Watchdog Namespace

# Timer Fields

The Timer type exposes the following members.

## Fields

| | Name | Description |
|---|---|---|
| | DefaultDevice | Default watchdog timer device name. |
| | DefaultTimeout | Default watchdog timer timeout value (disabled). |

Top

## See Also

**Reference**

Timer Class

IO.Objects.SimpleIO.Watchdog Namespace

# TimerDefaultDevice Field

Default watchdog timer device name.

**Namespace:** IO.Objects.SimpleIO.Watchdog
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const string DefaultDevice = "/dev/watchdog"
```

### Field Value

Type: String

## See Also

### Reference

Timer Class
IO.Objects.SimpleIO.Watchdog Namespace

# TimerDefaultTimeout Field

Default watchdog timer timeout value (disabled).

**Namespace:** IO.Objects.SimpleIO.Watchdog
**Assembly:** libsimpleio (in libsimpleio.dll) Version: 2.2023.62.1

## Syntax

C# VB F#

Copy

```
public const int DefaultTimeout = 0
```

### Field Value

Type: Int32

## See Also

### Reference

Timer Class
IO.Objects.SimpleIO.Watchdog Namespace

# Index


IO.Bindings 1
libsimpleio Class 2
libsimpleio Methods 18
ADC_close Method 26
ADC_get_name Method 27
ADC_open Method 29
ADC_read Method 31
DAC_close Method 33
DAC_get_name Method 34
DAC_open Method 36
DAC_write Method 38
EVENT_close Method 40
EVENT_modify_fd Method 41
EVENT_open Method 43
EVENT_register_fd Method 44
EVENT_unregister_fd Method 46
EVENT_wait Method 48
GPIO_chip_info Method 50
GPIO_close Method 52
GPIO_configure Method 53
GPIO_line_close Method 55
GPIO_line_event Method 57
GPIO_line_info Method 59
GPIO_line_open Method 61
GPIO_line_read Method 63
GPIO_line_write Method 65
GPIO_open Method 67

GPIO_read Method 69
GPIO_write Method 71
HIDRAW_close Method 73
HIDRAW_get_info Method 75
HIDRAW_get_name Method 77
HIDRAW_open1 Method 79
HIDRAW_open2 Method 81
HIDRAW_open3 Method 83
HIDRAW_receive Method 85
HIDRAW_send Method 87
I2C_close Method 89
I2C_open Method 90
I2C_transaction Method 92
IPV4_ntoa Method 94
IPV4_resolve Method 96
LINUX_closelog Method 98
LINUX_command Method 99
LINUX_detach Method 101
LINUX_drop_privileges Method 102
LINUX_errno Method 104
LINUX_openlog Method 105
LINUX_poll Method 107
LINUX_strerror Method 109
LINUX_syslog Method 111
LINUX_usleep Method 113
PWM_close Method 114
PWM_configure Method 115
PWM_open Method 117
PWM_write Method 119

SERIAL_close Method 121
SERIAL_open Method 122
SERIAL_receive Method 124
SERIAL_send Method 126
SPI_close Method 128
SPI_open Method 129
SPI_transaction Method 131
STREAM_decode_frame Method 133
STREAM_encode_frame Method 135
STREAM_receive_frame Method 137
STREAM_send_frame Method 139
TCP4_accept Method 141
TCP4_close Method 143
TCP4_connect Method 144
TCP4_receive Method 146
TCP4_send Method 148
TCP4_server Method 150
UDP4_close Method 152
UDP4_open Method 153
UDP4_receive Method 155
UDP4_send Method 157
WATCHDOG_close Method 159
WATCHDOG_get_timeout Method 161
WATCHDOG_kick Method 163
WATCHDOG_open Method 165
WATCHDOG_set_timeout Method 167
libsimpleio Fields 169
GPIO_DIRECTION_INPUT Field 178
GPIO_DIRECTION_OUTPUT Field 179

GPIO_DRIVER_OPENDRAIN Field 180
GPIO_DRIVER_OPENSOURCE Field 181
GPIO_DRIVER_PUSHPULL Field 182
GPIO_EDGE_BOTH Field 183
GPIO_EDGE_FALLING Field 184
GPIO_EDGE_NONE Field 185
GPIO_EDGE_RISING Field 186
GPIO_EVENT_REQUEST_BOTH Field 187
GPIO_EVENT_REQUEST_FALLING Field 188
GPIO_EVENT_REQUEST_NONE Field 189
GPIO_EVENT_REQUEST_RISING Field 190
GPIO_LINE_INFO_ACTIVE_LOW Field 191
GPIO_LINE_INFO_KERNEL Field 192
GPIO_LINE_INFO_OPEN_DRAIN Field 193
GPIO_LINE_INFO_OPEN_SOURCE Field 194
GPIO_LINE_INFO_OUTPUT Field 195
GPIO_LINE_REQUEST_ACTIVE_HIGH Field 196
GPIO_LINE_REQUEST_ACTIVE_LOW Field 197
GPIO_LINE_REQUEST_INPUT Field 198
GPIO_LINE_REQUEST_OPEN_DRAIN Field 199
GPIO_LINE_REQUEST_OPEN_SOURCE Field 200
GPIO_LINE_REQUEST_OUTPUT Field 201
GPIO_LINE_REQUEST_PUSH_PULL Field 202
GPIO_POLARITY_ACTIVEHIGH Field 203
GPIO_POLARITY_ACTIVELOW Field 204
INADDR_ANY Field 205
INADDR_BROADCAST Field 206
INADDR_LOOPBACK Field 207
LOG_ALERT Field 208

LOG_AUTH Field 209
LOG_AUTHPRIV Field 210
LOG_CONS Field 211
LOG_CRIT Field 212
LOG_CRON Field 213
LOG_DAEMON Field 214
LOG_DEBUG Field 215
LOG_EMERG Field 216
LOG_ERR Field 217
LOG_FTP Field 218
LOG_INFO Field 219
LOG_KERN Field 220
LOG_LOCAL0 Field 221
LOG_LOCAL1 Field 222
LOG_LOCAL2 Field 223
LOG_LOCAL3 Field 224
LOG_LOCAL4 Field 225
LOG_LOCAL5 Field 226
LOG_LOCAL6 Field 227
LOG_LOCAL7 Field 228
LOG_LPR Field 229
LOG_MAIL Field 230
LOG_NDELAY Field 231
LOG_NEWS Field 232
LOG_NOTICE Field 233
LOG_NOWAIT Field 234
LOG_ODELAY Field 235
LOG_PERROR Field 236
LOG_PID Field 237

LOG_PROGNAME Field 238
LOG_SYSLOG Field 239
LOG_USER Field 240
LOG_UUCP Field 241
LOG_WARNING Field 242
MSG_DONTROUTE Field 243
MSG_DONTWAIT Field 244
MSG_MORE Field 245
POLLERR Field 246
POLLHUP Field 247
POLLIN Field 248
POLLNVAL Field 249
POLLOUT Field 250
POLLPRI Field 251
PWM_POLARITY_ACTIVEHIGH Field 252
PWM_POLARITY_ACTIVELOW Field 253
SERIAL_PARITY_EVEN Field 254
SERIAL_PARITY_NONE Field 255
SERIAL_PARITY_ODD Field 256
SPI_AUTO_CS Field 257
IO.Devices.A4988 258
Device Class 259
Device Constructor 261
Device Properties 263
StepsPerRotation Property 264
Device Methods 265
Disable Method 266
Enable Method 267
Move Method 268

Reset Method 270
Sleep Method 271
Spin Method 272
Wakeup Method 274
IO.Devices.AD5593R 275
Device Class 276
Device Constructor 279
Device Properties 280
ADC_Reference Property 282
DAC_Reference Property 283
GPIO_Inputs Property 284
GPIO_Outputs Property 285
Device Methods 286
ADC_Create Method 287
ConfigureChannel Method 288
DAC_Create Method 289
GPIO_Create Method 291
Read_ADC Method 293
Write_DAC Method 294
Device Fields 295
ADC_Resolution Field 296
DAC_Resolution Field 297
MaxChannel Field 298
MinChannel Field 299
PinMode Enumeration 300
ReferenceMode Enumeration 302
IO.Devices.AD5593R.ADC 304
Sample Class 305
Sample Constructor 307

Sample Properties 308
resolution Property 309
sample Property 310
IO.Devices.AD5593R.DAC 311
Sample Class 312
Sample Constructor 314
Sample Properties 316
resolution Property 317
sample Property 318
IO.Devices.AD5593R.GPIO 319
Pin Class 320
Pin Constructor 322
Pin Properties 324
state Property 325
IO.Devices.ADC121C021 326
Sample Class 327
Sample Constructor 329
Sample Properties 330
resolution Property 331
sample Property 332
IO.Devices.ClickBoards.RemoteIO.ADAC 333
Board Class 334
Board Constructor 336
Board Properties 338
device Property 339
Board Methods 340
ADC Method 341
DAC Method 342
GPIO Method 344

Reset Method 346
Board Fields 347
DefaultAddress Field 348
IO.Devices.ClickBoards.RemoteIO.Expand 349
Board Class 350
Board Constructor 352
Board Properties 353
device Property 354
Board Methods 355
GPIO Method 356
Reset Method 358
IO.Devices.ClickBoards.RemoteIO.Expand2 359
Board Class 360
Board Constructor 362
Board Properties 364
device Property 365
Board Methods 366
GPIO Method 367
Reset Method 369
Board Fields 370
DefaultAddress Field 371
IO.Devices.ClickBoards.RemoteIO.PWM 372
Board Class 373
Board Constructor 375
Board Properties 377
dev Property 378
Board Methods 379
GPIO Method 380
PWM Method 382

Servo Method 384
Board Fields 386
DefaultAddress Field 387
IO.Devices.ClickBoards.RemoteIO.SevenSegment 388
Board Class 389
Board Constructor 392
Board Properties 394
blanking Property 396
brightness Property 397
leftdp Property 398
radix Property 399
rightdp Property 400
state Property 401
Board Methods 402
Clear Method 403
Reset Method 404
Board.Base Enumeration 405
Board.ZeroBlanking Enumeration 406
IO.Devices.ClickBoards.SimpleIO.ADAC 407
Board Class 408
Board Constructor 410
Board Properties 411
device Property 412
Board Methods 413
ADC Method 414
DAC Method 415
GPIO Method 417
Reset Method 419
Board Fields 420

DefaultAddress Field 421
IO.Devices.ClickBoards.SimpleIO.Expand 422
Board Class 423
Board Constructor 425
Board Properties 426
device Property 427
Board Methods 428
GPIO Method 429
Reset Method 431
IO.Devices.ClickBoards.SimpleIO.Expand2 432
Board Class 433
Board Constructor 435
Board Properties 436
device Property 437
Board Methods 438
GPIO Method 439
Reset Method 441
Board Fields 442
DefaultAddress Field 443
IO.Devices.ClickBoards.SimpleIO.PWM 444
Board Class 445
Board Constructor 447
Board Properties 449
dev Property 450
Board Methods 451
GPIO Method 452
PWM Method 454
Servo Method 456
Board Fields 458

DefaultAddress Field 459
IO.Devices.ClickBoards.SimpleIO.SevenSegment 460
Board Class 461
Board Constructor 464
Board Properties 466
blanking Property 468
brightness Property 469
leftdp Property 470
radix Property 471
rightdp Property 472
state Property 473
Board Methods 474
Clear Method 475
Reset Method 476
Board.Base Enumeration 477
Board.ZeroBlanking Enumeration 478
IO.Devices.Grove.ADC 479
Device Class 480
Device Constructor 482
Device Properties 483
voltage Property 484
IO.Devices.Grove.Temperature 485
Device Class 486
Device Constructor 488
Device Properties 489
Celsius Property 490
Fahrenheit Property 491
Kelvins Property 492
IO.Devices.Grove.Temperature_Humidity 493

Device Class 494
Device Constructor 496
Device Properties 497
IO.Devices.HDC1080 499
Device Class 500
Device Constructor 503
Device Properties 504
Celsius Property 506
DeviceID Property 507
Fahrenheit Property 508
Humidity Property 509
Kelvins Property 510
ManufacturerID Property 511
Device Methods 512
Read Method 513
Write Method 514
Device Fields 515
RegConfiguration Field 517
RegDeviceID Field 518
RegHumidity Field 519
RegManufacturerID Field 520
RegSerialNumberFirst Field 521
RegSerialNumberLast Field 522
RegSerialNumberMid Field 523
RegTemperature Field 524
IO.Devices.MCP23017 525
Device Class 526
Device Constructor 529
Device Properties 530

Direction Property 532
DirectionA Property 533
DirectionB Property 534
Polarity Property 535
PolarityA Property 536
PolarityB Property 537
Port Property 538
PortA Property 539
PortB Property 540
Pullups Property 541
PullupsA Property 542
PullupsB Property 543
Device Methods 544
GPIO_Create Method 545
Device Fields 547
MaxChannel Field 548
MinChannel Field 549
IO.Devices.MCP23017.GPIO 550
Pin Class 551
Pin Constructor 553
Pin Properties 555
state Property 556
IO.Devices.MCP23S17 557
Device Class 558
Device Constructor 562
Device Properties 563
Direction Property 565
DirectionA Property 566
DirectionB Property 567

Polarity Property 568
PolarityA Property 569
PolarityB Property 570
Port Property 571
PortA Property 572
PortB Property 573
Pullups Property 574
PullupsA Property 575
PullupsB Property 576
Device Methods 577
GPIO_Create Method 578
Device Fields 580
MaxChannel Field 581
MinChannel Field 582
SPI_Frequency Field 583
SPI_Mode Field 584
SPI_WordSize Field 585
IO.Devices.MCP23S17.GPIO 586
Pin Class 587
Pin Constructor 589
Pin Properties 591
state Property 592
IO.Devices.PCA8574 593
Device Class 594
Device Constructor 596
Device Properties 598
Latch Property 599
Device Methods 600
Read Method 601

Write Method 602
Device Fields 603
MAX_PINS Field 604
IO.Devices.PCA8574.GPIO 605
Pin Class 606
Pin Constructor 608
Pin Properties 610
state Property 611
IO.Devices.PCA9534 612
Device Class 613
Device Constructor 616
Device Properties 618
Config Property 619
Latch Property 620
Device Methods 621
Read Method 622
Read Method 623
Read Method (Byte) 624
Write Method 625
Write Method (Byte) 626
Write Method (Byte, Byte) 627
Device Fields 628
AllInputs Field 630
AllNormal Field 631
AllOff Field 632
AllOutputs Field 633
ConfigurationReg Field 634
InputPolarityReg Field 635
InputPortReg Field 636

MAX_PINS Field 637
OutputPortReg Field 638
IO.Devices.PCA9534.GPIO 639
Pin Class 640
Pin Constructor 642
Pin Properties 644
state Property 645
IO.Devices.PCA9685 646
Device Class 647
Device Constructor 650
Device Properties 652
Frequency Property 653
Device Methods 654
ReadChannel Method 655
WriteChannel Method 656
Device Fields 657
INTERNAL_CLOCK Field 658
MAX_CHANNEL Field 659
MAX_CLOCK Field 660
MIN_CHANNEL Field 661
MIN_CLOCK Field 662
IO.Devices.PCA9685.GPIO 663
Pin Class 664
Pin Constructor 666
Pin Properties 668
state Property 669
IO.Devices.PCA9685.PWM 670
Output Class 671
Output Constructor 673

Output Properties 675
dutycycle Property 676
IO.Devices.PCA9685.Servo 677
Output Class 678
Output Constructor 680
Output Properties 682
position Property 683
IO.Devices.Pmod.HYGRO 684
Device Class 685
Device Constructor 688
Device Properties 689
Device Methods 691
IO.Devices.SN74HC595 692
Device Class 693
Device Constructor 696
Device Properties 698
Length Property 699
state Property 700
Device Methods 701
ClrBit Method 702
ReadBit Method 703
SetBit Method 705
Device Fields 706
SPI_MaxFreq Field 707
SPI_Mode Field 708
IO.Devices.SN74HC595.GPIO 709
Pin Class 710
Pin Constructor 712
Pin Properties 714

state Property 715
IO.Devices.TH02 716
Device Class 717
Device Constructor 719
Device Properties 720
Celsius Property 721
DeviceID Property 722
Fahrenheit Property 723
Humidity Property 724
Kelvins Property 725
IO.Devices.Thermistor 726
NTC_B Class 727
NTC_B Constructor 729
NTC_B Methods 731
Kelvins Method 732
IO.Devices.USB.Munts 733
HID Class 734
HID Fields 736
Product Field 737
Vendor Field 738
Serial Class 739
Serial Fields 741
Product Field 742
Vendor Field 743
IO.Interfaces.ADC 744
Input Class 745
Input Constructor 747
Input Properties 749
voltage Property 750

Sample Interface 751
Sample Properties 752
resolution Property 753
sample Property 754
Voltage Interface 755
Voltage Properties 756
voltage Property 757
IO.Interfaces.DAC 758
Output Class 759
Output Constructor 761
Output Properties 763
voltage Property 764
Sample Interface 765
Sample Properties 767
resolution Property 768
sample Property 769
Voltage Interface 770
Voltage Properties 771
voltage Property 772
IO.Interfaces.GPIO 773
Direction Enumeration 774
Pin Interface 775
Pin Properties 776
state Property 777
IO.Interfaces.Humidity 778
Sensor Interface 779
Sensor Properties 780
Humidity Property 781
IO.Interfaces.I2C 782

Bus Interface 783
Bus Methods 785
Read Method 786
Transaction Method 788
Write Method 790
Device Class 792
Device Constructor 794
Device Methods 795
Read Method 796
Transaction Method 797
Write Method 799
Speeds Class 800
Speeds Fields 802
FastMode Field 803
FastModePlus Field 804
StandardMode Field 805
IO.Interfaces.Message.Text 806
Relay Interface 807
Relay Methods 808
Send Method 809
IO.Interfaces.Message64 811
Message Class 812
Message Constructor 814
Message Constructor 815
Message Constructor (Byte) 816
Message Fields 817
payload Field 818
Size Field 819
Messenger Interface 820

Messenger Methods 822
Receive Method 823
Send Method 824
Transaction Method 825
IO.Interfaces.Motor 826
Output Interface 827
Output Properties 828
velocity Property 829
Velocities Class 830
Velocities Fields 832
Maximum Field 833
Minimum Field 834
Stop Field 835
IO.Interfaces.PWM 836
DutyCycles Class 837
DutyCycles Fields 839
Maximum Field 840
Minimum Field 841
Output Interface 842
Output Properties 843
dutycycle Property 844
IO.Interfaces.Servo 845
Output Interface 846
Output Properties 847
position Property 848
Positions Class 849
Positions Fields 851
Maximum Field 852
Minimum Field 853

Neutral Field 854
IO.Interfaces.SPI 855
Device Interface 856
Device Methods 858
Read Method 859
Transaction Method 860
Write Method 862
IO.Interfaces.Stepper 863
Output Interface 864
Output Properties 866
StepsPerRotation Property 867
Output Methods 868
Move Method 869
Spin Method 871
IO.Interfaces.Temperature 873
Conversions Class 874
Conversions Constructor 876
Conversions Methods 877
CelsiusToFahrenheit Method 879
CelsiusToKelvins Method 880
FahrenheitToCelsius Method 881
FahrenheitToKelvins Method 882
KelvinsToCelsius Method 883
KelvinsToFahrenheit Method 884
Sensor Interface 885
Sensor Properties 887
Celsius Property 888
Fahrenheit Property 889
Kelvins Property 890

IO.Interfaces.Watchdog 891
Timer Interface 892
Timer Properties 894
timeout Property 895
Timer Methods 896
Kick Method 897
IO.Objects.Email.Mail 898
Relay Class 899
Relay Constructor 901
Relay Methods 902
Send Method 903
Send Method (String, String, String) 904
Send Method (String, String, String, String) 906
Send Method (String, String, String, String, String) 908
IO.Objects.Email.SMTP 910
Relay Class 911
Relay Constructor 913
Relay Methods 914
Send Method 915
Send Method (String, String, String) 916
Send Method (String, String, String, String) 918
Send Method (String, String, String, String, String) 920
IO.Objects.GPIO.PWM 922
Pin Class 923
Pin Constructor 925
Pin Properties 927
state Property 928
IO.Objects.Message64.UDP 929
Messenger Class 930

Messenger Constructor 932
Messenger Methods 934
Receive Method 935
Send Method 936
Transaction Method 937
IO.Objects.Motor.PWM 939
Output Class 940
Output Constructor 942
Output Constructor (Output, Output, Double) 943
Output Constructor (Pin, Output, Double) 945
Output Properties 947
velocity Property 948
IO.Objects.Motor.Servo 949
Output Class 950
Output Constructor 952
Output Properties 953
velocity Property 954
IO.Objects.RemoteIO 955
ADC Class 957
ADC Constructor 959
ADC Properties 961
resolution Property 962
sample Property 963
DAC Class 964
DAC Constructor 966
DAC Properties 968
resolution Property 969
sample Property 970
Device Class 971

Device Constructor 974
Device Constructor (Messenger) 975
Device Constructor (Int32, Int32, String, Int32) 976
Device Properties 978
Capabilities Property 979
Version Property 980
Device Methods 981
ADC_Available Method 983
ADC_Create Method 984
DAC_Available Method 985
DAC_Create Method 986
Dispatcher Method 988
GPIO_Available Method 989
GPIO_Create Method 990
I2C_Available Method 992
I2C_Create Method 993
PWM_Available Method 995
PWM_Create Method 996
SPI_Available Method 998
SPI_Create Method 999
Device Fields 1001
MAX_CHANNELS Field 1002
Unavailable Field 1003
GPIO Class 1004
GPIO Constructor 1006
GPIO Properties 1008
state Property 1009
I2C Class 1010
I2C Constructor 1012

I2C Methods 1014
Read Method 1015
Transaction Method 1017
Write Method 1019
MessageTypes Enumeration 1021
PeripheralTypes Enumeration 1027
PWM Class 1029
PWM Constructor 1031
PWM Properties 1033
dutycycle Property 1034
SPI Class 1035
SPI Constructor 1037
SPI Methods 1039
Read Method 1040
Transaction Method 1042
Write Method 1044
IO.Objects.RemoteIO.mikroBUS 1046
Shield Class 1047
Shield Properties 1049
kind Property 1050
Shield Fields 1051
I2CBus Field 1052
Shield.Kinds Enumeration 1053
Socket Class 1055
Socket Constructor 1058
Socket Properties 1059
AIN Property 1061
AN Property 1062
CS Property 1063

I2CBus Property 1064
INT Property 1065
MISO Property 1066
MOSI Property 1067
PWM Property 1068
PWMOut Property 1069
RST Property 1070
RX Property 1071
SCK Property 1072
SCL Property 1073
SDA Property 1074
SPIDev Property 1075
TX Property 1076
IO.Objects.RemoteIO.Platforms 1077
BeagleBone Class 1078
BeagleBone Fields 1089
AIN0 Field 1099
AIN1 Field 1100
AIN2 Field 1101
AIN3 Field 1102
AIN4 Field 1103
AIN5 Field 1104
AIN6 Field 1105
EHRPWM1A Field 1106
EHRPWM1B Field 1107
EHRPWM2A Field 1108
EHRPWM2B Field 1109
GPIO10 Field 1110
GPIO11 Field 1111

GPIO110 Field 1112
GPIO111 Field 1113
GPIO112 Field 1114
GPIO113 Field 1115
GPIO115 Field 1116
GPIO117 Field 1117
GPIO12 Field 1118
GPIO13 Field 1119
GPIO14 Field 1120
GPIO15 Field 1121
GPIO2 Field 1122
GPIO20 Field 1123
GPIO22 Field 1124
GPIO23 Field 1125
GPIO26 Field 1126
GPIO27 Field 1127
GPIO3 Field 1128
GPIO30 Field 1129
GPIO31 Field 1130
GPIO32 Field 1131
GPIO33 Field 1132
GPIO34 Field 1133
GPIO35 Field 1134
GPIO36 Field 1135
GPIO37 Field 1136
GPIO38 Field 1137
GPIO39 Field 1138
GPIO4 Field 1139
GPIO44 Field 1140

GPIO45 Field 1141
GPIO46 Field 1142
GPIO47 Field 1143
GPIO48 Field 1144
GPIO49 Field 1145
GPIO5 Field 1146
GPIO50 Field 1147
GPIO51 Field 1148
GPIO60 Field 1149
GPIO61 Field 1150
GPIO62 Field 1151
GPIO63 Field 1152
GPIO65 Field 1153
GPIO66 Field 1154
GPIO67 Field 1155
GPIO68 Field 1156
GPIO69 Field 1157
GPIO7 Field 1158
GPIO70 Field 1159
GPIO71 Field 1160
GPIO72 Field 1161
GPIO73 Field 1162
GPIO74 Field 1163
GPIO75 Field 1164
GPIO76 Field 1165
GPIO77 Field 1166
GPIO78 Field 1167
GPIO79 Field 1168
GPIO8 Field 1169

GPIO80 Field 1170
GPIO81 Field 1171
GPIO86 Field 1172
GPIO87 Field 1173
GPIO88 Field 1174
GPIO89 Field 1175
GPIO9 Field 1176
I2C2 Field 1177
SPI0_0 Field 1178
SPI1_0 Field 1179
SPI1_1 Field 1180
USERLED Field 1181
MUNTS_0018 Class 1182
MUNTS_0018 Fields 1186
D1 Field 1189
J10A0 Field 1190
J10A1 Field 1191
J11A0 Field 1192
J11A1 Field 1193
J2PWM Field 1194
J3PWM Field 1195
J4D0 Field 1196
J4D1 Field 1197
J5D0 Field 1198
J5D1 Field 1199
J6D0 Field 1200
J6D1 Field 1201
J6DIR Field 1202
J6PWM Field 1203

J7D0 Field 1204
J7D1 Field 1205
J7DIR Field 1206
J7PWM Field 1207
J9I2C Field 1208
SW1 Field 1209
USERLED Field 1210
PocketBeagle Class 1211
PocketBeagle Fields 1219
AIN0 Field 1227
AIN1 Field 1228
AIN2 Field 1229
AIN3 Field 1230
AIN4 Field 1231
AIN5 Field 1232
AIN6 Field 1233
AIN7 Field 1234
GPIO110 Field 1235
GPIO111 Field 1236
GPIO112 Field 1237
GPIO113 Field 1238
GPIO114 Field 1239
GPIO115 Field 1240
GPIO116 Field 1241
GPIO117 Field 1242
GPIO12 Field 1243
GPIO13 Field 1244
GPIO14 Field 1245
GPIO15 Field 1246

GPIO19 Field 1247
GPIO2 Field 1248
GPIO20 Field 1249
GPIO23 Field 1250
GPIO26 Field 1251
GPIO27 Field 1252
GPIO3 Field 1253
GPIO30 Field 1254
GPIO31 Field 1255
GPIO4 Field 1256
GPIO40 Field 1257
GPIO41 Field 1258
GPIO42 Field 1259
GPIO43 Field 1260
GPIO44 Field 1261
GPIO45 Field 1262
GPIO46 Field 1263
GPIO47 Field 1264
GPIO5 Field 1265
GPIO50 Field 1266
GPIO52 Field 1267
GPIO57 Field 1268
GPIO58 Field 1269
GPIO59 Field 1270
GPIO60 Field 1271
GPIO64 Field 1272
GPIO65 Field 1273
GPIO7 Field 1274
GPIO86 Field 1275

GPIO87 Field 1276
GPIO88 Field 1277
GPIO89 Field 1278
I2C1 Field 1279
I2C2 Field 1280
PWM0_0 Field 1281
PWM2_0 Field 1282
SPI0_0 Field 1283
SPI1_1 Field 1284
USERLED Field 1285
RaspberryPi Class 1286
RaspberryPi Fields 1290
AIN0 Field 1294
AIN1 Field 1295
GPIO10 Field 1296
GPIO11 Field 1297
GPIO12 Field 1298
GPIO13 Field 1299
GPIO14 Field 1300
GPIO15 Field 1301
GPIO16 Field 1302
GPIO17 Field 1303
GPIO18 Field 1304
GPIO19 Field 1305
GPIO2 Field 1306
GPIO20 Field 1307
GPIO21 Field 1308
GPIO22 Field 1309
GPIO23 Field 1310

GPIO24 Field 1311
GPIO25 Field 1312
GPIO26 Field 1313
GPIO27 Field 1314
GPIO3 Field 1315
GPIO4 Field 1316
GPIO5 Field 1317
GPIO6 Field 1318
GPIO7 Field 1319
GPIO8 Field 1320
GPIO9 Field 1321
I2C1 Field 1322
PWM0 Field 1323
PWM1 Field 1324
SPI0 Field 1325
SPI1 Field 1326
USERLED Field 1327
IO.Objects.Servo.PWM 1328
Output Class 1329
Output Constructor 1331
Output Properties 1333
position Property 1334
IO.Objects.SimpleIO.ADC 1335
Sample Class 1336
Sample Constructor 1338
Sample Properties 1339
fd Property 1340
resolution Property 1341
sample Property 1342

Sample Methods 1343
name Method 1344
IO.Objects.SimpleIO.DAC 1345
Sample Class 1346
Sample Constructor 1348
Sample Properties 1350
fd Property 1351
resolution Property 1352
sample Property 1353
Sample Methods 1354
name Method 1355
IO.Objects.SimpleIO.Device 1356
Designator Structure 1357
Designator Constructor 1359
Designator Properties 1360
available Property 1361
Designator Fields 1362
chan Field 1363
chip Field 1364
Unavailable Field 1365
IO.Objects.SimpleIO.GPIO 1366
Driver Enumeration 1367
Edge Enumeration 1368
Pin Class 1370
Pin Constructor 1372
Pin Properties 1374
fd Property 1375
state Property 1376
Polarity Enumeration 1377

IO.Objects.SimpleIO.HID 1378
Messenger Class 1379
Messenger Constructor 1382
Messenger Constructor (String, Int32) 1383
Messenger Constructor (Int32, Int32, String, Int32) 1385
Messenger Properties 1387
bustype Property 1388
fd Property 1389
name Property 1390
product Property 1391
vendor Property 1392
Messenger Methods 1393
Receive Method 1394
Send Method 1395
Transaction Method 1396
IO.Objects.SimpleIO.I2C 1398
Bus Class 1399
Bus Constructor 1401
Bus Constructor (Designator) 1402
Bus Constructor (String) 1403
Bus Properties 1404
fd Property 1405
Bus Methods 1406
Read Method 1407
Transaction Method 1409
Write Method 1411
IO.Objects.SimpleIO.mikroBUS 1413
Shield Class 1414
Shield Properties 1416

kind Property 1417
Shield Fields 1418
I2CBus Field 1419
Shield.Kinds Enumeration 1420
Socket Class 1423
Socket Constructor 1426
Socket Properties 1427
AIN Property 1429
AN Property 1430
CS Property 1431
I2CBus Property 1432
INT Property 1433
MISO Property 1434
MOSI Property 1435
PWM Property 1436
PWMOut Property 1437
RST Property 1438
RX Property 1439
SCK Property 1440
SCL Property 1441
SDA Property 1442
SPIDev Property 1443
TX Property 1444
UART Property 1445
IO.Objects.SimpleIO.Platforms 1446
BeagleBone Class 1447
BeagleBone Fields 1455
AIN0 Field 1462
AIN1 Field 1463

AIN2 Field 1464
AIN3 Field 1465
AIN4 Field 1466
AIN5 Field 1467
AIN6 Field 1468
EHRPWM0A Field 1469
EHRPWM0B Field 1470
EHRPWM1A Field 1471
EHRPWM1B Field 1472
EHRPWM2A Field 1473
EHRPWM2B Field 1474
GPIO10 Field 1475
GPIO11 Field 1476
GPIO110 Field 1477
GPIO111 Field 1478
GPIO112 Field 1479
GPIO113 Field 1480
GPIO115 Field 1481
GPIO117 Field 1482
GPIO12 Field 1483
GPIO13 Field 1484
GPIO14 Field 1485
GPIO15 Field 1486
GPIO2 Field 1487
GPIO20 Field 1488
GPIO22 Field 1489
GPIO23 Field 1490
GPIO26 Field 1491
GPIO27 Field 1492

GPIO3 Field 1493
GPIO30 Field 1494
GPIO31 Field 1495
GPIO32 Field 1496
GPIO33 Field 1497
GPIO34 Field 1498
GPIO35 Field 1499
GPIO36 Field 1500
GPIO37 Field 1501
GPIO38 Field 1502
GPIO39 Field 1503
GPIO4 Field 1504
GPIO44 Field 1505
GPIO45 Field 1506
GPIO46 Field 1507
GPIO47 Field 1508
GPIO48 Field 1509
GPIO49 Field 1510
GPIO5 Field 1511
GPIO50 Field 1512
GPIO51 Field 1513
GPIO60 Field 1514
GPIO61 Field 1515
GPIO62 Field 1516
GPIO63 Field 1517
GPIO65 Field 1518
GPIO66 Field 1519
GPIO67 Field 1520
GPIO68 Field 1521

GPIO69 Field 1522
GPIO7 Field 1523
GPIO70 Field 1524
GPIO71 Field 1525
GPIO72 Field 1526
GPIO73 Field 1527
GPIO74 Field 1528
GPIO75 Field 1529
GPIO76 Field 1530
GPIO77 Field 1531
GPIO78 Field 1532
GPIO79 Field 1533
GPIO8 Field 1534
GPIO80 Field 1535
GPIO81 Field 1536
GPIO86 Field 1537
GPIO87 Field 1538
GPIO88 Field 1539
GPIO89 Field 1540
GPIO9 Field 1541
I2C2 Field 1542
SPI0_0 Field 1543
SPI1_0 Field 1544
SPI1_1 Field 1545
MUNTS_0018 Class 1546
MUNTS_0018 Methods 1551
AnalogInputFactory Method 1553
ButtonInputFactory Method 1555
GPIOPinFactory Method 1556

I2CBusFactory Method 1558
LEDOutputFactory Method 1560
MotorOutputFactory Method 1561
ServoOutputFactory Method 1563
MUNTS_0018 Fields 1565
D1 Field 1568
J10A0 Field 1569
J10A1 Field 1570
J11A0 Field 1571
J11A1 Field 1572
J2PWM Field 1573
J3PWM Field 1574
J4D0 Field 1575
J4D1 Field 1576
J5D0 Field 1577
J5D1 Field 1578
J5I2C Field 1579
J6D0 Field 1580
J6D1 Field 1581
J6DIR Field 1582
J6PWM Field 1583
J7D0 Field 1584
J7D1 Field 1585
J7DIR Field 1586
J7PWM Field 1587
J9I2C Field 1588
SW1 Field 1589
PocketBeagle Class 1590
PocketBeagle Fields 1596

AIN0 Field 1601
AIN1 Field 1602
AIN2 Field 1603
AIN3 Field 1604
AIN4 Field 1605
AIN5 Field 1606
AIN6 Field 1607
AIN7 Field 1608
GPIO110 Field 1609
GPIO111 Field 1610
GPIO112 Field 1611
GPIO113 Field 1612
GPIO114 Field 1613
GPIO115 Field 1614
GPIO116 Field 1615
GPIO117 Field 1616
GPIO12 Field 1617
GPIO13 Field 1618
GPIO14 Field 1619
GPIO15 Field 1620
GPIO19 Field 1621
GPIO2 Field 1622
GPIO20 Field 1623
GPIO23 Field 1624
GPIO26 Field 1625
GPIO27 Field 1626
GPIO3 Field 1627
GPIO30 Field 1628
GPIO31 Field 1629

GPIO4 Field 1630
GPIO40 Field 1631
GPIO41 Field 1632
GPIO42 Field 1633
GPIO43 Field 1634
GPIO44 Field 1635
GPIO45 Field 1636
GPIO46 Field 1637
GPIO47 Field 1638
GPIO5 Field 1639
GPIO50 Field 1640
GPIO52 Field 1641
GPIO57 Field 1642
GPIO58 Field 1643
GPIO59 Field 1644
GPIO60 Field 1645
GPIO64 Field 1646
GPIO65 Field 1647
GPIO7 Field 1648
GPIO86 Field 1649
GPIO87 Field 1650
GPIO88 Field 1651
GPIO89 Field 1652
I2C1 Field 1653
I2C2 Field 1654
PWM0_0 Field 1655
PWM2_0 Figield 1656
SPI0_0 Field 1657
SPI1_1 Field 1658

RaspberryPi Class 1659
RaspberryPi Fields 1664
AIN0 Field 1669
AIN1 Field 1670
GPIO10 Field 1671
GPIO11 Field 1672
GPIO12 Field 1673
GPIO13 Field 1674
GPIO14 Field 1675
GPIO15 Field 1676
GPIO16 Field 1677
GPIO17 Field 1678
GPIO18 Field 1679
GPIO19 Field 1680
GPIO2 Field 1681
GPIO20 Field 1682
GPIO21 Field 1683
GPIO22 Field 1684
GPIO23 Field 1685
GPIO24 Field 1686
GPIO25 Field 1687
GPIO26 Field 1688
GPIO27 Field 1689
GPIO3 Field 1690
GPIO4 Field 1691
GPIO5 Field 1692
GPIO6 Field 1693
GPIO7 Field 1694
GPIO8 Field 1695

GPIO9 Field 1696
I2C1 Field 1697
I2C3 Field 1698
I2C4 Field 1699
I2C5 Field 1700
I2C6 Field 1701
PWM0_0 Field 1702
PWM0_1 Field 1703
SPI0_0 Field 1704
SPI0_1 Field 1705
SPI1_0 Field 1706
SPI1_1 Field 1707
SPI1_2 Field 1708
IO.Objects.SimpleIO.PWM 1709
Output Class 1710
Output Constructor 1712
Output Properties 1714
dutycycle Property 1715
fd Property 1716
IO.Objects.SimpleIO.Servo 1717
Output Class 1718
Output Constructor 1720
Output Properties 1722
fd Property 1723
position Property 1724
IO.Objects.SimpleIO.SPI 1725
Device Class 1726
Device Constructor 1728
Device Constructor (Designator, Int32, Int32, Int32, Pin) 1729

Device Constructor (String, Int32, Int32, Int32, Pin) 1731
Device Properties 1733
fd Property 1734
Device Methods 1735
Read Method 1736
Transaction Method 1738
Write Method 1740
Device Fields 1742
AUTOCHIPSELECT Field 1743
IO.Objects.SimpleIO.syslog 1744
Logger Class 1745
Logger Constructor 1749
Logger Methods 1751
Error Method 1753
Error Method (String) 1754
Error Method (String, Int32) 1755
Note Method 1757
Register Method 1758
Warning Method 1760
Write Method 1761
WriteLine Method 1762
Logger Fields 1763
LOG_ALERT Field 1766
LOG_AUTH Field 1767
LOG_CRIT Field 1768
LOG_DAEMON Field 1769
LOG_DEBUG Field 1770
LOG_EMERG Field 1771
LOG_ERR Field 1772

LOG_INFO Field 1773
LOG_LOCAL0 Field 1774
LOG_LOCAL1 Field 1775
LOG_LOCAL2 Field 1776
LOG_LOCAL3 Field 1777
LOG_LOCAL4 Field 1778
LOG_LOCAL5 Field 1779
LOG_LOCAL6 Field 1780
LOG_LOCAL7 Field 1781
LOG_MAIL Field 1782
LOG_NDELAY Field 1783
LOG_NOTICE Field 1784
LOG_ODELAY Field 1785
LOG_PERROR Field 1786
LOG_PID Field 1787
LOG_USER Field 1788
LOG_WARNING Field 1789
Logger.Registration Enumeration 1790
IO.Objects.SimpleIO.Watchdog 1791
Timer Class 1792
Timer Constructor 1794
Timer Properties 1795
fd Property 1796
timeout Property 1797
Timer Methods 1798
Kick Method 1799
Timer Fields 1800
DefaultDevice Field 1801
DefaultTimeout Field 1802