Package: Rduino (via r-universe)

August 24, 2024

Version 0.2

Date 2017-10-28

Title A Microcontroller Interface

Description Functions for connecting to and interfacing with an 'Arduino' or similar device. Functionality includes uploading of sketches, setting and reading digital and analog pins, and rudimentary servo control. This project is not affiliated with the 'Arduino' company, <https://www.arduino.cc/>.

Author Chris Suh, Peter Hoff

Maintainer Peter Hoff <peter.hoff@duke.edu>

RoxygenNote 6.0.1

License GPL-3

Depends serial

LazyData true

NeedsCompilation no

Date/Publication 2017-10-30 12:16:46 UTC

Repository https://pdhoff.r-universe.dev

RemoteUrl https://github.com/pdhoff/rduino

RemoteRef HEAD

RemoteSha 0814c8d3449d5c0119b5d61b1243057733bd8208

Contents

`BoardControlIno` *BoardControlIno*

Description

Board control file for the arduino and similar devices

`getApin` *Get analog pin*

Description

Get the value of an analog pin

Usage

```
getApin(pin)
```

Arguments

`pin`	the number of the pin to get (integer)

Value

the value of the pin.

Examples

```
## Not run:
rduinoConnect()
# set position of servo to position of potentiometer
off<-getDpin(4)
while (!off)
{
  angle<-getApin(5)
  angle<- 1.68 * angle + 575
  setServo(9,angle)
  off<-getDpin(4)
}
offServo()
```

```
rduinoClose()

## End(Not run)
```

---

getDpin *Get digital pin*

Description

Get the value of a digital pin

Usage

```
getDpin(pin)
```

Arguments

pin	the number of the pin to get (integer)

Value

the binary value of the pin.

Examples

```
## Not run:
rduinoConnect()
# LED remains on until button is pressed
setDpin(5,1)
isPressed<-getDpin(4)
while (!isPressed){ isPressed<-getDpin(4) }
setDpin(5,0)
rduinoClose()

## End(Not run)
```

---

offServo *Off servo*

Description

deactivate a servo

Usage

```
offServo()
```

offSignal *Off Signal*

Description

Turns off the square wave generated by onSignal

Usage

```
offSignal()
```

onServo *Set servo*

Description

Activate a servo and set a value

Usage

```
onServo(pin, value)
```

Arguments

pin	the number of the pin connected to the servo
value	value to set for the servo

Examples

```
## Not run:
rduinoConnect()
# set position of servo to position of potentiometer
off<-getDpin(4)
while (!off)
{
  angle<-getApin(5)
  angle<- 1.68 * angle + 575
  setServo(9,angle)
  off<-getDpin(4)
}
offServo()

rduinoClose()

## End(Not run)
```

onSignal *On Signal Generate a square wave, from 1Hz to 31.25kHz, with resolution of 16us*

Description

On Signal Generate a square wave, from 1Hz to 31.25kHz, with resolution of 16us

Usage

```
onSignal(freq, dutyCycle1, dutyCycle2)
```

Arguments

freq	the frequency of the wave, in Hz
dutyCycle1	the percentage of time that pin 9 should spend at HIGH
dutyCycle2	the percentage of time that pin 10 should spend at HIGH

rduinoClose *Rduino disconnect*

Description

Disconnect a previously connected Arduino or similar device

Usage

```
rduinoClose()
```

rduinoConnect *Rduino connect*

Description

Make a serial connection to an Arduino or similar device

Usage

```
rduinoConnect(baud = 38400, mode = "n,8,1", upload = FALSE,
  arduino = NULL, sdPin = 8)
```

Arguments

baud	baud rate
mode	communication mode
upload	if TRUE, upload the ino file to the device
arduino	command used to run arduino as a shell command including the path
	This function does two things - uploads a .ino file to an Arduino, and acts as a wrapper for the `serialConnection` function of the serial package. The options for the communication mode are available via the helpfile for the `serialConnection` command.

Examples

```
## Not run:
rduinoConnect()
rduinoClose()

## End(Not run)
```

---

rduinoSample *Rduino Sample*

Description

Samples in as quickly as possible from given pin for specified duration

Usage

```
rduinoSample(readPin, time)
```

Arguments

readPin	the analog pin to read in from
time	the length of time (ms) that the Arduino should read data for

setApin *Set analog pin*

Description

Set a analog pin to on or off

Usage

```
setApin(pin, value)
```

Arguments

pin	the number of the pin to set (integer)
value	the value to which to set the pin (real)

Examples

```
## Not run:
rduinoConnect()
# gradually increase intensity of LED
for (i in seq(1,256,by=5))
{
  setApin(11,i)
  Sys.sleep(0.05)
}
rduinoClose()

## End(Not run)
```

setDpin *Set digital pin*

Description

Set a digital pin to on or off

Usage

```
setDpin(pin, value)
```

Arguments

pin	the number of the pin to set (integer)
value	the value to which to set the pin (binary)

Examples

```
## Not run:
rduinoConnect()
# flash LED rapidly
for (i in 0:9)
{
  setDpin(8,1)
  Sys.sleep(0.05)
  setDpin(8,0)
  Sys.sleep(0.05)
}
rduinoClose()

## End(Not run)
```

Index