NAME

anxt-config − get information about the installed version of aNXT

SYNOPSIS

anxt-config [−−prefix] [−−exec−prefix] [−−version] [−−libs] [−−static-libs] [−−cflags]

DESCRIPTION

nxt-config is a tool that is used to configure to determine the compiler and linker flags that should be used to compile and link programs that use NXT.

AVAILABILITY

Linux

OPTIONS

anxt-config accepts the following options:

−−version
Print the currently installed version of NXT on the standard output.

−−libs Print the linker flags that are necessary to link a *aNXT* program.

−−libs-static
Print the linker flags for static libraries that are necessary to link a *aNXT* program.

−−cflags
Print the compiler flags that are necessary to compile a *aNXT* program.

−−prefix
Print the base directory of *aNXT* files.

−−exec−prefix
Print the executable directory of *aNXT* files.

EXAMPLES

```
gcc -o foobar foobar.c ‘anxt-config --cflags --libs‘
```

Commandline to compile a program with the aNXT library.

AUTHOR

g10 Code GmbH

SEE ALSO

libanxt(3)

NAME

nxt_beep − Play sound on the LEGO mindstorms NXT brick

SYNOPSIS

nxt_beep [*options* **]**

DESCRIPTION

Play sound on the NXT brick. *nxt_beep* will not wait, till the sound is played.

AVAILABILITY

Linux

OPTIONS

-d duration

Play the sound *duration* ms long. The default duration is 200.

-f frequency

Use *frequency* in Hertz to play the sound. The default frequency is 440.

-n nxtname

Use the NXT with name *nxtname* . The default is the first found brick.

Additionally, the bluetooth address of the NXT brick can be used as *nxtname* . You can get the bluetooth address and name of your USB connected NXT brick by using *nxt_info(1)*

EXIT STATUS

The following exit values shall be returned:

0 Successful completion.

>0 A error occured. If the error is caused by a problem of the NXT brick itself, a matching errorstring to the exit value can be displayed with the *nxt_error(1)* command.

EXAMPLES

```
nxt_beep -n 01:23:45:67:89:ab -d 1000
```

Connect to the NXT brick with bluetooth address "01:23:45:67:89:ab" via bluetooth and beep 1 second with 440 Hertz on it.

CAVEATS

You can not get automatically access to the NXT brick.

Either you need access rights to the NXT usb device. Use root rights or see *nxt_udev(8)* for more information.

Or you need to pair the bluetooth devices of the host computer and the NXT brick. There are several programs to do this, one is "kbluetoothd".

AUTHOR

Janosch Graef

SEE ALSO

libanxt(3), **nxt_error**(1), **nxt_udev**(8), **nxt_info**(1)

NAME

nxt_calibrate − calibrates a LEGO mindstorms NXT sensor

SYNOPSIS

nxt_calibrate [*options* **]**

DESCRIPTION

calibrates a LEGO mindstorms NXT sensor.
The *nxt_calibrate* program will ask you first to produce the lowest input and then to press enter to continue.
In the second step it will ask you to produce the highest input and then to press enter to continue.
The result is stored on the NXT brick filesystem on files with the extension *.cal*
Currently implemented are the LEGO light and the LEGO sound sensor.
The filename for the light sensor is *Light Sensor.cal*
The filename for the sound sensor is *Sound Sensor.cal*

AVAILABILITY

Linux

OPTIONS

-n nxtname

Use the NXT with name *nxtname* . The default is the first found brick.

Additionally, the bluetooth address of the NXT brick can be used as *nxtname* . You can get the bluetooth address and name of your USB connected NXT brick by using *nxt_info(1)*

-s sensor

The *sensor* values are: 1, 2, 3, 4. If the -s option is missing, but the -t option is used, the value is depends on the type/sensorportnumber combinations, defined by Lego itself.

Type	Sensorport
light	3
sound	2

-t type The sensor type defines, how to interpret the values from a sensor.
Valid sensor types for *nxt_calibrate* are:
light
sound

EXIT STATUS

The following exit values shall be returned:

0 Successful completion.

>0 A error occured. If the error is caused by a problem of the NXT brick itself, a matching errorstring to the exit value can be displayed with the *nxt_error(1)* command.

EXAMPLES

```
nxt_calibrate -n 01:23:45:67:89:ab -t light
```

Connect to the NXT brick with bluetooth address "01:23:45:67:89:ab" via bluetooth, calibrates the light-sensor at port 2 and write the resulting *Light Sensor.cal* file to the NXT brick filesystem.

CAVEATS

You can not get automatically access to the NXT brick.

Either you need access rights to the NXT usb device. Use root rights or see *nxt_udev(8)* for more information.

Or you need to pair the bluetooth devices of the host computer and the NXT brick. There are several programs to do this, one is "kbluetoothd".

AUTHOR

Janosch Graef

SEE ALSO

libanxt(3), **nxt_list**(1) **nxt_error**(1), **nxt_udev**(8), **nxt_info**(1)

NAME

nxt_delflash − Remove all flash data on the LEGO mindstorms NXT brick

SYNOPSIS

nxt_delflash [*options* **]**

DESCRIPTION

Remove all flash data on a NXT brick including configuration data and the files on the NXT filesystem (execpt the file NVConfig.sys).

AVAILABILITY

Linux

OPTIONS

-f Do not ask for permission before deleting data.

-n nxtname

Use the NXT with name *nxtname* . The default is the first found brick.

Additionally, the bluetooth address of the NXT brick can be used as *nxtname* . You can get the bluetooth address and name of your USB connected NXT brick by using *nxt_info(1)*

EXIT STATUS

The following exit values shall be returned:

0 Successful completion.

>0 A error occured. If the error is caused by a problem of the NXT brick itself, a matching errorstring to the exit value can be displayed with the *nxt_error(1)* command.

EXAMPLES

nxt_delflash -n 01:23:45:67:89:ab -f

Connect to the NXT brick with bluetooth address "01:23:45:67:89:ab" via bluetooth and remove all flash data on this brick without asking for permission.

CAVEATS

You can not get automatically access to the NXT brick.

Either you need access rights to the NXT usb device. Use root rights or see *nxt_udev(8)* for more information.

Or you need to pair the bluetooth devices of the host computer and the NXT brick. There are several programs to do this, one is "kbluetoothd".

AUTHOR

Janosch Graef

SEE ALSO

libanxt(3), **nxt_error**(1), **nxt_udev**(8), **nxt_info**(1)

NAME

nxt_download − Download a file from the LEGO mindstorms NXT brick

SYNOPSIS

nxt_download [*options* **]** input [output]

DESCRIPTION

Download a file from the NXT brick to the host computer. If the name of the outputfile on the hostcomputer (*output*) is not given, the name of the inputfile on the NXT brick (*input*) is used.

AVAILABILITY

Linux

OPTIONS

-n nxtname

Use the NXT with name *nxtname* . The default is the first found brick.

Additionally, the bluetooth address of the NXT brick can be used as *nxtname* . You can get the bluetooth address and name of your USB connected NXT brick by using *nxt_info(1)*

EXIT STATUS

The following exit values shall be returned:

0 Successful completion.

>0 A error occured. If the error is cause by a problem of the NXT brick itself, a matching errorstring to the exit value can be displayed with the *nxt_error(1)* command.

EXAMPLES

nxt_download -n 01:23:45:67:89:ab test.rxe

Connect to the NXT brick with bluetooth address "01:23:45:67:89:ab" via bluetooth and download the file test.rxe from the NXT brick to the host computer as test.rxe.

CAVEATS

You can not get automatically access to the NXT brick.

Either you need access rights to the NXT usb device. Use root rights or see *nxt_udev(8)* for more information.

Or you need to pair the bluetooth devices of the host computer and the NXT brick. There are several programs to do this, one is "kbluetoothd".

AUTHOR

Janosch Graef

SEE ALSO

libanxt(3), **nxt_error**(1), **nxt_udev**(8), **nxt_info**(1)

NAME

nxt_error − Display error of NXT brick matching to errornumber

SYNOPSIS

nxt_error *errornumber*

DESCRIPTION

Display error of LEGO mindstroms NXT brick matching to errornumber. *nxtlib* shell commands return integer errors as return code. *nxt_error* can be used to get the matching errorstring.

AVAILABILITY

Linux

EXAMPLES

nxt_error 192

This command display the string "Data contains out-of-range values"

NAME

nxt_getprog − Get name of currently running program on the NXT brick

SYNOPSIS

nxt_getprog [*options* **]**

DESCRIPTION

Display the name of currently running program on the LEGO mindstorms NXT brick.

AVAILABILITY

Linux

OPTIONS

-n nxtname

Use the NXT with name *nxtname* . The default is the first found brick.

Additionally, the bluetooth address of the NXT brick can be used as *nxtname* . You can get the bluetooth address and name of your USB connected NXT brick by using *nxt_info(1)*

EXIT STATUS

The following exit values shall be returned:

0 Successful completion.

>0 A error occured. If the error is caused by a problem of the NXT brick itself, a matching errorstring to the exit value can be displayed with the *nxt_error(1)* command.

EXAMPLES

nxt_getprog -n 01:23:45:67:89:ab

Connect to the NXT brick with bluetooth address "01:23:45:67:89:ab" via bluetooth and print the name of the currently running program. host computer as test.rxe.

CAVEATS

You can not get automatically access to the NXT brick.

Either you need access rights to the NXT usb device. Use root rights or see *nxt_udev(8)* for more information.

Or you need to pair the bluetooth devices of the host computer and the NXT brick. There are several programs to do this, one is "kbluetoothd".

AUTHOR

Janosch Graef

SEE ALSO

libanxt(3), **nxt_error**(1), **nxt_udev**(8), **nxt_info**(1)

NAME

nxt_info − Show information about LEGO mindstorms NXT brick

SYNOPSIS

nxt_info [*options* **]**

DESCRIPTION

Show information about a LEGO mindstorms NXT brick. The information includes the name of the brick, its battery status, the sound volume setting, the bluetooth address, the bluetooth strength, the amount of free flash, the firmware version and the protocol version.

AVAILABILITY

Linux

OPTIONS

-n nxtname

Use the NXT with name *nxtname* . The default is the first found brick.

Additionally, the bluetooth address of the NXT brick can be used as *nxtname* . You can get the bluetooth address and name of your USB connected NXT brick by using *nxt_info(1)*

EXIT STATUS

The following exit values shall be returned:

0 Successful completion.

>0 A error occured. If the error is caused by a problem of the NXT brick itself, a matching errorstring to the exit value can be displayed with the *nxt_error(1)* command.

EXAMPLES

nxt_info -n 01:23:45:67:89:ab

Connect to the NXT brick with bluetooth address "01:23:45:67:89:ab" via bluetooth and print informations like the following:

```
Name:                nxt
Battery:             8294mV
Volume:              4
Bluetooth Address:   01:23:45:67:89:ab
Bluetooth Strength:  0%
Free flash:          31657984 bytes
Firmware version:    1.5
Protocol version:    1.124
```

CAVEATS

You can not get automatically access to the NXT brick.

Either you need access rights to the NXT usb device. Use root rights or see *nxt_udev(8)* for more information.

Or you need to pair the bluetooth devices of the host computer and the NXT brick. There are several programs to do this, one is "kbluetoothd".

AUTHOR

Janosch Graef

SEE ALSO

libanxt(3), **nxt_error**(1), **nxt_udev**(8), **nxt_info**(1)

NAME

nxt_list − List files on a LEGO mindstorms NXT brick

SYNOPSIS

nxt_list [*options* **]** *wildcard*

DESCRIPTION

List files of the filesystem on a LEGO mindstorms NXT brick. The only valid wildcard character is "*", it can be used to represent the whole filename or extension, not parts of it.

AVAILABILITY

Linux

OPTIONS

-a Show all files (also files starting with ".")

-n nxtname

Use the NXT with name *nxtname* . The default is the first found brick.

Additionally, the bluetooth address of the NXT brick can be used as *nxtname* . You can get the bluetooth address and name of your USB connected NXT brick by using *nxt_info(1)*

-s Also show filesizes

EXIT STATUS

The following exit values shall be returned:

0 Successful completion.

>0 A error occured. If the error is cause by a problem of the NXT brick itself, a matching errorstring to the exit value can be displayed with the *nxt_error(1)* command.

EXAMPLES

```
nxt_list -s -n 01:23:45:67:89:ab "*.*"
```

Connect to the NXT brick with bluetooth address "01:23:45:67:89:ab" via bluetooth and print informations like the following:

```
runtest.rxe        594
NVConfig.sys       1
```

CAVEATS

You can not get automatically access to the NXT brick.

Either you need access rights to the NXT usb device. Use root rights or see *nxt_udev(8)* for more information.

Or you need to pair the bluetooth devices of the host computer and the NXT brick. There are several programs to do this, one is "kbluetoothd".

AUTHOR

Janosch Graef

SEE ALSO

libanxt(3), **nxt_error**(1), **nxt_udev**(8), **nxt_info**(1)

NAME

nxt_lsmod − list the modules of a LEGO mindstorms NXT brick

SYNOPSIS

nxt_lsmod [*options* **]** *wildcard*

DESCRIPTION

List the modules of a LEGO mindstorms NXT brick. The only valid wildcard character is "*", it can be used to represent the whole filename or extension, not parts of it. The *nxt_lsmod* command lists in one line: the name of the module, the module id and the IO-Map size.

AVAILABILITY

Linux

OPTIONS

-n nxtname

Use the NXT with name *nxtname* . The default is the first found brick.

Additionally, the bluetooth address of the NXT brick can be used as *nxtname* . You can get the bluetooth address and name of your USB connected NXT brick by using *nxt_info(1)*

EXIT STATUS

The following exit values shall be returned:

0 Successful completion.

>0 A error occured. If the error is caused by a problem of the NXT brick itself, a matching errorstring to the exit value can be displayed with the *nxt_error(1)* command.

EXAMPLES

```
nxt_lsmod -s -n 01:23:45:67:89:ab "*.*"
```

Connect to the NXT brick with bluetooth address "01:23:45:67:89:ab" via bluetooth and print informations like the following:

```
Comm.mod         00050001   0   1896
Input.mod        00030001   0   80
Button.mod       00040001   0   36
Display.mod      000a0001   0   1720
Loader.mod       00090001   0   8
Low Speed.mod    000b0001   0   167
Output.mod       00020001   0   100
Sound.mod        00080001   0   30
IOCtrl.mod       00060001   0   2
Command.mod      00010001   0   32820
Ui.mod           000c0001   0   40
```

CAVEATS

You can not get automatically access to the NXT brick.

Either you need access rights to the NXT usb device. Use root rights or see *nxt_udev(8)* for more information.

Or you need to pair the bluetooth devices of the host computer and the NXT brick. There are several programs to do this, one is "kbluetoothd".

AUTHOR

Janosch Graef

SEE ALSO

libanxt(3), **nxt_error**(1), **nxt_udev**(8), **nxt_info**(1)

NAME

nxt_motor − Control a LEGO mindstorms NXT motor

SYNOPSIS

nxt_motor [*options* **]**

DESCRIPTION

Control a LEGO mindstorms NXT motor.

When used without arguments, *nxt_motor* will run infinite with the lowest power. It is possible to set the power, set a rotation limit, brake or block the motor with commandline arguments.

AVAILABILITY

Linux

OPTIONS

-b
: Brake (block) motor.

-m motor
: Valid *motor* values are: A, B, C, ABC for motors at the ports A, B or C or all ports (ABC). The default value is A.

-n nxtname
: Use the NXT with name *nxtname* . The default is the first found brick.

  Additionally, the bluetooth address of the NXT brick can be used as *nxtname* . You can get the bluetooth address and name of your USB connected NXT brick by using *nxt_info(1)*

-p power
: The *power* setting controls the power/speed of the motor. The default valid is 50, which means, that the motor is barly moving.

-r rotation
: The *rotation* setting limits the movement of the motor. If the tacho value of the motor (see *nxt_tacho(1)*) reaches *rotation* the motor is stopped (or depending on the options blocked).
  The default *rotation* is 0, which means unlimited movement.

-s
: Stop (coast) motor. Same as -p 0

-y
: Synchronise motor with another motor.
  Be carefull when using motor synchronisation without using the brake *-b* setting.
  Useing motor synchronisation without the brake setting and without a slowing external force on the motors can easily result in a wild oscillating movement of the motors.

-i
: Idle motor

EXIT STATUS

The following exit values shall be returned:

0
: Successful completion.

>0
: A error occured. If the error is caused by a problem of the NXT brick itself, a matching errorstring to the exit value can be displayed with the *nxt_error(1)* command.

EXAMPLES

```
nxt_motor -n 01:23:45:67:89:ab -r 1000 -p 55
```

Connect to the NXT brick with bluetooth address "01:23:45:67:89:ab" via bluetooth and rotate motor A with power 55 (very weak/slow) till the tacho value reaches 1000.

CAVEATS

You can not get automatically access to the NXT brick.

Either you need access rights to the NXT usb device. Use root rights or see *nxt_udev(8)* for more information.

Or you need to pair the bluetooth devices of the host computer and the NXT brick. There are several programs to do this, one is "kbluetoothd".

AUTHOR

Janosch Graef

SEE ALSO

libanxt(3), **nxt_tacho**(1), **nxt_error**(1), **nxt_udev**(8), **nxt_info**(1)

NAME

nxt_motor_playback − Move a NXT motor based on previous recorded values

SYNOPSIS

nxt_motor_playback [*options* **]**

DESCRIPTION

Move a LEGO mindstorms NXT motor based on previous recorded values from standard input.
The format of the input is line based. The time in seconds till the start of the movement and the difference of the tacho value till start is in each line.

AVAILABILITY

Linux

OPTIONS

-m motor

Valid *motor* values are: A, B, C, ABC for motors at the ports A, B or C or all ports (ABC). The default value is A.

-n nxtname

Use the NXT with name *nxtname* . The default is the first found brick.

Additionally, the bluetooth address of the NXT brick can be used as *nxtname* . You can get the bluetooth address and name of your USB connected NXT brick by using *nxt_info(1)*

-p power

The *power* setting controls the power/speed of the motor. The default valid is 58, which means, that the motor is moving rather slowly.

-s Stop (coast) motor after playback.
The default is to brake (block) the motor.

EXIT STATUS

The following exit values shall be returned:

0 Successful completion.

>0 A error occured. If the error is caused by a problem of the NXT brick itself, a matching errorstring to the exit value can be displayed with the *nxt_error(1)* command.

EXAMPLES

```
nxt_motor_playback -n 01:23:45:67:89:ab -p 55 << EOT
0.074935 -16
0.096921 -20
EOT
```

Connect to the NXT brick with bluetooth address "01:23:45:67:89:ab" via bluetooth and try to the rotate motor A with power 55 (very weak/slow) from the tacho value -16 to the tacho value -20 after 0.096921 seconds.

CAVEATS

You can not get automatically access to the NXT brick.

Either you need access rights to the NXT usb device. Use root rights or see *nxt_udev(8)* for more information.

Or you need to pair the bluetooth devices of the host computer and the NXT brick. There are several programs to do this, one is "kbluetoothd".

AUTHOR

Janosch Graef

SEE ALSO

libanxt(3), **nxt_motor_record**(1), **nxt_error**(1), **nxt_udev**(8), **nxt_info**(1)

NAME

nxt_motor_record − Record the movement of a LEGO mindstorms NXT motor

SYNOPSIS

nxt_motor_record [*options* **]**

DESCRIPTION

Record the movement of a LEGO mindstorms NXT motor and print it on standard output. The format of the output is line based. In each line, the time in seconds till the start of recording and the change of the tacho value till start is printed. The start and end of recording is signaled with a sound similar to the output of *nxt_beep(1)* with option "-d 1".

AVAILABILITY

Linux

OPTIONS

-m motor

Valid *motor* values are: A, B, C, ABC for motors at the ports A, B or C or all ports (ABC). The default value is A.

-n nxtname

Use the NXT with name *nxtname* . The default is the first found brick.

Additionally, the bluetooth address of the NXT brick can be used as *nxtname* . You can get the bluetooth address and name of your USB connected NXT brick by using *nxt_info(1)*

-t duration

The *duration* value controls the time till the recording is stopped.

EXIT STATUS

The following exit values shall be returned:

0 Successful completion.

>0 A error occured. If the error is caused by a problem of the NXT brick itself, a matching errorstring to the exit value can be displayed with the *nxt_error(1)* command.

EXAMPLES

```
nxt_motor_record -n 01:23:45:67:89:ab -t 0.15
```

Connect to the NXT brick with bluetooth address "01:23:45:67:89:ab" via bluetooth, record the movement of motor A 0.15 seconds long and output something like the following:

```
0.074935 -16
0.096921 -20
```

CAVEATS

You can not get automatically access to the NXT brick.

Either you need access rights to the NXT usb device. Use root rights or see *nxt_udev(8)* for more information.

Or you need to pair the bluetooth devices of the host computer and the NXT brick. There are several programs to do this, one is "kbluetoothd".

AUTHOR

Janosch Graef

SEE ALSO

libanxt(3), **nxt_tacho**(1), **nxt_beep**(1), **nxt_error**(1), **nxt_udev**(8), **nxt_info**(1)

NAME

nxt_motor_travel − Control/synchronise two LEGO mindstorms NXT motors

SYNOPSIS

nxt_motor_travel [*options* **]**

DESCRIPTION

Control/synchronise LEGO mindstorms NXT motors.
This is usefull when the two motors are used for mobile wheeldriven robots.
When both motors run similar, the synchronisation will take effekt. In case one motor would be slowed down by a force, the second motor will move similar. This makes it more easy to keep direction, when driving forward.
When synchronisation is in effect, the motors have deliver more power as without synchronisation.
There is also a *turnratio* setting, to drive curves. If turnratio is not zero, there will be no synchronisation effect, when one motor would be slowed down.
Using nxt_travel is not exactly the same as using *nxt_motor* with the *-y* (synchronisation) option.
nxt_travel is similar to using *nxt_motor* with the *-y -b* (synchronisation and brake/block) options.
When used without arguments, *nxt_motor_travel* will run infinite with a low power setting. It is possible to set the power, set a rotation limit, brake or block the motor with commandline arguments.

AVAILABILITY

Linux

OPTIONS

-b Brake (block) motors.

-m motors
Valid *motors* values are: AB, AC, BC for motors at two of the ports A, B or C. The default value is BC.

-n nxtname
Use the NXT with name *nxtname* . The default is the first found brick.

Additionally, the bluetooth address of the NXT brick can be used as *nxtname* . You can get the bluetooth address and name of your USB connected NXT brick by using *nxt_info(1)*

-p power
The *power* setting controls the power/speed of the motor. The default valid is 50, which means, that the motor is barly moving in curves.

-r rotation
The *rotation* setting limits the movement of the motor. If the tacho value of the motor (see *nxt_tacho(1)*) reaches *rotation* the motor is stopped (or depending on the options blocked).
The default *rotation* is 0, which means unlimited movement.

-s Stop (coast) motor. Same as -p 0 Set turnratio.
The turnratio defines the curve radius. The default 0 means no curve. The -100/100 turnratio means, that one motor is moving forward and the the other motor is moving backward in the same way.

-i Idle motor

EXIT STATUS

The following exit values shall be returned:

0 Successful completion.

>0 A error occured. If the error is caused by a problem of the NXT brick itself, a matching errorstring to the exit value can be displayed with the *nxt_error(1)* command.

EXAMPLES

```
nxt_motor_travel -n 01:23:45:67:89:ab -p 55
```

Connect to the NXT brick with bluetooth address "01:23:45:67:89:ab" via bluetooth and rotate motors BC with power 55.

CAVEATS

You can not get automatically access to the NXT brick.

Either you need access rights to the NXT usb device. Use root rights or see *nxt_udev(8)* for more information.

Or you need to pair the bluetooth devices of the host computer and the NXT brick. There are several programs to do this, one is "kbluetoothd".

AUTHOR

Janosch Graef

SEE ALSO

libanxt(3), **nxt_motor(1), nxt_tacho**(1), **nxt_error**(1), **nxt_udev**(8), **nxt_info**(1)

NAME

nxt_pilot − control a NXT brick without touching it

SYNOPSIS

nxt_pilot [*options* **]**

DESCRIPTION

nxt_pilot is a program to control a Lego Mindstorm NXT brick without touching it. .br The *nxt_pilot* program displays a window with a image of a NXT brick. Pressing the buttons on the image of the *nxt_pilot* window cause the same reaction as pressing the similar buttons on the NXT brick.
It requires the windowmanager to quit the nxt_pilot program (usually by using a icon on the window frame).

AVAILABILITY

Linux

OPTIONS

-n nxtname

Use the NXT with name *nxtname* . The default is the first found brick.

Additionally, the bluetooth address of the NXT brick can be used as *nxtname* . You can get the bluetooth address and name of your USB connected NXT brick by using *nxt_info(1)*

EXIT STATUS

The following exit values shall be returned:

0 Successful completion.

>0 A error occured. If the error is caused by a problem of the NXT brick itself, a matching errorstring to the exit value can be displayed with the *nxt_error(1)* command.

EXAMPLES

```
nxt_pilot -n 01:23:45:67:89:ab
```

Connect to the NXT brick with bluetooth address "01:23:45:67:89:ab" via bluetooth and show the *nxt_pilot* window.

CAVEATS

You can not get automatically access to the NXT brick.

Either you need access rights to the NXT usb device. Use root rights or see *nxt_udev(8)* for more information.

Or you need to pair the bluetooth devices of the host computer and the NXT brick. There are several programs to do this, one is "kbluetoothd".

AUTHOR

Janosch Graef

SEE ALSO

libanxt(3), **nxt_error**(1), **nxt_udev**(8), **nxt_info**(1)

NAME

nxt_pollcmd − Poll data from a LEGO mindstorms NXT brick

SYNOPSIS

nxt_pollcmd [*options* **]**

DESCRIPTION

Poll data from a LEGO mindstorms NXT brick.

AVAILABILITY

Linux

OPTIONS

-b buffer

Selects the used buffer.
Valid buffers are: "poll" and "highspeed".
The default is "poll".

-n nxtname

Use the NXT with name *nxtname* . The default is the first found brick.

Additionally, the bluetooth address of the NXT brick can be used as *nxtname* . You can get the bluetooth address and name of your USB connected NXT brick by using *nxt_info(1)*

EXIT STATUS

The following exit values shall be returned:

0 Successful completion.

>0 A error occured. If the error is caused by a problem of the NXT brick itself, a matching errorstring to the exit value can be displayed with the *nxt_error(1)* command.

EXAMPLES

```
nxt_pollcmd -n 01:23:45:67:89:ab
```

Connect to the NXT brick with bluetooth address "01:23:45:67:89:ab" via bluetooth and poll data from NXT brick.

CAVEATS

You can not get automatically access to the NXT brick.

Either you need access rights to the NXT usb device. Use root rights or see *nxt_udev(8)* for more information.

Or you need to pair the bluetooth devices of the host computer and the NXT brick. There are several programs to do this, one is "kbluetoothd".

AUTHOR

Janosch Graef

SEE ALSO

libanxt(3), **nxt_error**(1), **nxt_udev**(8), **nxt_info**(1)

NAME

nxt_recv − Receives a message from a LEGO mindstorms NXT brick

SYNOPSIS

nxt_recv [*options* **]**

DESCRIPTION

Receives a message from a LEGO mindstorms NXT brick.

AVAILABILITY

Linux

OPTIONS

-c Clear mailbox after receiving from it

-m mailbox

Selects the used mailbox.
mailbox is a number between 1 and 10.
The default is 1.

-n nxtname

Use the NXT with name *nxtname* . The default is the first found brick.

Additionally, the bluetooth address of the NXT brick can be used as *nxtname* . You can get the bluetooth address and name of your USB connected NXT brick by using *nxt_info(1)*

-v Verbose mode

EXIT STATUS

The following exit values shall be returned:

0 Successful completion.

>0 A error occured. If the error is caused by a problem of the NXT brick itself, a matching errorstring to the exit value can be displayed with the *nxt_error(1)* command.

EXAMPLES

```
nxt_recv -n 01:23:45:67:89:ab
```

Connect to the NXT brick with bluetooth address "01:23:45:67:89:ab" via bluetooth and receive data from mailbox 1 of NXT brick.

CAVEATS

You can not get automatically access to the NXT brick.

Either you need access rights to the NXT usb device. Use root rights or see *nxt_udev(8)* for more information.

Or you need to pair the bluetooth devices of the host computer and the NXT brick. There are several programs to do this, one is "kbluetoothd".

AUTHOR

Janosch Graef

SEE ALSO

libanxt(3), **nxt_send**(1), **nxt_error**(1), **nxt_udev**(8), **nxt_info**(1)

NAME

nxt_remove − remove files from the LEGO mindstorms NXT brick

SYNOPSIS

nxt_remove [*options* **]** file ...

DESCRIPTION

Remove files from the NXT brick.

AVAILABILITY

Linux

OPTIONS

-n nxtname

Use the NXT with name *nxtname* . The default is the first found brick.

Additionally, the bluetooth address of the NXT brick can be used as *nxtname* . You can get the bluetooth address and name of your USB connected NXT brick by using *nxt_info(1)*

EXIT STATUS

The following exit values shall be returned:

0 Successful completion.

>0 A error occured. If the error is caused by a problem of the NXT brick itself, a matching errorstring to the exit value can be displayed with the *nxt_error(1)* command.

EXAMPLES

nxt_remove -n 01:23:45:67:89:ab test.rxe

Connect to the NXT brick with bluetooth address "01:23:45:67:89:ab" via bluetooth and remove the file test.rxe from the NXT brick to the host computer.

CAVEATS

You can not get automatically access to the NXT brick.

Either you need access rights to the NXT usb device. Use root rights or see *nxt_udev(8)* for more information.

Or you need to pair the bluetooth devices of the host computer and the NXT brick. There are several programs to do this, one is "kbluetoothd".

AUTHOR

Janosch Graef

SEE ALSO

libanxt(3), **nxt_error**(1), **nxt_udev**(8), **nxt_info**(1)

NAME

nxt_resetbt − reset the bluetooth connection to a NXT brick

SYNOPSIS

nxt_resetbt [*options* **]**

DESCRIPTION

Reset the bluetooth connection to a Lego mindstorms NXT brick.

AVAILABILITY

Linux

OPTIONS

-n nxtname

Use the NXT with name *nxtname* . The default is the first found brick.

Additionally, the bluetooth address of the NXT brick can be used as *nxtname* . You can get the bluetooth address and name of your USB connected NXT brick by using *nxt_info(1)*

EXIT STATUS

The following exit values shall be returned:

0 Successful completion.

>0 A error occured. If the error is caused by a problem of the NXT brick itself, a matching errorstring to the exit value can be displayed with the *nxt_error(1)* command.

EXAMPLES

```
nxt_resetbt -n 01:23:45:67:89:ab
```

Connect to the NXT brick with bluetooth address "01:23:45:67:89:ab" via bluetooth and reset the bluetooth stack.

CAVEATS

You can not get automatically access to the NXT brick.

Either you need access rights to the NXT usb device. Use root rights or see *nxt_udev(8)* for more information.

Or you need to pair the bluetooth devices of the host computer and the NXT brick. There are several programs to do this, one is "kbluetoothd".

AUTHOR

Janosch Graef

SEE ALSO

libanxt(3), **nxt_error**(1), **nxt_udev**(8), **nxt_info**(1)

NAME

nxt_ricc − convert between bitmap and robot picture graphics files

SYNOPSIS

nxt_ricc [*options* **]** file

DESCRIPTION

Ric converter: Convert between bitmap images files (eg. jpeg or png format) and robot picture graphics files.
Robot picture graphics files (extension ".ric") are usually used on the Lego Mindstorms NXT brick.

AVAILABILITY

Linux

OPTIONS

-f outputformat

Set output format.
This option is only required for RIC to PNG/JPEG conversion.
Valid values for *outputformat* are png, jpg and gif.

-i Invert image.
Convert white pixels to black and vise versa.

-o outputfile

Set the name of output file.
The usage of this option is recommended, cause the default *outputfile* is made from name of the inputfile (including extension) and the extension of the target format.

-q quality

Set quality.
This option is only usefull when saving to JPEG files.
The default *quality* is -1, which means the best quality/memory ratio. files)

-v Verbose mode

EXAMPLES

nxt_ricc test.png

Convert the inputfile *test.png* to a robot picture graphics file named test.png.ric

AUTHOR

Janosch Graef

NAME

nxt_rmdc − convert between midi and robot melody files

SYNOPSIS

nxt_rmdc [*options* **]** file

DESCRIPTION

Rmd converter: Convert between midi sound files and robot melody files.
Robot melody files (extension ".rmd") are usually used on the Lego Mindstorms NXT brick.
Robot melody files hold information about notes, that is frequency and duration.

AVAILABILITY

Linux

OPTIONS

-o outputfile

Set the name of output file.
The usage of this option is recommended, cause the default *outputfile* is made from name of the inputfile (including extension) and the extension of the target format.

-v Verbose mode

EXAMPLES

nxt_rmdc test.mid

Convert the inputfile *test.mid* to a robot melody file named test.mid.rmd

AUTHOR

Janosch Graef

NAME

nxt_rsoc − convert between wav and robot sound files

SYNOPSIS

nxt_rsoc [*options* **]** file

DESCRIPTION

Rso converter: Convert between wav sound files and robot melody files.
Robot sound files (extension ".rso") are usually used on the Lego Mindstorms NXT brick.
Robot sound files hold sound data containing waveform information.

AVAILABILITY

Linux

OPTIONS

-o outputfile
: Set the name of output file.
The usage of this option is recommended, cause the default *outputfile* is made from name of the inputfile (including extension) and the extension of the target format.

-v Verbose mode

EXAMPLES

nxt_rsoc test.wav

Convert the inputfile *test.wav* to a robot sound file named test.wav.rso

AUTHOR

Janosch Graef

NAME

nxt_run − run a executable file on the LEGO mindstorms NXT brick

SYNOPSIS

nxt_run [*options* **]** program

DESCRIPTION

Start a executable file from the NXT brick filesystem on the NXT brick.

AVAILABILITY

Linux

OPTIONS

-n nxtname

Use the NXT with name *nxtname* . The default is the first found brick.

Additionally, the bluetooth address of the NXT brick can be used as *nxtname* . You can get the bluetooth address and name of your USB connected NXT brick by using *nxt_info(1)*

EXIT STATUS

The following exit values shall be returned:

0 Successful completion.

>0 A error occured. If the error is caused by a problem of the NXT brick itself, a matching errorstring to the exit value can be displayed with the *nxt_error(1)* command.

EXAMPLES

nxt_run -n 01:23:45:67:89:ab test.rxe

Connect to the NXT brick with bluetooth address "01:23:45:67:89:ab" via bluetooth and run the file test.rxe on the NXT brick.

CAVEATS

You can not get automatically access to the NXT brick.

Either you need access rights to the NXT usb device. Use root rights or see *nxt_udev(8)* for more information.

Or you need to pair the bluetooth devices of the host computer and the NXT brick. There are several programs to do this, one is "kbluetoothd".

AUTHOR

Janosch Graef

SEE ALSO

libanxt(3), **nxt_up_run**(1), **nxt_error**(1), **nxt_udev**(8), **nxt_info**(1)

NAME

nxt_scan − scan for NXT bricks connected via USB or paired bluetooth

SYNOPSIS

nxt_scan

DESCRIPTION

Scan for Lego Mindstorms NXT bricks and print their name on standart output. The bricks can be either connected via USB or paired bluetooth.

AVAILABILITY

Linux

EXIT STATUS

The following exit values shall be returned:

0 Successful completion.

>0 A error occured. If the error is caused by a problem of the NXT brick itself, a matching errorstring to the exit value can be displayed with the *nxt_error(1)* command.

EXAMPLES

nxt_scan

Scan the USB bus and the paired bluetooth connections for Lego Mindstorms NXT devices. A typical output can be:

```
NXT
NXT
```

CAVEATS

If a brick is connected both via USB and bluetooth, the device is reported twice.

You can not get automatically access to the NXT brick.

Either you need access rights to the NXT usb device. Use root rights or see *nxt_udev(8)* for more information.

Or you need to pair the bluetooth devices of the host computer and the NXT brick. There are several programs to do this, one is "kbluetoothd".

AUTHOR

Janosch Graef

SEE ALSO

libanxt(3), **nxt_error**(1), **nxt_udev**(8), **nxt_info**(1)

NAME

nxt_screenshot − Take a picture of the display of a NXT brick

SYNOPSIS

nxt_screenshot [*options* **]**

DESCRIPTION

Take a picture of the display of a LEGO mindstorms NXT brick and store it as a bitmap file on the host computer.

AVAILABILITY

Linux

OPTIONS

-f format

Select file format of the saved image.
Valid fileformats are "png" and "jpeg". The default fileformat is "png".

-n nxtname

Use the NXT with name *nxtname* . The default is the first found brick.

Additionally, the bluetooth address of the NXT brick can be used as *nxtname* . You can get the bluetooth address and name of your USB connected NXT brick by using *nxt_info(1)*

-o output

Name of the output file. The default name of the output file is "screen.png" in case of the png file format and "screen.jpg" in case of the jpeg file format.

-t Produce a picture with transparency. The jpeg file format do not support transparency, therefore this works only with the png file format.

EXIT STATUS

The following exit values shall be returned:

0 Successful completion.

>0 A error occured. If the error is caused by a problem of the NXT brick itself, a matching errorstring to the exit value can be displayed with the *nxt_error(1)* command.

EXAMPLES

nxt_screenshot -n 01:23:45:67:89:ab -d 1000

Connect to the NXT brick with bluetooth address "01:23:45:67:89:ab" via bluetooth and screenshot 1 second with 440 Hertz on it.

CAVEATS

You can not get automatically access to the NXT brick.

Either you need access rights to the NXT usb device. Use root rights or see *nxt_udev(8)* for more information.

Or you need to pair the bluetooth devices of the host computer and the NXT brick. There are several programs to do this, one is "kbluetoothd".

AUTHOR

Janosch Graef

SEE ALSO

libanxt(3), **nxt_pilot**(1), **nxt_error**(1), **nxt_udev**(8), **nxt_info**(1)

NAME

nxt_send − Sends a message to a LEGO mindstorms NXT brick

SYNOPSIS

nxt_send [*options* **]** message

DESCRIPTION

Sends *message* to a LEGO mindstorms NXT brick.

AVAILABILITY

Linux

OPTIONS

-c Clear mailbox after receiving from it

-m mailbox

Selects the used mailbox.
mailbox is a number between 1 and 10.
The default is 1.

-n nxtname

Use the NXT with name *nxtname* . The default is the first found brick.

Additionally, the bluetooth adress of the NXT brick can be used as *nxtname* . You can get the bluetooth adress and name of your USB connected NXT brick by using *nxt_info(1)*

EXIT STATUS

The following exit values shall be returned:

0 Successful completion.

>0 A error occured. If the error is caused by a problem of the NXT brick itself, a matching errorstring to the exit value can be displayed with the *nxt_error(1)* command.

EXAMPLES

```
nxt_send -n 01:23:45:67:89:ab "test 1 2 3"
```

Connect to the NXT brick with bluetooth adress "01:23:45:67:89:ab" via bluetooth and send test 1 2 3 to mailbox 1 of the NXT brick.

CAVEATS

You can not get automatically access to the NXT brick.

Either you need access rights to the NXT usb device. Use root rights or see *nxt_udev(8)* for more information.

Or you need to pair the bluetooth devices of the host computer and the NXT brick. There are several programs to do this, one is "kbluetoothd".

AUTHOR

Janosch Graef

SEE ALSO

libanxt(3), **nxt_recv**(1), **nxt_error**(1), **nxt_udev**(8), **nxt_info**(1)

NAME

nxt_sensor − Get a value from a analog LEGO mindstorms NXT sensor

SYNOPSIS

nxt_sensor [*options* **]**

DESCRIPTION

Get a value from a analog LEGO mindstorms NXT sensor.
Analog sensors are the Lego mindstorms NXT light sensor, the Lego mindstorms NXT touch sensor and the Lego mindstorms NXT sound sensor.
Some third party sensors (like the HiTechnic NXT Gyro Sensor) are also analog sensors which can be used with the *nxt_sensor (1)* program. Please note, that most third party sensors are digital sensors and require either the use of *nxt_sensorus (1)* or special written functions similar to the nxt_us_* *libanxt(3)* functions.

The use of the shell command *nxt_sensor (1)* itself is not very usefull over bluetooth, cause the initialisation of the bluetooth connection needs about 5 seconds in average.
See the *nxt_server (1)* program and its examples for a better shell script solution.

AVAILABILITY

Linux

OPTIONS

-m mode

The sensor *mode* setting is responsible for the interpretation of the output. Valid sensors modes are:

raw
boolean
transition_count
period_count
percent
celsius
fahrenheit
angle_step

The default *mode* is "raw".

-n nxtname

Use the NXT with name *nxtname* . The default is the first found brick.

Additionally, the bluetooth address of the NXT brick can be used as *nxtname* . You can get the bluetooth address and name of your USB connected NXT brick by using *nxt_info(1)*

-s sensor

The *sensor* values are: 1, 2, 3, 4. If the -s option is missing, but the -t option is used, the value is depends on the type/sensorportnumber combinations, defined by Lego itself.

Type	Sensorport
reflection	3
light*	3
sound*	2
switch	1
custom	1
lowspeed*	1

If neither the sensorport nor the sensortype is given, the default sensorport is 1.

-t type The sensor type defines, how to interpret the values from a sensor.
Valid sensor types are:

none
switch
temperature
reflection
angle
light_active
light_inactive
sound_db
sound_dba
custom
lowspeed
lowspeed_9v
The default *type* is "none".

-r Reset sensor after reading

-v Verbose mode: print value and unit.

EXIT STATUS

The following exit values shall be returned:

0 Successful completion.

>0 A error occured. If the error is caused by a problem of the NXT brick itself, a matching errorstring to the exit value can be displayed with the *nxt_error(1)* command.

EXAMPLES

nxt_sensor -n 01:23:45:67:89:ab

Connect to the NXT brick with bluetooth address "01:23:45:67:89:ab" via bluetooth and get the value of the sensor at port 1 in raw mode.

CAVEATS

You can not get automatically access to the NXT brick.

Either you need access rights to the NXT usb device. Use root rights or see *nxt_udev(8)* for more information.

Or you need to pair the bluetooth devices of the host computer and the NXT brick. There are several programs to do this, one is "kbluetoothd".

AUTHOR

Janosch Graef

SEE ALSO

libanxt(3), **nxt_sensor_us**(1), **nxt_server**(1), **nxt_error**(1), **nxt_udev**(8), **nxt_info**(1)

NAME

nxt_sensorus − Get a value from the ultrasonic LEGO mindstorms NXT sensor

SYNOPSIS

nxt_sensorus [*options* **]**

DESCRIPTION

Get a value from the ultrasonic digital LEGO mindstorms NXT sensor.
Most third party sensors (e.g. from HiTech or Mindsensor) are compatible to the ultrasonic sensor. So the *nxt_sensorus (1)* program can be used for them too. Some of the third party sensors can be used with the *nxt_sensorus (1)* but do not deliver their full accuracy. Some of the third party sensors are not compatible to LEGOs ultrasonic sensor and there require special written functions similar to the nxt_us_* *libanxt(3)* functions.

The use of the shell command *nxt_sensorus (1)* itself is not very usefull over bluetooth, cause the initialisation of the bluetooth connection needs about 5 seconds in average.
See the *nxt_server (1)* program and its examples for a better shell script solution.

AVAILABILITY

Linux

OPTIONS

-n nxtname

Use the NXT with name *nxtname* . The default is the first found brick.

Additionally, the bluetooth address of the NXT brick can be used as *nxtname* . You can get the bluetooth address and name of your USB connected NXT brick by using *nxt_info(1)*

-s sensor

The *sensor* values are: 1, 2, 3, 4.
The default is the sensorport 4, the usual port (defined by Lego itself) for the ultrasonic sensor.

-r Reset sensor after reading

-v Verbose mode: print value and unit.

EXIT STATUS

The following exit values shall be returned:

0 Successful completion.

>0 A error occured. If the error is caused by a problem of the NXT brick itself, a matching errorstring to the exit value can be displayed with the *nxt_error(1)* command.

EXAMPLES

nxt_sensorus -n 01:23:45:67:89:ab

Connect to the NXT brick with bluetooth address "01:23:45:67:89:ab" via bluetooth and get the value of the ultrasonic sensor at port 4.

CAVEATS

You can not get automatically access to the NXT brick.

Either you need access rights to the NXT usb device. Use root rights or see *nxt_udev(8)* for more information.

Or you need to pair the bluetooth devices of the host computer and the NXT brick. There are several programs to do this, one is "kbluetoothd".

AUTHOR

Janosch Graef

SEE ALSO

libanxt(3), **nxt_sensor**(1), **nxt_server**(1), **nxt_error**(1), **nxt_udev**(8), **nxt_info**(1)

NAME

nxt_server − Start server program to steer the NXT brick

SYNOPSIS

nxt_server [*options* **] [** *key* **]**

DESCRIPTION

The *nxt_server* program is similar to a "open" command for the LEGO mindstorms NXT brick.
Especially when communicating with a NXT brick over bluetooth, multiple shell commands are very slow. Each shell command needs to initialise a bluetooth connection, which take 5 seconds in average. A complete scan for available bluetooth devices take additional 10 seconds.
The *nxt_server* program solves this problem, by initalising the connection to the NXT brick once and then wait for commands via a named pipe.

The *nxt_server* program, makes it possible to program the NXT brick with any programming language, that support open, read, write and flush operations. The name of the named pipe for commands is generated as */tmp/nxt_server_in_$KEY* where *$KEY* is replaced by the *key* commandline argument of the *nxt_server* command (if present), or generated from the process id of the *nxt_server* command (which can be investigated with the *ps(1)* command). The output of the commands is written to the */tmp/nxt_server_out_$KEY* named pipe and the error output is written to the */tmp/nxt_server_err_$KEY* named pipe.

The name of the commands are identical to the names of the aNXT tools, without the "nxt_" prefix. Before a command is written into the input pipe, arguments and options can be used.

A argument is identical to the commandline argument of the matching aNXT tool, but with a preceding "1" or "2" character (followed by a space) for the first or the second commandline argument. No command supports more than two arguments.

Sending data from/to the standard output is not possible. Instead files has to be used. The equivalents of nxt_motor_playback(1) and nxt_motor_record(1) use a "1 filename" command to specify files.

A option is identical to the option of the matching aNXT tool, but without the leading "-" character. Options with values are composed by the option-character, a space and the value.

Sending/receiving data over named pipes require a file open of the named pipe by both partners at the same time.

Sending is blocked, till the other partner opened the pipe and receive data.

Receiving is blocked, till the other partner opened the pipe and send data.

If one partner close the pipe, the communication is lost.

So sucessfull communication needs to know the sequence of open, read and write operations. You can see the sequence for the *nxt_server* operations, when you use the verbose option.

nxt_server has two such sequences (depending on the commandline options): One sequence is made for usual programming languages and one for tools like shell scripts, which need to open/close the named pipe on every command.

The sequence for usage with a usual programming languages is:

Start nxt_server

Open command pipe for output
Open error pipe for input

```
Open result pipe for input

Write 1. option/argument to the command pipe
Write 2. option/argument to the command pipe
Write 3. option/argument to the command pipe
 ...
Write command to the command pipe.

Read error message (maybe a single carriage return) from the error pipe
Read 1. data line from the result pipe
Read 2. data line from the result pipe
 ...
Read single carriage return from the result pipe

Write 1. option/argument to the command pipe
Write 2. option/argument to the command pipe
Write 3. option/argument to the command pipe
 ...
Write command to the command pipe

Read error message from the error pipe
Read 1. data line from the result pipe
Read 2. data line from the result pipe
 ...
Read single carriage return from the result pipe.

etc.
 ...

write the string "exit" to the command pipe to stop nxt_server

Close command pipe.
Close error pipe.
Close result pipe.
```

The sequence for usage with a shell script is:

```
Start nxt_server -o

echo 1. option/argument > command pipe
echo 2. option/argument >> command pipe
echo 3. option/argument >> command pipe
 ...
echo command >> command pipe

cat << error pipe
cat << result pipe

echo 1. option/argument >> command pipe
echo 2. option/argument >> command pipe
echo 3. option/argument >> command pipe
 ...
echo command >> command pipe
```

```
cat << error pipe
cat << result pipe

etc.

echo exit >> command pipe
```

The *nxt_server* program should support almost all aNXT tools, but not administrative tools (tools documented in the section 8 of the manpages) and only commands that require the communication with a NXT brick (e.g. not file converters, not tools for errormessages).

AVAILABILITY

Linux

OPTIONS

-n nxtname

Use the NXT with name *nxtname* . The default is the first found brick.

Additionally, the bluetooth address of the NXT brick can be used as *nxtname* . You can get the bluetooth address and name of your USB connected NXT brick by using *nxt_info(1)*

EXIT STATUS

The following exit values shall be returned:

0 Successful completion.

>0 A error occured. If the error is caused by a problem of the NXT brick itself, a matching errorstring to the exit value can be displayed with the *nxt_error(1)* command.

EXAMPLES

```
nxt_server -n 01:23:45:67:89:ab -o something
```

Connect to the NXT brick with bluetooth address "01:23:45:67:89:ab" via bluetooth and open the connection to the NXT brick for further shell script commands based on key "something" like

```
KEY=something
IN=/tmp/nxt_server_in_$KEY
OUT=/tmp/nxt_server_out_$KEY
ERR=/tmp/nxt_server_err_$KEY

echo "s 1" > $IN
echo "t sound_db" >> $IN
echo "sensor" >> $IN
cat < $ERR
cat < $OUT

echo "1 test.rxe" >> $IN
echo run >> $IN
cat < $ERR
cat < $OUT

echo "exit" >> $IN
```

These commands are similar to

```
nxt_sensor -t -s 1
nxt_run test.rxe
```

The same for a usual programming language:

nxt_server -n 01:23:45:67:89:ab something

Connect to the NXT brick with bluetooth address "01:23:45:67:89:ab" via bluetooth and open the connection to the NXT brick for further commands based on key "something" with a Ada program like

```
with Ada.Text_IO; use Ada.Text_IO;
with Ada.Text_IO.Unbounded_IO; use Ada.Text_IO.Unbounded_IO;
with Ada.Strings; use Ada.Strings;
with Ada.Strings.Unbounded; use Ada.Strings.Unbounded;

procedure Main is
  Key : Unbounded_String;
  Line : Unbounded_String;
  Commandpipe, ErrorPipe, ResultPipe : Unbounded_String;
  Command, Error, Result: File_Type;
begin
  Key := To_Unbounded_String("something");

  CommandPipe := "/tmp/nxt_server_in_" & Key;
  Create(Command, Out_File, To_String(CommandPipe));
  ErrorPipe := "/tmp/nxt_server_err_" & Key;
  Open(Error, In_File, To_String(ErrorPipe));
  ResultPipe := "/tmp/nxt_server_out_" & Key;
  Open(Result, In_File, To_String(ResultPipe));

  Put_Line(Command, "s 1");
  Put_Line(Command, "t sound_db");
  Put_Line(Command, "sensor");
  Flush(Command);

  Get_Line(Error, Line);
  if Length(Line) > 0 then
    Put_Line(Line);
  end if;

  loop
    Line := Get_Line(Result);
    exit when Length(Line) = 0;
    Put_Line(Line);
  end loop;

  Put_Line(Command, "1 test.rxe");
  Put_Line(Command, "run");
  Flush(Command);

  Get_Line(Error, Line);
  if Length(Line) > 0 then
    Put_Line(Line);
  end if;

  loop
    Line := Get_Line(Result);
    exit when Length(Line) = 0;
```

```
      Put_Line(Line);
    end loop;

    Put_Line(Command, "exit");
    Flush(Command);

    Close(Command);
    Close(Error);
    Close(Result);
end;
```

CAVEATS

You can not get automatically access to the NXT brick.

Either you need access rights to the NXT usb device. Use root rights or see *nxt_udev(8)* for more information.

Or you need to pair the bluetooth devices of the host computer and the NXT brick. There are several programs to do this, one is "kbluetoothd".

The communication over named pipes often require flush operations.

AUTHOR

J. "MUFTI" Scheurich (IITS Universitaet Stuttgart)

SEE ALSO

libanxt(3), **nxt_error**(1), **nxt_udev**(8), **nxt_info**(1) **mkfifo**(1) **ps**(1) **nxt_beep**(1), **nxt_delflash**(1), **nxt_download**(1), **nxt_getprog**(1), **nxt_list**(1), **nxt_lsmod**(1), **nxt_motor**(1), **nxt_motor_playback**(1), **nxt_motor_record**(1), **nxt_pollcmd**(1), **nxt_recv**(1), **nxt_remove**(1), **nxt_resetbt**(1), **nxt_run**(1), **nxt_scan**(1), **nxt_screenshot**(1), **nxt_send**(1), **nxt_sensor**(1), **nxt_sensorus**(1), **nxt_setbutton**(1), **nxt_setname**(1), **nxt_stop**(1), **nxt_tacho**(1), **nxt_turnoff**(1), **nxt_up_run**(1), **nxt_upload**(1),

NAME

nxt_setbutton − Remote controll buttons on the LEGO mindstorms NXT brick

SYNOPSIS

nxt_setbutton [*options* **]**

DESCRIPTION

Remote control buttons on the NXT brick via USB or bluetooth.

AVAILABILITY

Linux

OPTIONS

-b button

button Name of button to be sent to the NXT brick. The accepted values are "Enter", "Left", "Right" and "Exit". The default button is "Enter".

-n nxtname

Use the NXT with name *nxtname* . The default is the first found brick.

Additionally, the bluetooth address of the NXT brick can be used as *nxtname* . You can get the bluetooth address and name of your USB connected NXT brick by using *nxt_info(1)*

EXIT STATUS

The following exit values shall be returned:

0 Successful completion.

>0 A error occured. If the error is caused by a problem of the NXT brick itself, a matching errorstring to the exit value can be displayed with the *nxt_error(1)* command.

EXAMPLES

```
nxt_setbutton -n 01:23:45:67:89:ab -b Left
```

Connect to the NXT brick with bluetooth address "01:23:45:67:89:ab" via bluetooth and cause the action, that would occure after pressing the left light gray arrow button on the NXT brick.

CAVEATS

You can not get automatically access to the NXT brick.

Either you need access rights to the NXT usb device. Use root rights or see *nxt_udev(8)* for more information.

Or you need to pair the bluetooth devices of the host computer and the NXT brick. There are several programs to do this, one is "kbluetoothd".

AUTHOR

Janosch Graef

SEE ALSO

libanxt(3), **nxt_pilot**(1), **nxt_error**(1), **nxt_udev**(8), **nxt_info**(1)

NAME

nxt_setname − Play sound on the LEGO mindstorms NXT brick

SYNOPSIS

nxt_setname [*options* **]** *newname*

DESCRIPTION

Change the name of the NXT brick to *newname*

AVAILABILITY

Linux

OPTIONS

-n nxtname

Use the NXT with name *nxtname* . The default is the first found brick.

Additionally, the bluetooth address of the NXT brick can be used as *nxtname* . You can get the bluetooth address and name of your USB connected NXT brick by using *nxt_info(1)*

EXIT STATUS

The following exit values shall be returned:

0 Successful completion.

>0 A error occured. If the error is caused by a problem of the NXT brick itself, a matching errorstring to the exit value can be displayed with the *nxt_error(1)* command.

EXAMPLES

nxt_setname -n 01:23:45:67:89:ab nxtname

Connect to the NXT brick with bluetooth address "01:23:45:67:89:ab" via bluetooth and set its name to *nxtname*

CAVEATS

When involved via bluetooth, the NXT brick has to be switched on and off before the next command via bluetooth can be used. Otherwise, you get a "Could not find NXT" error.

You can not get automatically access to the NXT brick.

Either you need access rights to the NXT usb device. Use root rights or see *nxt_udev(8)* for more information.

Or you need to pair the bluetooth devices of the host computer and the NXT brick. There are several programs to do this, one is "kbluetoothd".

AUTHOR

Janosch Graef

SEE ALSO

libanxt(3), **nxt_pilot**(1), **nxt_error**(1), **nxt_udev**(8), **nxt_info**(1)

NAME

nxt_stop − Stops the currently running program on the NXT brick

SYNOPSIS

nxt_stop [*options* **]**

DESCRIPTION

Stops the currently running program on the LEGO mindstorms NXT brick.

AVAILABILITY

Linux

OPTIONS

-n nxtname

Use the NXT with name *nxtname* . The default is the first found brick.

Additionally, the bluetooth address of the NXT brick can be used as *nxtname* . You can get the bluetooth address and name of your USB connected NXT brick by using *nxt_info(1)*

EXIT STATUS

The following exit values shall be returned:

0 Successful completion.

>0 A error occured. If the error is caused by a problem of the NXT brick itself, a matching errorstring to the exit value can be displayed with the *nxt_error(1)* command.

EXAMPLES

```
nxt_stop -n 01:23:45:67:89:ab
```

Connect to the NXT brick with bluetooth address "01:23:45:67:89:ab" via bluetooth and stops the currently running program.

CAVEATS

You can not get automatically access to the NXT brick.

Either you need access rights to the NXT usb device. Use root rights or see *nxt_udev(8)* for more information.

Or you need to pair the bluetooth devices of the host computer and the NXT brick. There are several programs to do this, one is "kbluetoothd".

AUTHOR

Janosch Graef

SEE ALSO

libanxt(3), **nxt_error**(1), **nxt_udev**(8), **nxt_info**(1)

NAME

nxt_tacho − Get current rotation value of a LEGO mindstorms NXT motor

SYNOPSIS

nxt_tacho [*options* **]**

DESCRIPTION

Get current rotation value ("tachometer count") of a LEGO mindstorms NXT motor.
A LEGO mindstorms NXT motor is also a rotation senor. It can detect, how much rotations/which angle the motor has turned, even when it was switched off.

AVAILABILITY

Linux

OPTIONS

-m motor

Valid *motor* values are: A, B, C for motors at the ports A, B or C. The default value is A.

-n nxtname

Use the NXT with name *nxtname* . The default is the first found brick.

Additionally, the bluetooth address of the NXT brick can be used as *nxtname* . You can get the bluetooth address and name of your USB connected NXT brick by using *nxt_info(1)*

-r Reset tachometer count to 0 after reading.

-v Verbose mode.

EXIT STATUS

The following exit values shall be returned:

0 Successful completion.

>0 A error occured. If the error is caused by a problem of the NXT brick itself, a matching errorstring to the exit value can be displayed with the *nxt_error(1)* command.

EXAMPLES

nxt_tacho -n 01:23:45:67:89:ab

Connect to the NXT brick with bluetooth address "01:23:45:67:89:ab" via bluetooth and print out current tachmeter value of motor A.

CAVEATS

You can not get automatically access to the NXT brick.

Either you need access rights to the NXT usb device. Use root rights or see *nxt_udev(8)* for more information.

Or you need to pair the bluetooth devices of the host computer and the NXT brick. There are several programs to do this, one is "kbluetoothd".

AUTHOR

Janosch Graef

SEE ALSO

libanxt(3), **nxt_error**(1), **nxt_udev**(8), **nxt_info**(1)

NAME

nxt_turnoff − Turn off LEGO mindstorms NXT brick

SYNOPSIS

nxt_turnoff [*options* **]**

DESCRIPTION

Turn off LEGO mindstorms NXT brick.

AVAILABILITY

Linux

OPTIONS

-n nxtname

Use the NXT with name *nxtname* . The default is the first found brick.

Additionally, the bluetooth address of the NXT brick can be used as *nxtname* . You can get the bluetooth address and name of your USB connected NXT brick by using *nxt_info(1)*

EXIT STATUS

The following exit values shall be returned:

0 Successful completion.

>0 A error occured. If the error is caused by a problem of the NXT brick itself, a matching errorstring to the exit value can be displayed with the *nxt_error(1)* command.

EXAMPLES

nxt_turnoff -n 01:23:45:67:89:ab

Connect to the NXT brick with bluetooth address "01:23:45:67:89:ab" via bluetooth and it turn off.

CAVEATS

You can not get automatically access to the NXT brick.

Either you need access rights to the NXT usb device. Use root rights or see *nxt_udev(8)* for more information.

Or you need to pair the bluetooth devices of the host computer and the NXT brick. There are several programs to do this, one is "kbluetoothd".

AUTHOR

Janosch Graef

SEE ALSO

libanxt(3), **nxt_error**(1), **nxt_udev**(8), **nxt_info**(1)

NAME

nxt_upload − upload a file to the LEGO mindstorms NXT brick

SYNOPSIS

nxt_upload [*options* **]** input [output]

DESCRIPTION

Upload a file from the host computer to the NXT brick.
If the name of the outputfile on the NXT brick (*output*) is not given, the name of the inputfile on the host-computer (*input*) is used.

AVAILABILITY

Linux

OPTIONS

-f Force overwriting of already existing files.

-n nxtname

Use the NXT with name *nxtname* . The default is the first found brick.

Additionally, the bluetooth address of the NXT brick can be used as *nxtname* . You can get the bluetooth address and name of your USB connected NXT brick by using *nxt_info(1)*

-o writemode

For *writemode* use one of the following write modes:
fragment
linear
append
It is also possible to use "f" instead of "fragment", "l" instead of "linear" and "a" instead of "append".
Acording to some information in the internet, user application programs, "try me programs", lego sound files and lego icon files have to be linear files. All other files can be fragmented files.
The use of the append flag will append the file data at the end of a already existing file.
The default *writemode* is "linear".

EXIT STATUS

The following exit values shall be returned:

0 Successful completion.

>0 A error occured. If the error is cause by a problem of the NXT brick itself, a matching errorstring to the exit value can be displayed with the *nxt_error(1)* command.

EXAMPLES

nxt_upload -n 01:23:45:67:89:ab test.rxe

Connect to the NXT brick with bluetooth address "01:23:45:67:89:ab" via bluetooth and upload the file test.rxe from host computer as to the NXT brick as a linear file test.rxe.

CAVEATS

You can not get automatically access to the NXT brick.

Either you need access rights to the NXT usb device. Use root rights or see *nxt_udev(8)* for more information.

Or you need to pair the bluetooth devices of the host computer and the NXT brick. There are several programs to do this, one is "kbluetoothd".

AUTHOR

Janosch Graef

SEE ALSO

libanxt(3), **nxt_up_run**(1), **nxt_error**(1), **nxt_udev**(8), **nxt_info**(1)

NAME

nxt_up_run − upload and run a file to the LEGO mindstorms NXT brick

SYNOPSIS

nxt_up_run [*options* **]** input [output]

DESCRIPTION

Upload a file from the host computer to the NXT brick and then start it on the NXT brick.
If the name of the outputfile on the NXT brick (*output*) is not given, the name of the inputfile on the hostcomputer (*input*) is used.

AVAILABILITY

Linux

OPTIONS

-n nxtname

Use the NXT with name *nxtname* . The default is the first found brick.

Additionally, the bluetooth address of the NXT brick can be used as *nxtname* . You can get the bluetooth address and name of your USB connected NXT brick by using *nxt_info(1)*

EXIT STATUS

The following exit values shall be returned:

0 Successful completion.

>0 A error occured. If the error is cause by a problem of the NXT brick itself, a matching errorstring to the exit value can be displayed with the *nxt_error(1)* command.

EXAMPLES

```
nxt_up_run -n 01:23:45:67:89:ab test.rxe
```

Connect to the NXT brick with bluetooth address "01:23:45:67:89:ab" via bluetooth, upload the file test.rxe from host computer to the NXT brick as a linear file test.rxe and start this file on the NXT brick.

CAVEATS

You can not get automatically access to the NXT brick.

Either you need access rights to the NXT usb device. Use root rights or see *nxt_udev(8)* for more information.

Or you need to pair the bluetooth devices of the host computer and the NXT brick. There are several programs to do this, one is "kbluetoothd".

AUTHOR

Janosch Graef

SEE ALSO

libanxt(3), **nxt_upload**(1), **nxt_run**(1), **nxt_error**(1), **nxt_udev**(8), **nxt_info**(1)