NXT control

The NXT object is used to manage the connection between your Android phone and the Lego Mindstorms NXT brick.

Description

Create an NXT object using the **CreateNxt** function of the **app** object:

```
nxt = app.CreateNxt();
```

Methods

Method	Description
Beep(frequency, duration)	Commands the NXT brick to play a tone of a given **frequency** for a given **duration** in milliseconds.
Brake(motors)	Commands the NXT brick to Brake one or more of it's motors. The **motors** parameter can be a single motor, for example just "A" or a combination of motors such as "ABC". Unlike the Stop function, this function actively brakes the motors using power from the battery.
CheckConnection()	Returns true if the NXT brick is currently connected to the phone/tablet but it will show a popup warning message and return false if no brick is currently connected. This function is useful if you want to warn the user that he/she have no connection.
Connect(name)	Connects to the Bluetooth device with given **name**.
Disconnect()	Disconnects the Bluetooth link between your phone and the NXT brick.
Drive(motors, power, rotation)	Commands the NXT brick to drive one or more of it's motors with a given power and number of rotations. The **motors** parameter can be a single motor, for example just "A" or a combination of motors such as "ABC". **Power** is given as a percentage value and negative values make the motor reverse. If the number of **rotations** is set to zero, then the motor will continue until the Stop function is called.
FileFindFirst(p1)	
FileFindNext(p1)	
GetBtAddress(obj)	Returns Bluetooth address of a connected NXT brick or NXT brick address given as optional **obj** parameter.
GetBtName(obj)	Returns Bluetooth name of a connected NXT brick or NXT brick name given as optional **obj** parameter.
GetCurrentProgram()	Gets the name of the NXT-G program currently running on the NXT brick.
GetRotationCount(motor)	

Method	Description
IsBluetoothEnabled()	Checks if Bluetooth is enabled on the phone/tablet. Returns **true** if Bluetooth is enabled and **false** if it is disabled. This is alias for function IsEnabled.
IsConnected()	Returns true if the NXT brick is currently connected to the phone/tablet. Unlike CheckConnection method, it will **not show a popup warning message** and return false if no brick is currently connected.
IsEnabled()	Checks if Bluetooth is enabled on the phone/tablet. Returns **true** if Bluetooth is enabled and **false** if it is disabled. This is the same function as IsBluetoothEnabled.
IsMotorIdle(motor)	Checks if the specified **motor** ("A","B","C") is currently powered. This can be useful in order to check if a previously sent motor command has completed.
IsPaired(name)	Checks if Bluetooth device with **name** is on the paired devices list of our phone/tablet. Returns "true" or "false".
PlaySoundFile(file, repeat)	Commands the NXT brick to play a sound file (.rso) file which is available on the brick. The **file** parameter should be the name of the sound file and the **repeat** parameter should be number of times you want the sound to repeat.
ReadColorSensor(input, mode)	Reads the color currently being 'seen' by the NXT color sensor. (Note: brick firmware 1.28 or greater is required for this function) The **input** parameter should be an NXT input port number between 1 and 4 (which the color sensor is plugged into). The **mode** parameter should be one of the following values:- "ColorDetect", "LightSense", "RedSense", "GreenSense", "BlueSense". If the mode ColorDetect is chosen, then the returned values will be a number between 1 and 6 which represent the following colors: black, blue, green, yellow, red, white. You can use the ToColorName function to convert from these six values to a color name. If any of the other modes are chosen, then the result will be a color intensity value between 0 and 1023.
ReadDistanceSensor(input)	Reads the distance measured by the ultrasonic sensor in centimeters. The **input** parameter should be an NXT input port number between 1 and 4 (which the ultrasonic sensor is plugged into).
ReadMail(mailbox, type, remove)	Reads a message from the NXT brick's mail box. This message can be written using a normal NXT-G program running on the brick. This allows you to read values from NXT-G programs with your phone or tablet. The **mailbox** parameter is the target mailbox number and can be a value between 1 and 10. The **type** parameter should be one of the following values:- "Text", "Number" or "Logic" depending on what type of value you wish to read from the brick's mailbox. The **remove** parameter should be true if you wish to remove the message from the NXT's mailbox after reading it or false if you wish to leave the message in the mailbox.

Method	Description
ReadLightSensor(input, active)	Reads the intensity of the light currently being 'seen' by the NXT light sensor. The **input** parameter should be an NXT input port number between 1 and 4 (which the light sensor is plugged into). The **active** parameter should be set to true if you want the white light to be turned on during sensing and false if the light is not required. The returned value will be a light level value between 0 and 100.
ReadTouchSensor(input)	Reads the state of the NXT touch sensor, which will be true if the switch is currently pushed in and false otherwise. The **input** parameter should be an NXT input port number between 1 and 4 (which the touch sensor is plugged into).
ReadSoundSensor(input,mode)	Reads the sound pressure level (loudness) of the sound currently being 'heard' by the sound sensor (Note: Sound sensors are not included with the standard NXT kit) The **input** parameter should be an NXT input port number between 1 and 4 (which the sound sensor is plugged into). The **mode** parameter should be one of the following values:- "DB" or "DbA" depending if you want the returned value in standard Decibels or A-weighted Decibels.
RequestEnable()	Invokes system dialog box which could enable Bluetooth on tablet/phone when Bluetooth is disabled. If Bluetooth is enabled, this function doesn't invoke any visible dialog box. Be careful when checking Bluetooth state immediately after this function by calling IsEnabled or IsBluetoothEnabled, while RequestEnable doesn't stop program execution, and there is no SetOnRequestEnabled method called after this function invoke. This function is internally invoked by ShowDevices method when Bluetooth is disabled.
SendMail(mailbox, type, message)	Sends a **message** to the NXT brick's mail box. This message can be read using a normal NXT-G program running on the brick. This allows you to communicate with NXT-G programs from your phone or tablet. The **mailbox** parameter is the target mailbox number and can be a value between 1 and 10. The **type** parameter should be one of the following values:- "Text", "Number" or "Logic" depending on what type of value you wish to send to the brick's mailbox.
SetInvert(boolean)	Provides a convenient way to invert the direction of the motor commands.
SetLampColor(input, color)	Commands the NXT brick to set the **color** sensor lamp to one of the following: 'White', 'Red', 'Green', 'Blue', 'Off'. The **input** parameter should be an NXT input port number between 1 and 4 (which the color sensor is plugged into).
SetOnConnect(myfunc)	The **SetOnConnect** function allows you to set the name of a function that you would like to be called when the NXT has been successfully connected via Bluetooth. Callback function returns status of connection (true if connection is successful, and false, if connection failed) and caller nxt object.
SetOnConnected(myfunc)	Allows you to set the name of a function that you would like to be called when the NXT has been successfully connected via Bluetooth.

Method	Description
ShowDevices()	Shows the user a dialog box that contains a list of NXT bricks which are paired with the phone. The user can then select the brick to connect to via Bluetooth.
StartProgram(program)	Starts an NXT-G **program** (.rxe file) on the NXT brick (if the program is available on the brick). Note: This function will also launch .rso sound files.
Stop(motors)	Commands the NXT brick to Stop powering one or more of it's motors and allow them to coast to a halt. The **motors** parameter can be a single motor, for example just "A" or a combination of motors such as "ABC".
StopProgram()	Stops the NXT-G program (.rxe file) which is currently running on the NXT brick.
ToColorName(colorNum)	Function returns string with one of the six colors: black, blue, green, yellow, red, white converted from **colorNum** parameter. You can use the ToColorName function to convert value obtained from color sensor by method ReadColorSensor with mode set to ColorDetect.

From:

https://wiki.droidscript.me.uk/ - **DroidScript wiki**

Permanent link:

https://wiki.droidscript.me.uk/doku.php?id=built_in:nxt

Last update: **2015/04/18 10:02**