

## EZ-Script Functions

### Sleep (milliseconds)

Pauses for specified milliseconds

Example sleeps for 1 second: Sleep(1000)

### SleepRandom (lowMilliSec, highMilliSec)

Pauses for a random millisecond delay between the 2 provided values

Example: SleepRandom(1000, 5000)

### Servo (servoPort, position)

Move servo to the specified position

Servo position is between 0 and 100

Example: Servo(D14, 25)

### PWM (digitalPort, speed)

Set the PWM (Pulse Width Modulation) to the desired duty percentage cycle

This simulates voltage on the specified pin (Between 0 and 5v)

PWM Value is between 0 and 100

Example: PWM(D14, 90)

### GetPWM (digitalPort)

Gets the PWM (Pulse Width Modulation) of specified port

PWM is between 0 and 100

Example: \$x = GetPWM(D14)

### PWMRandom (digitalPort, lowSpeed, highSpeed)

Set the PWM (Pulse Width Modulation) to a random percentage duty cycle

This simulates voltage on the specified pin (Between low and high percentage value, scaled between 0 and 5 volts)

The value is between 0 and 100

Example: PWMRandom(D14, 10, 90)

### ServoSpeed (servoPort, speed)

Set the speed of servo. This is the speed of which the servo will move between positions.

The servo speed is a number between 0 (fastest) and 10 (slowest)

\*Note: If the ServoSpeed command is being used in an initialization script, ensure the servo positions are set first. This is because with Servo

Example: ServoSpeed(D14, 25)

### ServoSpeedRandom (servoPort, lowSpeed, highSpeed)

Set the servo speed to a random value

The servo speed is a number between 0 (fastest) and 10 (slowest)

Example: ServoSpeedRandom(D14, 10, 20)

### ServoUp (servoPort, count)

Increment the servo position value by specified count

Servo position is between 0 and 100

Example: ServoUp(D14, 1)

### ServoDown (servoPort, count)

Decrement the servo position value by specified count

Servo position is between 0 and 100

Example: ServoDown(D14, 1)

### ServoRandom (servoPort, lowPosition, highPosition)

Move the servo to a random position between low and high

Servo position is between 0 and 100

Example: ServoRandom(D14, 10, 20)

### Release (servoPort)

Release a servo from holding its position

Example: Release(D14)

### ReleaseAll ( [boardIndex] )

Release all servos from holding their position

BoardIndex is optional, and specified the EZ-B board to use

Example: ReleaseAll()

Move (servoPort, forward/stop/reverse)

Set a modified servo to move

Example: Move(D14, "forward")

Set (digitalPort, on/off/true/false)

Set a digital port state to either on or off

Example: Set(D2, OFF)

SetRandom (digitalPort)

Set a digital port to a random state of either on or off

Example: SetRandom(D2)

ToggleDigital (digitalPort )

Toggle the digital port

Example: ToggleDigital(D2)

Digital\_Wait (digitalPort, on/off/true/false, [delay ms])

Wait until the digital port status has changed

The optional parameter Delay MS is the millisecond delay for checking. This value determines the delay between checks.

Example: Digital\_Wait(D12, ON)

Example: Digital\_Wait(D12, ON, 50)

ADC\_Wait (adcPort, higher/lower/equals, value, [delay ms])

Wait until ADC port is higher or lower than specified value

The optional parameter Delay MS is the millisecond delay for checking. This value determines the delay between checks.

Example: ADC\_Wait(ADC0, HIGHER, 50)

Example: ADC\_Wait(ADC0, HIGHER, 50, 50)

ADC\_Wait\_Between (adcPort, low, high, [delay ms])

Wait (pauses script) until ADC port is between the specified values. Soon as the ADC port is between the low and high values, it will stop wait

The optional parameter Delay MS is the millisecond delay for checking. This value determines the delay between checks.

Example: ADC\_Wait\_Between(ADC0, 20, 50)

Example: ADC\_Wait\_Between(ADC0, 20, 50, 50)

Movement\_Wait ( forward/reverse/stop/left/right )

Wait until a movement from the movement panel is specified.

Even if another script executes a movement, this will execute

Example: Movement\_Wait("FORWARD")

Servo\_Wait ( digitalPort, higher/lower/equals, value )

Wait until the Servo Port is higher or lower than specified value.

Zero can be specified as a value for a stopped servo.

Example: Servo\_Wait(D5, HIGHER, 20)

WaitForServoMove (servoPort)

Waits for the specified servo to move.

Unlike Servo\_Wait, this function does not wait for a specific value. It simply returns once the servo has moved to a new position.

Example: WaitForServoMove(d0)

Ping\_Wait (triggerPort, echoPort, higher/lower/equals, distance)

Wait until the Ping Sensor distance is higher or lower than specified distance value.

Trigger and Echo are Digital Ports

Example: Ping\_Wait(D3, D4, HIGHER, 50)

Forward( [speed], [milliSeconds] )

Using a Movement Panel Control, this will start your robot in the Forward direction.

Optionally, you can provide the speed and/or number of milliseconds to move.

You will require at least one movement panel to be configured within the project. This function will control that movement panel.

Speed is a number between 0 and 255

Example: Forward()

Example: Forward(200)

Example: Forward(255, 5000)

Reverse( [speed], [milliSeconds] )

Using a Movement Panel Control, this will start your robot in the Reverse direction.

Optionally, you can provide the speed and/or number of milliseconds to move.

You will require at least one movement panel to be configured within the project. This function will control that movement panel.

Speed is a number between 0 and 255

Example: Reverse()

Example: Reverse(200)

Example: Reverse(255, 5000)

Stop()

Using Movement Panel Control, this will stop your robot.

You will require at least one movement panel to be configured within the project. This function will control a movement panel.

Example: Stop()

Left( [speed], [milliSeconds] )

Using a Movement Panel Control, this will turn your robot left.

You will require at least one movement panel to be configured within the project. This function will control that movement panel.

Optionally, you can specify the speed and/or number of milliseconds to turn.

Speed is a number between 0 and 255

Example #1: Left()

Example #2: Left(200)

Example #2: Left(200, 5000)

Right( [speed], [milliSeconds] )

Using a Movement Panel Control, this will turn your robot right.

You will require at least one movement panel to be configured within the project. This function will control that movement panel.

Optionally, you can specify the speed and/or number of milliseconds to turn.

Speed is a number between 0 and 255

Example #1: Right()

Example #2: Right(200)

Example #2: Right(200, 5000)

Up( [milliSeconds] )

Using the servo port settings from a Movement Panel Control, this will raise your drone robot up

Optionally, you can specify the number of milliseconds to turn.

Example #1: Up()

Example #2: Up(1000)

Down( [milliSeconds] )

Using the servo port settings from a Movement Panel Control, this will lower your drone robot up

Optionally, you can specify the number of milliseconds to turn.

Example #1: Down()

Example #2: Down(1000)

RollRight( [milliSeconds] )

Using the servo port settings from a Movement Panel Control, this will roll your drone robot right

Optionally, you can specify the number of milliseconds to turn.

Example #1: RollRight()

Example #2: RollRight(1000)

RollLeft( [milliSeconds] )

Using the servo port settings from a Movement Panel Control, this will roll your drone robot left

Optionally, you can specify the number of milliseconds to turn.

Example #1: RollLeft()

Example #2: RollLeft(1000)

Land()

Tell your flying drone to land

Example: Land()

TakeOff()

Tell your flying drone to take off

Example: TakeOff()

DroneEmergency()

Tell your flying drone to reset from emergency or power down when flying.  
This command should be added a button on the joystick so you may stop the drone so it does not get away or in danger.

Say( text to speech )

Speaks the text that is specified within the brackets out of the PC Sound Card in the background. This command does not block, the script will continue to run.  
Example: Say("Hello, I am a robot")

SayWait( text to speech )

Speaks the text that is specified within the brackets out of the PC Sound Card and blocks until done speaking.  
Example: SayWait("Hello, I am a robot")

SpeakStop( )

Stops speaking the current specified phrases.  
Example: SpeakStop()

SpeakRSS( url, [story index] )

Speaks the title and text of the rss url  
Example #1: SpeakRSS("http://rss.cbc.ca/lineup/world.xml")  
Example #2: SpeakRSS("http://rss.cbc.ca/lineup/world.xml", 3)

SpeakRSSDescription( url, [story index] )

Speaks only the text of the rss url  
Example #1: SpeakRSSDescription("http://rss.cbc.ca/lineup/world.xml")  
Example #2: SpeakRSSDescription("http://rss.cbc.ca/lineup/world.xml", 3)

SpeakTwitter( twitterUserName, [story index] )

Speaks the twitter feed for the specific username  
Example #1: SpeakTwitter("EZ\_Robot")  
Example #2: SpeakTwitter("EZ\_Robot", 3)

SpeakVolume( value )

The volume of the speech synthesizer.  
The value is between 0 and 100  
Example: SpeakVolume(30)

I2CWrite( boardIndex, deviceAddress, data, .... )

Send a series of data to the specified device hex address over the i2c interface.  
This command will Start i2c, Write Data, and Stop i2c.  
boardIndex is the EZ-B you wish to use (0 is first EZ-B)  
Device Hex Address of i2c device must be in 0x00 format.  
Data can be Hex (0x09), string ("string"), or decimal (188)  
Example: I2CWrite(0, 0x09, 0x02, 0x05, 0x06)  
Example: I2CWrite(0, 0x09, 244)  
Example: I2CWrite(0, 0x09, "This is text " + \$variable)

I2CRead( boardIndex, ack, deviceAddress, bytes to expect )

Returns a series of data from the specified address over the i2c interface  
boardIndex is the EZ-B you wish to use (0 is first EZ-B)  
Device Hex Address must be hexadecimal (0x5e)  
Ack is either TRUE, FALSE or AUTO  
AUTO ACK will acknowledge every byte read except the last byte (recommended)  
TRUE ACK will acknowledge every byte read  
FALSE ACK will not acknowledge any bytes read  
You must specify the number of Bytes To Expect  
Example: \$Val = I2CRead(0, Auto, 0x5e, 2)

SendSerial( digitalPort, baudRate, data, ... )

Send a series of data over the specified port and baud rate  
Data can be Hex (0x09), string ("string"), or decimal (188)  
Example: SendSerial(d0, 9600, 0x00, 0x04, 0x05)  
Example: SendSerial(d0, 9600, 244, 200, "a")  
Example: SendSerial(d0, 9600, "This is text")  
Example: SendSerial(d0, 9600, "Hello " + \$name)

WaitUntilTime( hour, minute )

Waits until the specified time. The script will stop at this command and not continue until the specified time. The time is declared in 24 hour format.  
Example: WaitUntilTime(17, 30)

MP3TriggerPlayTrack( digitalPort, baud, trackNumber, [pause time] )

Plays the specified MP3 track from the MP3 Trigger Shield

Optionally, the Pause Time value can be used which disables the Speech Recognition control for the specified number of milliseconds.

Example: MP3TriggerPlayTrack( d0, 38400, 1 )

Example: MP3TriggerPlayTrack( d0, 38400, 1, 3000 )

MP3TriggerVolume( digitalPort, baud, volume )

Set the volume of the mp3 trigger between 0 and 255. 0 is louded, 255 is quiet.

Example: MP3TriggerVolume( d0, 38400, 20 )

MP3TriggerPlayRandomTrack( digitalPort, baud, lowestTrackNum, highestTrackNum )

Plays a random MP3 track from the MP3 Trigger Shield between the supplied track numbers

Example: MP3TriggerPlayRandomTrack( d0, 38400, 1, 10 )

MP3TriggerNext( digitalPort, baud )

Plays the next MP3 track from the MP3 Trigger Shield

Example: MP3TriggerPlayNext( d0, 38400 )

MP3TriggerPrev( digitalPort, baud )

Plays the previous MP3 track from the MP3 Trigger Shield

Example: MP3TriggerPrevious( d0, 38400 )

MP3TriggerStop( digitalPort, baud )

Stops the current MP3 track from the MP3 Trigger Shield

Starts the mp3 file if not playing

Example: MP3TriggerStop( d0, 38400 )

ControlCommand( windowName, ControlCommandParameter, [values])

Sends a command to the window by its name. Look further down in this document for available control command parameters under the Control Command section.

Some commands require an additional parameter. When editing EZ-Script, check the Control Details tab to view all available commands for the window.

This command has a shorthand alias which is "CC" (See examples below).

Example: ControlCommand( "ADC Graph", pauseOn )

Example: ControlCommand( "SoundBoard", Track\_3 )

Example: ControlCommand( "Camera", CameraTweet, "This is an Image Description" )

Example: ControlCommand( "Script Manager", ScriptStart, "MyScript" )

Example: ControlCommand( "Speech Recognition", PauseMS, 3000 )

Example: cc("Auto Position", AutoPositionAction, "Action Name")

Example: cc("Auto Position", AutoPositionFrame, "Frame Name")

Example: cc("Auto Position", AutoPositionFrame, "Frame Name", 50, 3)

Example: cc("Speech Recognition", PauseMS, 2000)

# Commented Text

Comment a line of code

Example: # This is a comment. This code will not run

:Label

Defines a label for a GOTO() command

Example: :My\_Label

Goto( label )

Goto a specific :Label location

Example: Goto(My\_Label)

Return()

Return from a Goto()

If you jump to a position of code with a Goto(), the Return statement will allow you to return back to that piece of code following the last Goto()

If you attempt to Return() with an empty stack, nothing will happen. The script will ignore the Return() statement.

Example: Return()

If (Value Condition Value )

IF condition can support multiple comparisons and functions.

Condition tree must be closed with an ENDIF  
See the Functions section of this document.  
Condition can be =, <, >, <=, >=, !=, AND, OR

Example:

```
If (GetDigital(D0) = 1)
  Print("One")
EndIf
```

Example:

```
If ($Day = 2 AND $Hour = 3)
  Print("Hello")
EndIf
```

Example:

```
If (GetServo(D5) >20 OR ($x >= 3 and $y < 2))
  Print("Yup!")
EndIf
```

Elseif (Value Condition Value )

Elseif condition can support multiple comparisons and functions.

Condition tree must be closed with an ENDIF

See the Functions section of this document.

Condition can be =, <, >, <=, >=, !=, AND, OR

Example:

```
If (GetServo(D0) = 1)
  Print("One")
Elseif (GetServo(D0) = 2)
  Print("Two")
Else
  Print("Something Else")
EndIf
```

Else

Condition tree must be closed with an ENDIF

Example:

```
If (GetDigital(D0) = 1)
  Print("Yes")
Else
```

Else

```
  Print("No")
```

```
EndIf
```

GetServo( Port )

Returns the Servo Position value of the specified port

Servo position is between 0 and 100

Example: GetServo(d0)

GetServoSpeed( Port )

Returns the Servo Speed value of the specified port

Speed is between 0 and 10

Example: GetServoSpeed(d0)

SetSpeed( speed, [speedRight] )

Sets the global Movement Speed value

If you specify only one parameter, the speed of both the left and right wheel will be modified

If you specify two parameters, the first parameter is the speed of the Left wheel and second parameter is the speed of the Right wheel.

Speed is between 0 (slow) and 255 (fast)

Example: SetSpeed(50)

Example: SetSpeed(50, 100)

SetSpeedLeft( speed )

Sets the global Movement Speed value of the Left wheel

Speed is between 0 (slow) and 255 (fast)

Example: SetSpeedLeft(50)

SetSpeedRight( speed )

Sets the global Movement Speed value of the Right wheel

Speed is between 0 (slow) and 255 (fast)

Example: SetSpeedRight(50)

GetSpeed( )

Returns the global Movement Speed value

Speed is between 0 and 255

Example: GetSpeed()

GetSpeedLeft( )

Returns the global Movement Speed value of Left wheel

Speed is between 0 and 255

Example: GetSpeedLeft()

GetSpeedRight( )

Returns the global Movement Speed value of Right wheel

Speed is between 0 and 255

Example: GetSpeedRight()

GetADC( Port )

Returns the ADC value of the specified port

Example: GetADC(adc0)

GetRandom( Min, Max )

Returns a random number between Min and Max

Example: GetRandom(10, 50)

GetRandomUnique( Min, Max )

Returns a random number between Min and Max

This function attempts to make the number unique from the last time it was called

Example: GetRandomUnique(10, 50)

GetDigital( Port )

Returns the Digital value of the specified port as a 0 or 1

Example: \$x = GetDigital(d0)

ASin( value )

Returns the math ASin() function (also called ArcSin)

Example: \$x = ASin(27)

ACos( value )

Returns the math ACos() function (also called ArcCos)

Example: \$x = Acos(27)

Sqrt( value )

Returns the math Square Root function

Example: \$x = Sqrt(9)

Power( value, power )

Returns the math Power() function

First parameter is the input value

The second parameter is the power

Example: \$x = Power(2, 4)

Sin( value )

Returns the math SIN() function

Example: \$x = Sin(27)

Cos( value )

Returns the math COS() function

Example: \$x = Cos(27)

Abs( value )

Returns the absolute value of a number

Converts a negative into a positive number

Example: \$x = Abs(-22)

Round( value, decimals )

Returns the math Round() of a number

Returns the number rounded to the specified decimal places

Example: \$x = Round(\$pi, 2)

Example: \$x = Round(9.3848291, 1)

GetPing( trigger port, echo port )

Return the Ping HC-SR04 value of the specified port

Example: GetPing(d0, d1)

CompassTurn( degrees, resolution, [time to turn], [time to stop] )

Turns your robot towards the specified compass degrees. A movement panel and compass control needs to be included with your project. The

Compass must be connected to EZ-B Board #0

Example: CompassTurn( 60, 10)

Example: CompassTurn( 60, 10, 1000, 2000)

CompassForward( degrees, resolution, [time to turn], [time to stop] )

Drives your robot forward and maintains the specified compass degree heading

Compass must be connected to EZ-B Board #0

Example: CompassForward( 60, 10)

Example: CompassForward( 60, 10, 1000, 2000)

GPSStop( latitude, longitude, resolution )

Uses the attached GPS control and stops the movement panel when the coordinates are within the specified resolution for the latitude and lon

Example: GPSStop( 54.01438, -110.4931, 0.0005)

RoboSapien( RoboSapienCommand )

Send a command to a RoboSapien connected on port D1 on EZ-B 0. Find the available RoboSapien commands further down in this document

Example: RoboSapien( WalkForward )

Example: RoboSapien( LeftArmUp )

RoboQuad( RoboQuadCommand )

Send a command to a RoboQuad connected on port D1 on EZ-B 0. Find the available RoboQuad commands further down in this document.

Example: RoboQuad( Walk\_Forward )

Example: RoboQuad( Left\_Turn\_Roll )

Tweet( message )

Send a Twitter message using the configured Twitter account. Configure your Twitter account under File->Twitter Settings.

Example: Tweet("I Love EZ-Robot!")

You may also use the ControlCommand to Tweet images with text from a Camera Control.

Example: ControlCommand( "Camera", CameraTweet, "Our New Image" )

HTTPGet( url )

Send an HTTP Get command to the provided address and return the contents

Example: HTTPGet("http://192.168.0.10/decoder\_control.cgi?command=35&onestep=5&user=admin&pwd=admin")

Example: \$temp = HTTPGet("http://192.168.0.15/GetTemperature.cgi ")

Roomba( cmd )

Execute the specified command on a connected Roomba Vacuum on Port D0 and EZ-B 0. Look for the available Roomba commands further

You may also add the Roomba Movement Panel for graphical controls.

Example: Roomba(init)

Example: Roomba(SideBrushOn)

Halt()

Exit the current running script.

Example: Halt()

Print( txt )

Outputs the specified text to the debug console

Example: Print("This is some text")

Example: Print("Today is \$Day")

Example: Print("\$pi rounded is Round(\$pi, 2)")

PrintHex( txt )



Outputs the hex values of the specified variable to the debug console

Example: PrintHex(\$myVariable)

Exec( EXE/Bat File, [parameters] )

Executes a windows application or batch file. The second parameter is a list of optional parameters

Example: Exec("C:\Windows\notepad.exe")

Example: Exec("C:\Windows\notepad.exe", "C:\MyFile.txt")

Browser( url )

Launches the default web browser with the specified URL.

Example: Browser("http://www.google.com")

FileDelete( filename )

Deletes a file on your computer

Example: FileDelete("c:\temp\mylog.txt")

FileWrite( filename, contents )

Appends text to the specified file. This does not append a new line.

Example: FileWrite("c:\temp\mylog.txt", "My Variable: " + \$x)

Example: FileWrite("c:\temp\mylog.txt", "Servo Position: GetServo(d2)")

FileWriteLine( filename, contents )

Appends text as a new line to the specified file.

Example: FileWriteLine("c:\temp\mylog.txt", "My Variable: " + \$x)

Example: FileWriteLine("c:\temp\mylog.txt", "Servo Position: GetServo(d2)")

FileReadClose( filename )

Closes the file from reading.

This must call must be performed before writing to the file. Once you begin reading from the file, the file is OPEN. Closing the file will reset to

Example: FileReadClose("c:\temp\mylog.txt")

FileReadReset( filename )

Resets the file to the beginning.

If you read to the end of a file, this function must be called to reset reading from the beginning of the file.

Example: FileReadReset("c:\temp\mylog.txt")

FileExists( filename )

Returns a 1 or 0 if the specified file exists.

Example: \$fileExists = FileExists("c:\temp\mylog.txt")

FileReadEnd( filename )

Returns a 1 or 0 if the file has reached the end.

Example: \$fileEnd = FileReadEnd("c:\temp\mylog.txt")

FileReadChar( filename )

Returns the next character in the specified file

Example: \$char = FileReadChar("c:\temp\mylog.txt")

FileReadLine( filename )

Returns the next line of the specified filename.

Example: \$line = FileReadLine("c:\temp\mylog.txt")

FileReadAll( filename )

Returns the entire contents of the specified file.

Example: \$contents = FileReadAll("c:\temp\mylog.txt")

FileReadLineRandom( filename )

Returns a random line within the specified file

Example: \$randomLine = FileReadLineRandom("c:\temp\mylog.txt")

Split( text, splitChar, index )

Splits a line of text by the specified SplitChar into an array and returns the Index.

The Index is zero based, which means 0 (zero) is the first item within the array.

Example: \$contents = Split("One, Two, Three", ",", 1)

Example: \$contents = Split("One - Two - Three", "-", 2)

WaitForChange( value )

Pauses the execution of a script until the specified value has changed.

The value can be either a variable or Get function.

Example: WaitForChange(\$x)

Example: WaitForChange(GetServo(d0))

Example: WaitForChange(GetDigital(d0))

Length( value )

Returns the length of the specified variable or string in characters/bytes.

Example: \$len = Length("This string is 33 characters long")

GetCharAt( value, index )

There are two methods to obtain a character within a string. This method, which is GetCharAt(), or using the [] square brackets. See the examples.

Returns the character at the specified index.

If the character at the specified position is outside of readable ASCII, you will want to use GetByteAt() or GetByte() functions instead.

The Index is zero based, which means 0 (zero) is the first character.

Example: \$char = GetCharAt("Hello World", 2)

Example: \$char = GetCharAt(\$x, 2)

Example: \$byte = \$x[3]

Example: \$byte = \$x[\$y]

GetByteAt( value, index )

Returns the ASCII Ordinal value of the byte at the specified location within the array.

If the byte is 0x05, this function will return an integer value of 53. Use this function to convert data read by i2c into ordinal values.

The Index is zero based, which means 0 (zero) is the first character.

Example: \$value = GetByteAt("Hello World", 2)

Example: \$value = GetByteAt(\$x, 2)

GetByte( value )

Returns the ASCII Ordinal value of a byte or byte array.

Technically, this function returns a number and not specifically a byte.

The number of bytes in the variable will determine the size of the integer returned. If one byte is passed, an 8 bit number is returned. If two bytes are passed, a 16 bit number is returned.

If the variable contains 0x37, this function will return an integer value of 53. Use this function to convert data read by i2c into ordinal values.

Example: \$value = GetByte("H")

Example: \$value = GetByte(\$x)

GetAsByte( value )

Returns the byte of the integer or first character of string.

If you pass 75 as the value, you will get the ASCII value of the letter K

Example: \$value = GetAsByte("H")

Example: \$value = GetAsByte(\$x)

DefineArray( variable, size, [default byte value] )

Creates a byte array of the variable to the specified size.

Optionally, you may also pass a byte that will be used for the default values. If it is not set, a default of \0 (null) will be set for each byte.

Example: DefineArray(\$myArray, 10)

Example: DefineArray(\$myArray, 10, 2)

AppendArray( variable, byte )

Append the specified byte to an existing array.

This will grow an array for every byte you add.

Example: AppendArray(\$myArray, 88)

IsNumeric( value )

Returns True or False if the specified value is numeric.

Example: \$value = IsNumeric(\$x)

WaitForSpeech( timeOut Seconds, phrases )

Pauses and waits for one of the specified phrases to be spoken.

Returns the phrase that was spoken.

Will return "timeout" if no word is detected in the specified timeout length.

Example: WaitForSpeech(30, "Yes", "No")

Example: \$value = WaitForSpeech(30, "Yes", "No")

Contains( string1, string2 )

Returns TRUE or FALSE if string2 is found within string1.

This search is case insensitive.

Example: \$value = Contains("Cat In The Hat", "Cat")

IndexOf( string1, string2 )

Returns the index within string1 of the first occurrence of string2.

This search is case insensitive.

Example: \$value = IndexOf("Cat In The Hat", "In")

SubString( string1, start, length )

Returns the specified substring within string1.

Example: \$value = SubString("Cat In The Hat", 4, 2)

PlayAudio( filename )

Plays the specified audio file to the default audio device

File formats can be MP3 or WAV

Example: PlayAudio("c:\temp\my Audio.mp3")

StopAudio()

Stops the current audio file that is playing through the default audio device by PlayAudio()

Example: StopAudio()

IsConnected( boardIndex )

Returns TRUE or FALSE if the specified EZ-B board index is connected

Example: \$status = IsConnected(0)

ToString( value )

Converts the parameter into a string by stripping unreadable characters and terminates the end of a string with the first occurrence of a 0x00.

Example: \$string = ToString(\$hexData)

Min( value1, value2 )

Returns the lowest of the two values specified.

Example: \$lowest = Min(3, 5)

Max( value1, value2 )

Returns the highest of the two values specified.

Example: \$highest = Min(3, 5)

ToHex( value )

Converts the integer parameter into a readable hex value string. This will convert the integer 56 into the string "0x39". Great for debugging by

Example: \$hex = ToString(\$hexData)

SetBits( bit7, bit6, bit5, bit4, bit3, bit2, bit1, bit0 )

Returns the value of each bit of a byte. Returns a decimal number byte of the bits. This is a useful function for i2c communication because ma

Example: \$val = SetBits(1, 0, 0, 0, 0, 0, 0, 0)

Example: \$val = SetBits(true, false, false, false, false, false, false, false)

ToBinaryString( value )

Displays the specified value in its binary representation

Example: \$str = ToBinaryString( 254 )

GetBit( value, bit )

Returns the value of the specified bit. The LSB is bit 0, and MSB is bit 7.

Example: \$bit = GetBit(255, 1)

Pause( )

Pauses the script at the location where it is called. To resume the script, another script must call the ControlCommand's Resume function. Ot

Example: Pause()

PushVar( NameSpace, Cell, Value )

Send the value to the EZ-Cloud.

Example: PushVar("DJ Sures", "test", "I am testing")

PullVar( NameSpace, Cell )

Retrieve the value from the EZ-Cloud.

Example: \$x = PullVar("DJ Sures", "test")

LoadProject( filename )

Loads the specified project and replaces the existing project.

Established connections will be maintained. OnConnect scripts within the Connection Control will be executed if a connection is already established.

If no path is specified, this command searches for the file in the default My Documents\EZ-Builder folder.

If no extension is provided, the .ezb default extension is assumed.

For obvious reasons, no further commands following LoadProject() are executed.

Example: LoadProject("MyTest")

Example: LoadProject("MyTest.ezb")

Example: LoadProject("C:\Temp\MyTest.ezb")

## References

### Multiple EZ-B Boards

EZ-Builder supports multiple physical EZ-B Boards connected to your computer. You can specify the board by putting the board number in front of the command.

### ADC Ports

ADC0

ADC1

ADC2

ADC3

ADC4

ADC5

ADC6

ADC7

ADC8

### Servo/Digital/Uart Ports

D0

D1

D2

D3

D4

D5

D6

D7

D8

D9

D10

D11

D12

D13

D14

D15

D16

D17

D18

D19

### Virtual Servo Ports

V0

V1

V2

V3

V4

V5

V6

V7

V8

V9

V10

V11

V12

V13  
V14  
V15  
V16  
V17  
V18  
V19

#### Baud Rates

300  
1200  
2400  
4800  
9600  
19200  
38400  
57600

#### RoboQuad Commands

Stop  
Walk\_Forward  
Right\_Crab\_Walk  
Left\_Crab\_Walk  
Left\_Crab\_Four\_Steps  
Right\_Crab\_Four\_Steps  
Backward\_Four\_Steps  
Walk\_Backward  
Forward\_Four\_Steps  
Rotate\_Counter\_Clockwise  
Counter\_Clockwise\_Four\_Steps  
Rotate\_Clockwise  
Clockwise\_Four\_Steps  
Head\_Up  
Head\_Down  
Head\_Clockwise  
Head\_Counter\_Clockwise  
Top\_Left\_Shuffle  
Top\_Right\_Shuffle  
Bottom\_Left\_Shuffle  
Bottom\_Right\_Shuffle  
Left\_Strafe  
Right\_Strafe  
Left\_Turn\_Roll  
Right\_Turn\_Roll  
Burst  
Single\_Shot  
Stomp\_Walk  
Left\_Legs\_In  
Left\_Legs\_Out  
Left\_Forward\_Leg\_In  
Left\_Forward\_Leg\_Out  
Left\_Backward\_Leg\_In  
Left\_Backward\_Leg\_Out  
Right\_Legs\_In  
Right\_Legs\_Out  
Right\_Forward\_Leg\_In  
Right\_Forward\_Leg\_Out  
Right\_Backward\_Leg\_In  
Right\_Backward\_Leg\_Out  
Program  
Play\_Program  
Program\_Delete\_Last\_Step  
Erase\_Program  
Scan\_Left\_For\_Object  
Scan\_Right\_For\_Object

Smart\_Scan  
Approach\_Nearest\_Object  
Escape\_Walk  
Toggle\_Activity\_Level\_1  
Toggle\_Activity\_Level\_2  
Toggle\_Activity\_Level\_3  
Toggle\_Aggression\_1  
Toggle\_Aggression\_2  
Toggle\_Aggression\_3  
Toggle\_Awareness\_1  
Toggle\_Awareness\_2  
Toggle\_Awareness\_3  
Leg\_Reset  
Full\_Reset  
Volume\_Up  
Volume\_Down  
Guard  
Sleep  
Toggle\_Autonomy  
Toggle\_Sensors  
Twitch  
Surprise  
Wave  
Dizzy  
Attack  
Roar  
Aware\_Stance  
High\_Stance  
Aggressive\_Stance  
Dance\_Demo  
Movement\_Demo  
Leg\_Check

#### RoboSapien Commands

TurnRight  
RightArmUp  
RightArmOut  
TiltBodyRight  
RightArmDown  
RightArmIn  
WalkForward  
WalkReverse  
TurnLeft  
LeftArmUp  
LeftArmOut  
TiltBodyLeft  
LeftArmDown  
LeftArmIn  
Stop  
WakeUp  
Burp  
RightHandStrike  
RightHandSweep  
RightHandStrike2  
HighFive  
Fart  
LeftHandStrike  
LeftHandSweep  
Whistle  
Roar

#### ControlCommand Parameters

All Controls  
PauseOn  
PauseOff

PauseToggle

Auto Position  
AutoPositionStop  
AutoPositionAction  
AutoPositionFrame  
AutoPositionFrameJump

#### Connection

ConnectAll  
Connect0  
Connect1  
Connect2  
Connect3  
Connect4  
DisconnectAll  
Disconnect0  
Disconnect1  
Disconnect2  
Disconnect3  
Disconnect4

#### Camera Device

CameraServoTrackEnable  
CameraServoTrackDisable  
CameraServoTrackToggle  
CameraMovementTrackEnable  
CameraMovementTrackDisable  
CameraMovementTrackToggle  
CameraServoTrackRelativeEnable  
CameraServoTrackRelativeDisable  
CameraServoTrackRelativeToggle  
CameraSnapshot  
CameraTweet (Requires additional parameter)  
CameraViewRealtime  
CameraViewProcessed  
CameraResetQRCode

#### Camera Recording

CameraRecordToggle  
CameraRecordStart  
CameraRecordStop  
CameraRecordPauseOn  
CameraRecordPauseOff  
CameraRecordPauseToggle

#### Camera Device Tracking Modes

CameraView  
CameraColorTracking  
CameraColorTrackingEnable  
CameraColorTrackingDisable  
CameraMotionTracking  
CameraMotionTrackingEnable  
CameraMotionTrackingDisable  
CameraFaceTracking  
CameraFaceTrackingEnable  
CameraFaceTrackingDisable  
CameraGlyphTracking  
CameraGlyphTrackingEnable  
CameraGlyphTrackingDisable  
CameraQRCodeTracking  
CameraQRCodeTrackingEnable  
CameraQRCodeTrackingDisable

CameraCustomTracking  
CameraCustomTrackingEnable  
CameraCustomTrackingDisable  
CameraMultiColorTracking  
CameraMultiColorTrackingEnable  
CameraMultiColorTrackingDisable  
CameraAutoTracking

HTTP Server  
StartServer  
StopServer

#### Scripting Controls

ScriptStart  
ScriptStartWait  
ScriptStop  
ScriptStopAll (Script Manager Only)  
Resume

#### Personality Generator

RunOnce

#### PandoraBots

SetPhrase (Requires additional parameter)

#### Sound Board

Track\_0 ... Track\_50  
Track\_Random  
Stop

#### Sound Board

PauseMS (Requires additional parameter)  
PhrasesShow  
PhrasesHide

#### Recorder

RecorderPlayForward  
RecorderPlayReverse  
RecorderStop  
RecorderRecord

#### Wii Remote

WiiRumbleOn  
WiiRumbleOff  
WiiRumbleAlert  
(Optionally pass an additional parameter for the rumble time in MS, otherwise the default of 150ms is used)

#### Variable Types

Variables are global throughout the entire EZ-Builder environment. The Variable Watcher Control allows you to watch variable values in real-

#### Examples:

String: \$str = "This is a string"  
String Concat: \$str = "fu" + \$bar  
Integer: \$number = 6  
Floating Point: \$dec = 3.14  
Byte: \$byte = 0x52  
Boolean: \$bool = true  
Result Condition: \$result = (\$x > \$y)  
Increment numeric: \$number++  
Decrement numeric: \$number--  
Binary Shift Left: \$x = 123 << 1  
Binary Shift Right: \$x = 123 >> 1

#### Scientific Math Functions



Sin()  
Cos()  
Tan()  
Sec()  
Csc()  
Cot()  
ASin()  
ACos()  
ATan()  
SinH()  
CosH()  
TanH()  
Abs()  
Sqrt()  
Ciel()  
Floor()  
Exp()  
Log10()  
Log()  
Max()  
Min()  
Round()  
E()  
Pi()  
Now()  
Today()

#### DateTime Functions

MinDate()  
MaxDate()  
MonthName()  
AddDays()  
AddMonths()  
AddYears()  
AddHours()  
AddMinute()  
AddSeconds()  
FmtNum()  
FmtDate()

#### Casting Functions

These functions are to cast objects from one datatype to another

To Double: CDBL()  
To Integer: CInt()  
To Long: CLong()  
To Unsigned Integer: CUInt()  
To Unsigned Long: CULong()  
To DateTime: CDateTime()

#### Variable Constants/Reserved Words

These variables are read-only reserved words and cannot be assigned.

\$direction  
\$date  
\$month  
\$year  
\$day  
\$dayName  
\$hour  
\$minute  
\$second  
\$monthName  
\$time  
\$pi

#### Roomba Commands

Init  
EnableSensors  
DisableSensors  
PowerOff  
SpotClean  
Clean  
MaxClean  
DisableAllBrushes  
MainBrushOn  
MainBrushOff  
SideBrushOn  
SideBrushOff  
VacuumOn  
VacuumOff  
SeekDockingStation

#### Music Notes

C1  
Db1  
D1  
Eb1  
E1  
F1  
Gb1  
G1  
Ab1  
A1  
Bb1  
B1  
C2  
Db2  
D2  
Eb2  
E2  
F2  
Gb2  
G2  
Ab2  
A2  
Bb2

#### Auto Start Command Line Options

EZ-Builder supports two command line options for loading a project, and/or executing a script. If you wish to have EZ-Builder auto-load from

Parameter 1: The path and filename of the project file you wish to load upon startup.

Parameter 2: The name of the script control you wish to execute upon startup.

```
"c:\Program Files\EZ-Builder\EZ-Builder.exe "C:\files\My File.ezb"
```

```
"c:\Program Files\EZ-Builder\EZ-Builder.exe "C:\files\My File.ezb" "InitScript"
```