

# NLA50-USB(C#) User Guide

UM-111 Ver.01

## Operation via DLL Software Library

### 一、 Overview

Controlling the USB Stackable Signal Tower via the DLL Library.

### 二、 Development Environment

Item	Contents		Overview
Programming Language	C#		—
Development Software	Visual Studio 2017® 以上		.NET Framework 4.0 or later must be installed.
Required External Files	Software Library(DLL)	USB_NIKKI_Tower.dll	The main library used to control this product.
		USB_NIKKI_Tower.lib	Required files for static library usage (not required for dynamic library usage).
		USB_NIKKI_Tower_DLL.h	Header file declaring the library's functions.
		USB_NIKKI_Definition.h	Header file that defines the parameters.
	Windows Standard	HID.dll	Windows built-in files.
		setupapi.dll	
	Other	vcruntime140.dll	Microsoft Visual C++® 2015 Redistributable is required when developing with the DLL.

### 三、API list

No	Function	Overview
1	Start_HID	Start USB Communication.
2	Stop_HID	End USB Communication.
3	LED_Control	Control of LED Units by Color (Lighting Modes).
4	Led_All_Off	All LED Units Off.
5	Buzzer_Control	Buzzer Control by Selected Mode (Mode/Frequency/Volume).
6	Buzzer_Off	Buzzer Off.
7	Reset_Device	Reset: All LEDs Off and Buzzer Off.
8	Get_Firmware_Version	Get Firmware Version.
9	Get_DLL_Version	Get DLL Version.

#### 四、API Details

##### (一) Start\_HID

Item	Description
Name	int Start_HID()
Function Overview	USB Communication: Connect to USB Stackable Signal Tower.
Return Value	Returns 0 on success. Returns a negative value on error. For details, refer to the “Error Codes” table.
Precautions	This function internally obtains the device handle for USB communication. To release the device handle, the Stop_HID function must be called. This function cannot obtain more than one device handle.

## (二) Stop\_HID

Item	Description
Name	int Stop_HID ()
Function Overview	Terminate USB Communication with the USB Stackable Signal Tower.
Return Value	Returns 0 on success. Returns a negative value on error. For details, refer to the “Error Codes” table.

### (三)LED\_Control

Item	Description
Name	int LED_Control ( int R, int Y, int G, int B, int W )
Function Overview	Set the lighting mode for each LED color to activate the USB Stackable Signal Tower's LEDs. The buzzer will remain in its current state.
Parameters	R (Red), Y (Yellow), G (Green), B (Blue), W (White): Specify the LED mode to control.
Return Value	Returns a negative value on error. For details, refer to the "Error Codes" table.
Precautions	The following function must be called before calling this function:「Start_HID」.
Program example	<pre>int result = DLL_Function.Start_HID(); if(result==0) { DLL_Function.LED_Control(Parameter.RLED_Mode0, Parameter.YLED_Mode1,Parameter.GLED_Mode2, Parameter.BLED_Mode3, Parameter.WLED_Mode5); //RedMode0 , YellowMode3, GreenMode3, BlueMode0, WhiteMode0 } DLL_Function.Stop_HID();</pre>

#### (四) Led\_All\_Off

Item	Description
Name	int Led_All_Off ()
Function Overview	Turn off all LEDs on the USB Stackable Signal Tower.
Parameters	The following function must be called before calling this function:「Start_HID」.
Program example	<pre>int result = DLL_Function.Start_HID(); if(result==0) { DLL_Function.Led_All_Off(); } DLL_Function.Stop_HID();</pre>

## (五) Buzzer\_Control

Item	Description
Name	int Buzzer_Control(int mode, int f1, int f2, int volume);
Function Overview	Specify the buzzer mode, frequency 1, and frequency 2 to sound the buzzer on the USB Stackable Signal Tower. The LED units will remain in their current state.
Parameters	mode: Select the buzzer mode f1: Select frequency 1 f2: Select frequency 2 volume: Select the volume Specifies how the buzzer operates.
Return Value	Returns a negative value on error. For details, refer to the "Error Codes" table.
Precautions	The following function must be called before calling this function:「Start_HID」.
Program example	<pre>int result = DLL_Function.Start_HID(); if(result==0) { DLL_Function.Buzzer_Control(Parameter.Buz_Mode3, Parameter.Buz_F1_1864_7,Parameter.Buz_F2_2793_8, Parameter.Volume1); } DLL_Function.Stop_HID();</pre>

## (六) Buzzer\_Off

Item	Description
Name	int Buzzer_Off()
Function Overview	Stop the buzzer on the USB Stackable Signal Tower.
Precautions	The following function must be called before calling this function:「Start_HID」.
Program example	<pre>int result = DLL_Function.Start_HID(); if(result==0) { DLL_Function.Buzzer_Off(); } DLL_Function.Stop_HID();</pre>



### (七)Reset\_Device

Item	Description
Name	int Reset_Device()
Function Overview	Turn off all LED units and stop the buzzer.
Precautions	The following function must be called before calling this function:「Start_HID」。
Program example	<pre>int result = DLL_Function.Start_HID(); if(result==0) { DLL_Function.Reset_Device(); } DLL_Function.Stop_HID();</pre>

## (八)Get\_Firmware\_Version

Item	Description
Name	DLL_Function.Get_Firmware_Version(bufferPtr, bufferSize);
Function Overview	Retrieve the firmware version of the currently connected USB Stackable Signal Tower.
Return Value	<p>Front 8 bits = Major version  Middle 4 bits = Minor version  Last 4 bits = Revision version  All values are represented in HEX.  (例 : 00 00 00 01 00 00 00 00 = Ver1.0, 00 00 01 02 00 03 00 04 = Ver12.3.4),  Returns a negative value on error.  For details, refer to the "Error Codes" table.</p>
Precautions	<p>The following function must be called before calling this function:「Start_HID」.  "BufferPtr" must be greater than 8."bufferSize" must be equal to Buffer. LEDs and the buzzer will be reset.</p>

Program example	<pre>GCHandle handle = GCHandle.Alloc(buffer, GCHandleType.Pinned); IntPtr bufferPtr = handle.AddrOfPinnedObject(); try {     IntPtr retPtr =     DLL_Function.Get_Firmware_Version(bufferPtr, bufferSize);      string resultStr = Marshal.PtrToStringAnsi(retPtr);     Console.Write("Firmware version (hex): ");     for (int i = 0; i &lt; 8; i++)     {         Console.Write(\$"{buffer[i]:X2} ");     }     Console.WriteLine(); } finally {     handle.Free(); } //Get_Firmware_Version</pre>
-----------------	---

### (九) Get\_Dll\_Version

Item	Description
Name	const char* Get_Dll_Version()
Function Overview	Retrieve DLL Version.
Return Value	Example: 1.0 → returns the string "1.0".
Program example	<pre>IntPtr ptr = DLL_Function.Get_Dll_Version(); string DLLVer = Marshal.PtrToStringAnsi(ptr); Console.WriteLine("DLL Version: " + DLLVer); // Get DLL Version</pre>

## 五、Parameters

### (一) Color of the Controlled LED Unit

Specify the LED color and mode to control using the following parameters.

Control Items	Input	
Controlled LED Color	Parameters	Value
Red Mode0(OFF) Yellow Mode0(OFF) Green Mode0(OFF) Blue Mode0(OFF) White Mode0(OFF)	RLED_Mode0 YLED_Mode0 GLED_Mode0 BLED_Mode0 WLED_Mode0	0
Red Mode1 Yellow Mode1 Green Mode1 Blue Mode1 White Mode1	RLED_Mode1 YLED_Mode1 GLED_Mode1 BLED_Mode1 WLED_Mode1	1
Red Mode2 Yellow Mode2 Green Mode2 Blue Mode2 White Mode2	RLED_Mode2 YLED_Mode2 GLED_Mode2 BLED_Mode2 WLED_Mode2	2
Red Mode3 Yellow Mode3 Green Mode3 Blue Mode3 White Mode3	RLED_Mode3 YLED_Mode3 GLED_Mode3 BLED_Mode3 WLED_Mode3	3

Control Items	Input	
Controlled LED Color	Parameters	Value
Red Mode4 Yellow Mode4 Green Mode4 Blue Mode4 White Mode4	RLED_Mode4 YLED_Mode4 GLED_Mode4 BLED_Mode4 WLED_Mode4	4
Red Mode5 Yellow Mode5 Green Mode5 Blue Mode5 White Mode5	RLED_Mode5 YLED_Mode5 GLED_Mode5 BLED_Mode5 WLED_Mode5	5
Red Mode6 Yellow Mode6 Green Mode6 Blue Mode6 White Mode6	RLED_Mode6 YLED_Mode6 GLED_Mode6 BLED_Mode6 WLED_Mode6	6
Red Mode Hold Yellow Mode Hold Green Mode Hold Blue Mode Hold White Mode Hold	RLED_Keep YLED_Keep GLED_Keep BLED_Keep WLED_Keep	7
Associated API		
LED_Control		

## (二) Buzzer Mode

Specify the Buzzer Mode to control using the following parameters.

Control Items	Input	
Controlled Buzzer	Parameters	Value
Buzzer Mode 0	Buz_Mode0	0
Buzzer Mode 1	Buz_Mode1	1
Buzzer Mode 2	Buz_Mode2	2
Buzzer Mode 3	Buz_Mode3	3
Buzzer Mode 4	Buz_Mode4	4
Buzzer Mode 5	Buz_Mode5	5
Buzzer Mode 6	Buz_Mode6	6
Buzzer Mode HOLD	Buz_Keep	7
Associated API		
Buzzer_Control		

### (三) Buzzer pitch

Specify the Sound 1 and Sound 2 pitch using the following parameters.

Sound 1/2		Input	
Pitch	Frequency (reference value)	Parameters	Value
(Stop)	—	Buz_F1_OFF, Buz_F2_OFF	1
A6	1760.0 Hz	Buz_F1_1760、Buz_F2_1760	2
B $\flat$ 6	1864.7 Hz	Buz_F1_1864_7、Buz_F2_1864_7	3
B6	1975.5 Hz	Buz_F1_1975_5、Buz_F2_1975_5	4
C7	2093.0 Hz	Buz_F1_2093、Buz_F2_2093	5
D $\flat$ 7	2217.5 Hz	Buz_F1_2217_5、Buz_F2_2217_5	6
D7	2349.3 Hz	Buz_F1_2349_3、Buz_F2_2349_3	7
E $\flat$ 7	2489.0 Hz	Buz_F1_2480、Buz_F2_2480	8
E7	2637.0 Hz	Buz_F1_2637、Buz_F2_2637	9
F7	2793.8 Hz	Buz_F1_2793_8、Buz_F2_2793_8	10
G $\flat$ 7	2960.0 Hz	Buz_F1_2960、Buz_F2_2960	11
G7	3136.0 Hz	Buz_F1_3136、Buz_F2_3136	12
A $\flat$ 7	3322.4 Hz	Buz_F1_3322_4、Buz_F2_3322_4	13
A7	3520.0 Hz	Buz_F1_3520、Buz_F2_3520	14
Associated API			
Buzzer_Control			



#### (四) Buzzer Volume

Specify the volume using the following parameters.

Control Items	Input	
Select Volume	Parameters	Value
Volume OFF	Volume1	1
Volume 60%	Volume2	2
Volume 80%	Volume3	3
Volume 100%	Volume4	4
Volume HOLD	Vol_Keep	5
Associated API		
Buzzer_Control		

## 六、Error

### (一)Error list

Macro string	Description	Value
ERR_NOEXIST	Could not detect the USB signal tower. It is necessary to check the connection.	-1
ERR_LOCKED	The USB signal tower was detected, however, it was occupied by another program so communication could not be established.	-2
ERR_CONNECTION	A connection has not been established. It is necessary to call 「Start_HID」.	-3
ERR_PARAMETER	An out-of-range value was specified for a parameter. It is necessary to check the parameter.	-4
ERR_SEND_FAIL	Cannot send (e.g., due to connection interruption during transmission).	-5
ERR_FIRMRECV_FAIL	Cannot send or receive (e.g., due to connection interruption during transmission).	-6
ERR_DLL_LINK	setupapi.dll or HID.DLL are not installed, so it is necessary to acquire these.	-7

(二) List of APIs that return an error

API	Errors that may be returned
Start_HID Stop_HID	ERR_NOEXIST、 ERR_LOCKED、 ERR_CONNECTION、 ERR_DLL_LINK
LED_Control Buzzer_Control	ERR_SEND_FAIL、 ERR_PARAMETER
Reset_Device	ERR_SEND_FAIL
Get_Firmware_Version	ERR_CONNECTION、 ERR_FIRMRECV_FAIL、 ERR_PARAMETER

## 七、Mode Interval Time

### (一) LED Interval Frequency

MODE0	OFF							
MODE1	ON	OFF	ON	OFF	ON	OFF	ON	OFF
MODE2	ON	ON	OFF	OFF	ON	ON	OFF	OFF
MODE3	ON	ON	ON	ON	OFF	OFF	OFF	OFF
MODE4	ON	ON	ON	ON	ON	ON	ON	ON
time	0.125	0.125	0.125	0.125	0.125	0.125	0.125	0.125
	1 Second							

MODE5	0.08sec LED On	0.17sec LED OFF	0.08sec LED On	0.67sec LED OFF
MODE6	0.1sec LED On	0.4sec LED OFF	0.1sec LED On	0.4sec LED OFF
time	1 Second			

## (二) Buzzer Interval Frequency

MODE0	OFF							
MODE1	ON	OFF	ON	OFF	ON	OFF	ON	OFF
MODE2	ON	ON	OFF	OFF	ON	ON	OFF	OFF
MODE3	ON	ON	ON	ON	OFF	OFF	OFF	OFF
MODE4	ON	ON	ON	ON	ON	ON	ON	ON
time	0.125	0.125	0.125	0.125	0.125	0.125	0.125	0.125
	1 Second							

MODE5	0.08sec On	0.17sec OFF	0.08sec On	0.67sec OFF
MODE6	0.1sec On	0.4sec OFF	0.1sec On	0.4sec OFF
time	1 Second			

## 八、USB Communication Protocol

### (一)Communication Protocol Data Format

1st (0XFF)	MODE	Red	Yellow	Green	Blue	White	Volume	Buzzer Mode	Buzzer Pitch	CHECK SUM
1bytes	2bytes L	2bytes H	3bytes L	3bytes H	4bytes L	4bytes H	5bytes	6bytes	7bytes	8bytes

### (二)Checksum Calculation :

The checksum is calculated by summing data[1] through data[6] and applying  $\&= 0xFF$ .

The resulting value is stored in data[7] as the checksum.

# 日機株式会社

〒550-0003 大阪市西区京町堀二丁目 6-28

TEL:06-6225-2855 FAX:06-6225-2856

<https://www.nikki-tr.co.jp> e-mail:contactus@nikki-tr.co.jp