

# **LPU23X tools API**

## **user manual**

### **V4.0**

## table of contents

API Basic information.....	6
The exported functions of API( tg_lpu237_tools.dll ).....	6
The definition of return value.....	9
The definition of Interface.....	9
The definition of language.....	9
The definition of lbutton mode.....	10
The definition of callback function.....	11
type_lpu237_tools_callback_get_parameter.....	11
type_lpu237_tools_callback_set_parameter.....	11
The basic programming sequence.....	12
The important component.....	12
LPU237_tools_on.....	13
Prototype.....	13
parameters.....	13
return.....	13
LPU237_tools_off.....	14
Prototype.....	14
parameters.....	14
return.....	14
LPU237_tools_get_list_w.....	15
Prototype.....	15
parameters.....	15
return.....	15
LPU237_tools_open_w.....	17
Prototype.....	17
parameters.....	17
return.....	17
LPU237_tools_close.....	18
Prototype.....	18
parameters.....	18
return.....	18
LPU237_tools_msr_get_id.....	19
Prototype.....	19
parameters.....	19
return.....	19
LPU237_tools_msr_start_get_setting.....	20
Prototype.....	20

parameters.....	20
return.....	20
LPU237_tools_msr_start_set_setting.....	21
Prototype.....	21
parameters.....	21
return.....	21
LPU237_tools_msr_start_get_setting_except_combination.....	22
Prototype.....	22
parameters.....	22
return.....	22
LPU237_tools_msr_start_set_setting_except_combination.....	23
Prototype.....	23
parameters.....	23
return.....	23
LPU237_tools_msr_get_active_and_valied_interface.....	24
Prototype.....	24
parameters.....	24
return.....	24
LPU237_tools_msr_set_interface.....	25
Prototype.....	25
parameters.....	25
return.....	25
LPU237_tools_msr_set_interface_to_device_and_apply.....	26
Prototype.....	26
parameters.....	26
return.....	26
LPU237_tools_msr_get_buzzer.....	27
Prototype.....	27
parameters.....	27
return.....	27
LPU237_tools_msr_set_buzzer.....	28
Prototype.....	28
parameters.....	28
return.....	28
LPU237_tools_msr_get_language.....	29
Prototype.....	29
parameters.....	29
return.....	29
LPU237_tools_msr_set_language.....	30
Prototype.....	30

parameters.....	30
return.....	30
LPU237_tools_msr_get_track_status.....	31
Prototype.....	31
parameters.....	31
return.....	31
LPU237_tools_msr_set_track_status.....	32
Prototype.....	32
parameters.....	32
return.....	32
LPU237_tools_msr_get_private_tag.....	33
Prototype.....	33
parameters.....	33
return.....	33
LPU237_tools_msr_set_private_tag.....	34
Prototype.....	34
parameters.....	34
return.....	34
LPU237_tools_msr_get_ibutton_mode.....	35
Prototype.....	35
parameters.....	35
return.....	35
LPU237_tools_msr_set_ibutton_mode.....	36
Prototype.....	36
parameters.....	36
return.....	36
LPU237_tools_msr_get_ibutton_tag.....	37
Prototype.....	37
parameters.....	37
return.....	37
LPU237_tools_msr_set_ibutton_tag.....	38
Prototype.....	38
parameters.....	38
return.....	38
LPU237_tools_msr_set_default.....	39
Prototype.....	39
parameters.....	39
return.....	39
LPU237_tools_msr_is_support_msr.....	40
Prototype.....	40

parameters.....	40
return.....	40
LPU237_tools_msr_is_support_ibutton.....	41
Prototype.....	41
parameters.....	41
return.....	41
Log file.....	42
Log file generation condition.....	42
the folder of Log file .....	42
lpu230_tools.ini file.....	42
History.....	43

This document describes how to use the Application Programming Interface (API) to change the settings of the LPU237 card reader (MSR). This API needs that NDM( The Next Device Manager ) is running.

## API Basic information.

	value	etc
folder	%ProgramFiles(x86)%\Easyset\lpu230\bin\components\Wx86	32bit mapper on 64 bits windows.
	%ProgramFiles%\Easyset\lpu230\bin\components\Wx64	64bit mapper on 64 bits windows.
	%ProgramFiles%\Easyset\lpu230\bin\components\Wx86	32bit mapper on 32 bits windows.
File name	tg_lpu237_tools.dll	version 1.0.
type	win32 regular dynamic linked library( dll )	
Sub component	tg_rom.dll is the sun component of tg_lpu237_tools.dll.	tg_rom.dll and tg_lpu237_tools.dll exist in the same folder.

## The exported functions of API( tg\_lpu237\_tools.dll )

The exported name	prototype	Description.
LPU237_tools_on	DWORD WINAPI LPU237_tools_on()	Initialize the inner worker of dll.
LPU237_tools_off	DWORD WINAPI LPU237_tools_off()	terminates the inner worker of dll.
LPU237_tools_get_list_w	DWORD WINAPI LPU237_tools_get_list_w ( WCHAR *ssDevPaths )	gets the connected MSR list. unicode type.
LPU237_tools_open_w	HANDLE WINAPI LPU237_tools_open_w ( CONST WCHAR *sDevPath )	open the channel of MSR. unicode type.
LPU237_tools_close	DWORD WINAPI LPU237_tools_close ( HANDLE hDev )	close the channel of MSR.
LPU237_tools_msr_get_id	DWORD WINAPI LPU237_tools_msr_get_id ( HANDLE hDev, BYTE *sId )	Gets a device ID(16 bytes).
LPU237_tools_msr_start_get_setting	DWORD WINAPI LPU237_tools_msr_start_get_setting(const BYTE* sId, type_lpu237_tools_callback_get_parameter cb, void* pUser);	starts loading the setting values of MSR by async callback.
LPU237_tools_msr_start_set_setting	DWORD WINAPI LPU237_tools_msr_start_set_setting(const BYTE* sId, type_lpu237_tools_callback_set_parameter cb, void* pUser);	starts saving the setting values of MSR by async callback.
LPU237_tools_msr_start_get_setting_except_combination	DWORD WINAPI LPU237_tools_msr_start_get_setting_except_combination(const BYTE* sId, type_lpu237_tools_callback_get_parameter cb, void* pUser);	starts loading the setting values of MSR by async callback. (except combination setting)
LPU237_tools_msr_start_set_setting_except_combination	DWORD WINAPI LPU237_tools_msr_start_set_setting_except_combination	starts saving the setting values of MSR by async callback. (except combination setting)

	(const BYTE* sld, type_lpu237_tools_callback_set_parameter cb, void* pUser);	combination setting)
LPU237_tools_msr_get_active_and_valied_interface	DWORD WINAPI LPU237_tools_msr_get_active_and_valied_interface(HANDLE hDev, BYTE* s_inteface)	Get all valid MSR interfaces and active interface.
LPU237_tools_msr_set_interface	DWORD WINAPI LPU237_tools_msr_set_interface(HANDLE hDev, BYTE c_inteface)	Set the active interface.(not be applied)
LPU237_tools_msr_set_interface_to_device_and_apply	DWORD WINAPI LPU237_tools_msr_set_interface_to_device_and_apply(HANDLE hDev, BYTE* pc_inteface)	Set the active interface. it is applied to MSR.
LPU237_tools_msr_get_buzzer	DWORD WINAPI LPU237_tools_msr_get_buzzer(HANDLE hDev, BYTE* pc_on)	Get the buzzer status.
LPU237_tools_msr_set_buzzer	DWORD WINAPI LPU237_tools_msr_set_buzzer(HANDLE hDev, BYTE c_on)	Set the buzzer status.
LPU237_tools_msr_get_language	DWORD WINAPI LPU237_tools_msr_get_language(HANDLE hDev, BYTE* pc_lang)	Get the current language.
LPU237_tools_msr_set_language	DWORD WINAPI LPU237_tools_msr_set_language(HANDLE hDev, BYTE c_lang)	Set the current language.
LPU237_tools_msr_get_track_status	DWORD WINAPI LPU237_tools_msr_get_track_status(HANDLE hDev, BYTE* s_status_3_byte)	Get the status of each ISO track.
LPU237_tools_msr_set_track_status	DWORD WINAPI LPU237_tools_msr_set_track_status(HANDLE hDev, const BYTE* s_status_3_byte)	Set the status of each ISO track.
LPU237_tools_msr_get_private_tag	DWORD WINAPI LPU237_tools_msr_get_private_tag(HANDLE hDev, DWORD dw_track_zero_base, BYTE b_prefix, BYTE* s_tag)	Get the pre/postfix of each ISO track.
LPU237_tools_msr_set_private_tag	DWORD WINAPI LPU237_tools_msr_set_private_tag(HANDLE hDev, DWORD dw_track_zero_base, BYTE b_prefix, const BYTE* s_tag, DWORD dw_tag)	Set the pre/postfix of each ISO track.
LPU237_tools_msr_get_ibutton_mode	DWORD WINAPI LPU237_tools_msr_get_ibutton_mode(HANDLE hDev, BYTE* pc_mode)	Get the ibutton mode.
LPU237_tools_msr_set_ibutton_mode	DWORD WINAPI LPU237_tools_msr_set_ibutton_mode(HANDLE hDev, BYTE c_mode)	Set the ibutton mode.

LPU237_tools_msr_get_ibutton_tag	DWORD WINAPI LPU237_tools_msr_get_ibutton_tag(HANDLE hDev, BYTE b_remove, BYTE b_prefix, BYTE* s_tag)	Get the pre/postfix of ibutton.
LPU237_tools_msr_set_ibutton_tag	DWORD WINAPI LPU237_tools_msr_set_ibutton_tag(HANDLE hDev, BYTE b_remove, BYTE b_prefix, const BYTE* s_tag, DWORD dw_tag)	Set the pre/postfix of ibutton.
LPU237_tools_msr_set_default	DWORD WINAPI LPU237_tools_msr_set_default(HANDLE hDev)	set to the default.
LPU237_tools_msr_is_support_msr	DWORD WINAPI LPU237_tools_msr_is_support_msr(HANDLE hDev, BYTE* pc_support)	if or not support a magnetic card reading.
LPU237_tools_msr_is_support_ibutton	DWORD WINAPI LPU237_tools_msr_is_support_ibutton(HANDLE hDev, BYTE* pc_support)	if or not support a ibutton reading.

- the exported function must be used in the unicode project. If you use in MBSC project, the function with "\_w" have to be converted to unicode.



## The defintion of return value.

This values are the return of API function. Or the second parameter of callback funtion. The callback function is parameter of LPU237\_tools\_msr\_start\_x() .

Symbol	Hexcimal value( double word )	Description
LPU237_TOOLS_RESULT_SUCCESS	0x00000000	Success processing
LPU237_TOOLS_RESULT_ERROR	0xFFFFFFFF	Error
LPU237_TOOLS_RESULT_CANCEL	0xFFFFFFFFE	Processing is canceled.
LPU237_TOOLS_RESULT_NO_MSR	0xFFFFFFFFB	None MSR

## The defintion of Interface

Symbol	Hexcimal value( double word )	Description
LPU237_TOOLS_INF_USBKB	0	USB keyboard interface.
LPU237_TOOLS_INF_USBHID	1	USB hid interface.
LPU237_TOOLS_INF_USBVCOM	2	USB vital COM interface.(new v4.0)
LPU237_TOOLS_INF_UART	10	USB uart interface.

## The defintion of language

Symbol	decimal value( double word )	Description
LPU237_TOOLS_LANG_USA_ENGLISH	0	US keyboard.
LPU237_TOOLS_LANG_SPANISH	1	Spanish keyboard
LPU237_TOOLS_LANG_DANISH	2	Denish keyboard
LPU237_TOOLS_LANG_FRENCH	3	Frenich keyboard
LPU237_TOOLS_LANG_GERMAN	4	Germany keyboard
LPU237_TOOLS_LANG_ITALIAN	5	Italian keyboard
LPU237_TOOLS_LANG_NORWEGIAN	6	Norwegian keyboard
LPU237_TOOLS_LANG_SWEDISH	7	Swedish keyboard
LPU237_TOOLS_LANG_UK_ENGLISH	8	U.K keyboard
LPU237_TOOLS_LANG_ISRAEL	9	Hebrew keyboard
LPU237_TOOLS_LANG_TURKIYE	10	Turkiye keyboard

## The defintion of Ibutton mode

Symbol	decimal value( double word )	Description
LPU237_TOOLS_IBUTTON_MODE_NONE	0	when a ibutton is disconnected, sends the setting value(firmware v5.21 , 3.22 or later) or nothing.
LPU237_TOOLS_IBUTTON_MODE_ZEROS	1	when a ibutton is disconnected, sends "0000000000000000". zeros 16 times
LPU237_TOOLS_IBUTTON_MODE_F12	2	when a ibutton is disconnected, sends F12 key.
LPU237_TOOLS_IBUTTON_MODE_ZERO7	3	when a ibutton is disconnected, sends "0000000" zeros 7 times
LPU237_TOOLS_IBUTTON_MODE_ADDMIT	4	when a ibutton is disconnected, send the data that is defined by addmit codestick spec.

example <https://blog.naver.com/elpusk/222928056691>

## The defintion of callback function.

### ***type\_lpu237\_tools\_callback\_get\_parameter***

The second parameter of LPU237\_tools\_msr\_start\_get\_setting() and LPU237\_tools\_msr\_start\_get\_setting\_except\_combination(). The prototype is

DWORD WINAPI lpu237\_tools\_callback\_get\_parameter(void\*, DWORD, DWORD, DWORD)

- 1'st parameter – user memory pointer
- 2'nd parameter – the result of current step.( LPU237\_TOOLS\_RESULT\_SUCCESS, LPU237\_TOOLS\_RESULT\_ERROR or LPU237\_TOOLS\_RESULT\_CANCEL )
- 3'rd parameter – the index of current step. This value is 0~ (4'th parameter-1) .
- 4'th parameter – the numnber of total step.

This callback is executed by the inner worker thread(WTH).

### ***type\_lpu237\_tools\_callback\_set\_parameter***

The second parameter of LPU237\_tools\_msr\_start\_set\_setting() and LPU237\_tools\_msr\_start\_set\_setting\_except\_combination(). The prototype is

DWORD WINAPI lpu237\_tools\_callback\_set\_parameter(void\*, DWORD, DWORD, DWORD)

- 1'st parameter – user memory pointer
- 2'nd parameter – the result of current step.( LPU237\_TOOLS\_RESULT\_SUCCESS, LPU237\_TOOLS\_RESULT\_ERROR or LPU237\_TOOLS\_RESULT\_CANCEL )
- 3'rd parameter – the index of current step. This value is 0~ (4'th parameter-1) .
- 4'th parameter – the numnber of total step.
- 5'th parameter – reserved the future.

This callback is executed by the inner worker thread(WTH).

## The basic programming sequence.

1. Call LPU237\_tools\_on() - initialize dll.
2. Call LPU237\_tools\_get\_list() - gets the MSR path list.
3. Call LPU237\_tools\_open() - open a channel.
4. Call LPU237\_tools\_msr\_get\_id() - gets 16 bytes ID of MSR.
5. Call LPU237\_tools\_msr\_start\_get\_setting() or  
LPU237\_tools\_msr\_start\_get\_setting\_except\_combination() - starts the loading MSR parameters.  
And waits the end of loading.
6. Call LPU237\_tools\_msr\_get\_x() - gets a setting value.
7. Call LPU237\_tools\_msr\_set\_x() - changes a setting value.
8. Call LPU237\_tools\_msr\_start\_set\_setting() or  
LPU237\_tools\_msr\_start\_set\_setting\_except\_combination() - starts the saving MSR parameters. And  
waits the end of loading.
9. Call LPU237\_tools\_close(). - close the channel.
10. Call LPU237\_tools\_off(). - terminate the inner worker of dll.

## The important component.

- **Inner worker thread** - API use WTH(inner worker thread) for supporting async-IO.  
LPU237\_tools\_on() and LPU237\_tools\_off() are that starts or terminates WTH. Therefore  
LPU237\_tools\_on() must be called before another function. And LPU237\_tools\_off() have to be  
called after terminating your work.

## LPU237\_tools\_on

Creates and executes WTH. You must be called before another function.

If this function is called in DllMain(), it may occur the deadlock.

### ***Prototype***

DWORD WINAPI LPU237\_tools\_on()

### ***parameters***

none

### ***return***

condition	value	etc
success	LPU237_TOOLS_RESULT_SUCCESS	

## LPU237\_tools\_off

Terminates WTH. You must call this function before terminating main program.

### ***Prototype***

DWORD WINAPI LPU237\_tools\_off()

### ***parameters***

none

### ***return***

condition	value	etc
success	LPU237_TOOLS_RESULT_SUCCESS	Always

## LPU237\_tools\_get\_list\_w

Get the path list of MSR ( VID 0x134B, PID 0x0206). Path list is multi-zero string type.

### ***Prototype***

DWORD WINAPI LPU237\_tools\_get\_list\_w( WCHAR \*ssDevPaths ) - unicode version.

### ***parameters***

- ssDevPaths – [in/out] the buffer that saves MSR path. If this is zero, return the size of buffer.(byte unit)

### ***return***

condition	value	etc
ssDevPaths is 0	the buffer that saves MSR path	Byte unit
SsDevPaths isn't zero and process is success.	The number of MSR.	ssDevPaths is multi-zero strinf type.
error	LPU237_TOOLS_RESULT_ERROR	

- zero string – A way to mark the end of a string as 0 . In Unicode, each character is 2 bytes, so 0 is entered twice. (Standard form of string in Windows API and C language)
- multi-zero string - A method of concatenating multiple zero strings consecutively and adding 0 to the end to indicate the end of the strings. In this method, 0 for the last string and 0 to indicate the end of the string appear consecutively at the end. In Unicode, each character is 2 bytes, so the last zeros goes in 4 times.

ex) if the number of MSR is two and each MSR path are "ab" and "12", ssDevPaths value is unicode version

offset	value		etc
0	0x61	Unicode 'a'	
1	0x00		
2	0x62	Unicode 'b'	
3	0x00		
4	0x00	Unicode NULL	The end of "ab" string
5	0x00		
6	0x31	Unicode '1'	
7	0x00		
8	0x32	Unicode '2'	
9	0x00		
10	0x00	Unicode NULL	The end of "12" string
11	0x00		
12	0x00	Unicode NULL	The end of multi-zero string.
13	0x00		



## LPU237\_tools\_open\_w

Open channel.

### ***Prototype***

HANDLE WINAPI LPU237\_tools\_open\_w( CONST WCHAR \*sDevPath ) - unicode version.

### ***parameters***

- sDevPath – [in] MSR path, zero-string type.

### ***return***

condition	value	etc
success	The handle of MSR.	
error	INVALID_HANDLE_VALUE	long type address value(-1. Defined by Microsoft )

## LPU237\_tools\_close

Close channel.

### ***Prototype***

DWORD WINAPI LPU237\_tools\_close( HANDLE hDev )

### ***parameters***

- hDev – [in] MSR handle( from LPU237\_TOOLS\_open() )

### ***return***

condition	value	etc
success	LPU237_TOOLS_RESULT_SUCCESS	
error	LPU237_TOOLS_RESULT_ERROR	

## LPU237\_tools\_msr\_get\_id

Get 16 bytes ID.

### ***Prototype***

DWORD WINAPI LPU237\_tools\_msr\_get\_id( HANDLE hDev, BYTE \*sld )

### ***parameters***

- hDev – [in] MSR handle( from LPU237\_TOOLS\_open() )
- sld – [out] the buffer point that saves th 16 byets ID.

### ***return***

condition	value	etc
success	LPU237_TOOLS_RESULT_SUCCESS	
success	If sld is zero, return 16.	
error	LPU237_TOOLS_RESULT_ERROR	

Each MSR have a unique ID.

## LPU237\_tools\_msr\_start\_get\_setting

Starts loading the setting value by callback async.

### ***Prototype***

DWORD WINAPI LPU237\_tools\_msr\_start\_get\_setting(const BYTE\* sId,  
type\_lpu237\_tools\_callback\_get\_parameter cb, void\* pUser)

### ***parameters***

- sId – [in] MSR ID. (from LPU237\_tools\_msr\_get\_id() )
- cb – [in] callback function. This callback is called by WTH, and announce the status of loading data.
- pUser – [in] 1'st parameter of cb.

### ***return***

condition	value	etc
success	LPU237_TOOLS_RESULT_SUCCESS	
error	LPU237_TOOLS_RESULT_ERROR	
error	LPU237_TOOLS_RESULT_NO_MSR	None MSR that have the given ID.

## LPU237\_tools\_msr\_start\_set\_setting

Starts saving the setting value by callback async.

### ***Prototype***

DWORD WINAPI LPU237\_tools\_msr\_start\_set\_setting(const BYTE\* sId,  
type\_lpu237\_tools\_callback\_set\_parameter cb, void\* pUser)

### ***parameters***

- sId – [in] MSR ID. (from LPU237\_tools\_msr\_get\_id() )
- cb – [in] callback function. This callback is called by WTH, and announce the status of saving data.
- pUser – [in] 1'st parameter of cb.

### ***return***

condition	value	etc
success	LPU237_TOOLS_RESULT_SUCCESS	
error	LPU237_TOOLS_RESULT_ERROR	
error	LPU237_TOOLS_RESULT_NO_MSR	None MSR that have the given ID.

## LPU237\_tools\_msr\_start\_get\_setting\_except\_combination

Starts loading the setting value by callback async except combination parameters. For increasing performance.

### ***Prototype***

DWORD WINAPI LPU237\_tools\_msr\_start\_get\_setting\_except\_combination(const BYTE\* sld, type\_lpu237\_tools\_callback\_get\_parameter cb, void\* pUser)

### ***parameters***

- sld – [in] MSR ID. (from LPU237\_tools\_msr\_get\_id() )
- cb – [in] callback function. This callback is called by WTH, and announce the status of loading data.
- pUser – [in] 1'st parameter of cb.

### ***return***

condition	value	etc
success	LPU237_TOOLS_RESULT_SUCCESS	
error	LPU237_TOOLS_RESULT_ERROR	
error	LPU237_TOOLS_RESULT_NO_MSR	None MSR that have the given ID.

## LPU237\_tools\_msr\_start\_set\_setting\_except\_combination

Starts saving the setting value by callback async except combination parameters. For increasing performance.

### ***Prototype***

DWORD WINAPI LPU237\_tools\_msr\_start\_set\_setting\_except\_combination(const BYTE\* sId, type\_lpu237\_tools\_callback\_set\_parameter cb, void\* pUser)

### ***parameters***

- sId – [in] MSR ID. (from LPU237\_tools\_msr\_get\_id() )
- cb – [in] callback function. This callback is called by WTH, and announce the status of saving data.
- pUser – [in] 1'st parameter of cb.

### ***return***

condition	value	etc
success	LPU237_TOOLS_RESULT_SUCCESS	
error	LPU237_TOOLS_RESULT_ERROR	
error	LPU237_TOOLS_RESULT_NO_MSR	None MSR that have the given ID.

## LPU237\_tools\_msr\_get\_active\_and\_valied\_interface

Get all valid MSR interfaces and active interface.

### ***Prototype***

DWORD WINAPI LPU237\_tools\_msr\_get\_active\_and\_valied\_interface(HANDLE hDev, BYTE\* s\_inteface)

### ***parameters***

- hDev – [in] MSR handle( from LPU237\_TOOLS\_open() )
- s\_inteface – [in/out] the buffer that saves interfaces. 1'st byte is active interface. the supported interfaces starts from 2'nd byte. If this value is zero, return the size of buffer.(byte unit)

### ***return***

condition	value	etc
success	The size of saved interfaces/	If active interface is USB keyboard and , the supported interfaces are USB keyboard, USB hid and Uart, the returned value is 4, the saved interfaces are s_interface[0] is 0 s_interface[0] is 0 s_interface[1] is 1 s_interface[2] is 10
error	LPU237_TOOLS_RESULT_ERROR	



## LPU237\_tools\_msr\_set\_interface

set active interface.

### ***Prototype***

DWORD WINAPI LPU237\_tools\_msr\_set\_interface(HANDLE hDev, BYTE c\_inteface)

### ***parameters***

- hDev – [in] MSR handle( from LPU237\_TOOLS\_open() )
- c\_inteface – [in] active interface ( from LPU237\_tools\_msr\_get\_active\_and\_valied\_interface() )

### ***return***

condition	value	etc
success	LPU237_TOOLS_RESULT_SUCCESS	
error	LPU237_TOOLS_RESULT_ERROR	

## LPU237\_tools\_msr\_set\_interface\_to\_device\_and\_apply

Set active interface.it is applied to MSR. This function will be used to change the current interface temporarily.

### ***Prototype***

DWORD WINAPI LPU237\_tools\_msr\_set\_interface\_to\_device\_and\_apply(HANDLE hDev, BYTE\* pc\_inteface)

### ***parameters***

- hDev – [in] MSR handle( from LPU237\_TOOLS\_open() )
- pc\_inteface – [in/out] the new active interface in "in". The old active interface in "out".

### ***return***

condition	value	etc
success	LPU237_TOOLS_RESULT_SUCCESS	
error	LPU237_TOOLS_RESULT_ERROR	

## LPU237\_tools\_msr\_get\_buzzer

Get buzzer status.

### ***Prototype***

DWORD WINAPI LPU237\_tools\_msr\_get\_buzzer(HANDLE hDev, BYTE\* pc\_on)

### ***parameters***

- hDev – [in] MSR handle( from LPU237\_TOOLS\_open() )
- pc\_on – [in/out] the one byte buffer that saved the buzzer status.

### ***return***

condition	value	etc
success	LPU237_TOOLS_RESULT_SUCCESS	If the save value is zero in pc_on, the status of buzzer is off. If the save value is one in pc_on, the status of buzzer is on.
error	LPU237_TOOLS_RESULT_ERROR	

## LPU237\_tools\_msr\_set\_buzzer

Set buzzer status.

### ***Prototype***

DWORD WINAPI LPU237\_tools\_msr\_set\_buzzer(HANDLE hDev, BYTE c\_on)

### ***parameters***

- hDev – [in] MSR handle( from LPU237\_TOOLS\_open() )
- c\_on – [in] the new status of buzzer . 0 - off, 1 - on.

### ***return***

condition	value	etc
success	LPU237_TOOLS_RESULT_SUCCESS	
error	LPU237_TOOLS_RESULT_ERROR	

## LPU237\_tools\_msr\_get\_language

gets the language

### ***Prototype***

DWORD WINAPI LPU237\_tools\_msr\_get\_language(HANDLE hDev, BYTE\* pc\_lang)

### ***parameters***

- hDev – [in] MSR handle( from LPU237\_TOOLS\_open() )
- pc\_lang – [in/out] the one byte buffer that saved the language.  
(LPU237\_TOOLS\_LANG\_USA\_ENGLISH ~ LPU237\_TOOLS\_LANG\_TURKIYE.)

### ***return***

condition	value	etc
success	LPU237_TOOLS_RESULT_SUCCESS	
error	LPU237_TOOLS_RESULT_ERROR	

## LPU237\_tools\_msr\_set\_language

Sets the language.

### ***Prototype***

DWORD WINAPI LPU237\_tools\_msr\_set\_language(HANDLE hDev, BYTE c\_lang)

### ***parameters***

- hDev – [in] MSR handle( from LPU237\_TOOLS\_open() )
- c\_lang – [in] the new language( LPU237\_TOOLS\_LANG\_USA\_ENGLISH ~ LPU237\_TOOLS\_LANG\_TURKIYE ).

### ***return***

condition	value	etc
success	LPU237_TOOLS_RESULT_SUCCESS	
error	LPU237_TOOLS_RESULT_ERROR	

## LPU237\_tools\_msr\_get\_track\_status

Get the status of each ISO track.

### ***Prototype***

DWORD WINAPI LPU237\_tools\_msr\_get\_track\_status(HANDLE hDev, BYTE\* s\_status\_3\_byte)

### ***parameters***

- hDev – [in] MSR handle( from LPU237\_TOOLS\_open() )
- s\_status\_3\_byte – [in/out] 3 bytes buffer.

### ***return***

condition	value	etc
success	LPU237_TOOLS_RESULT_SUCCESS	s_status_3_byte[0] – the status of ISO1 track. s_status_3_byte[1] – the status of ISO2 track. s_status_3_byte[2] – the status of ISO3 track. value is 1 - enable, 0 - disable.
error	LPU237_TOOLS_RESULT_ERROR	

## LPU237\_tools\_msr\_set\_track\_status

Set the status of each ISO track.

### ***Prototype***

DWORD WINAPI LPU237\_tools\_msr\_set\_track\_status(HANDLE hDev, const BYTE\* s\_status\_3\_byte)

### ***parameters***

- hDev – [in] MSR handle( from LPU237\_TOOLS\_open() )
- s\_status\_3\_byte – [in] 3 bytes buffer. s\_status\_3\_byte[0] - ISO1 track. s\_status\_3\_byte[1] - ISO2 track. s\_status\_3\_byte[2] – ISO3 track. The value is 1 - enable, 0 - disable.

### ***return***

condition	value	etc
success	LPU237_TOOLS_RESULT_SUCCESS	
error	LPU237_TOOLS_RESULT_ERROR	



## LPU237\_tools\_msr\_get\_private\_tag

Get the pre/postfix of each ISO track.

### ***Prototype***

DWORD WINAPI LPU237\_tools\_msr\_get\_private\_tag(HANDLE hDev, DWORD dw\_track\_zero\_base, BYTE b\_prefix, BYTE\* s\_tag)

### ***parameters***

- hDev – [in] MSR handle( from LPU237\_TOOLS\_open() )
- dw\_track\_zero\_base – [in] ISO track. 0 - ISO1, 1 - ISO2, 2 - ISO3.
- b\_prefix – [in] 1 - prefix, 0 - postfix.
- s\_tag – [in/out] the buffer that will be saved the pre/postfix. If this value is zero, returns the size of buffer(byte unit). The save structure - see tag part of <https://blog.naver.com/elpusk/221987287359> . In this description, ignore '[' and ']'(these are separator.)

### ***return***

condition	value	etc
success	The size of saved data. In s_tag.	Unit byte
error	LPU237_TOOLS_RESULT_ERROR	

## LPU237\_tools\_msr\_set\_private\_tag

Set the pre/postfix of each ISO track.

### ***Prototype***

DWORD WINAPI LPU237\_tools\_msr\_set\_private\_tag(HANDLE hDev, DWORD dw\_track\_zero\_base, BYTE b\_prefix, const BYTE\* s\_tag, DWORD dw\_tag)

### ***parameters***

- hDev – [in] MSR handle( from LPU237\_TOOLS\_open() )
- dw\_track\_zero\_base – [in] ISO track. 0 - ISO1, 1 - ISO2, 2 - ISO3.
- b\_prefix – [in] 1 - prefix, 0 - postfix.
- s\_tag – [in] the new pre/postfix, If this value is zero, pre/postfix is removed. s\_tag structure is <https://blog.naver.com/elpusk/221987287359> . In this description, ignore '[' and ']'(these are separator.)
- dw\_tag – [in] the size of s\_tag. If this value is zero, pre/postfix is removed. The max value is 14. (max key is 7).

### ***return***

condition	value	etc
success	LPU237_TOOLS_RESULT_SUCCESS	
error	LPU237_TOOLS_RESULT_ERROR	

## LPU237\_tools\_msr\_get\_ibutton\_mode

Get the ibutton mode

### ***Prototype***

DWORD WINAPI LPU237\_tools\_msr\_get\_ibutton\_mode(HANDLE hDev, BYTE\* pc\_mode)

### ***parameters***

- hDev – [in] MSR handle( from LPU237\_TOOLS\_open() )
- pc\_mode – [in/out] one byte buffer.

### ***return***

condition	value	etc
success	LPU237_TOOLS_RESULT_SUCCESS	pc_mode is LPU237_TOOLS_IBUTTON_MODE_NONE ~ LPU237_TOOLS_IBUTTON_MODE_ADDMIT.
error	LPU237_TOOLS_RESULT_ERROR	

## LPU237\_tools\_msr\_set\_ibutton\_mode

Set the ibutton mode.

### ***Prototype***

DWORD WINAPI LPU237\_tools\_msr\_set\_ibutton\_mode(HANDLE hDev, BYTE c\_mode)

### ***parameters***

- hDev – [in] MSR handle( from LPU237\_TOOLS\_open() )
- c\_mode – [in] the new ibutton mode. LPU237\_TOOLS\_IBUTTON\_MODE\_NONE ~ LPU237\_TOOLS\_IBUTTON\_MODE\_ADDMIT.

### ***return***

condition	value	etc
success	LPU237_TOOLS_RESULT_SUCCESS	
error	LPU237_TOOLS_RESULT_ERROR	

## LPU237\_tools\_msr\_get\_ibutton\_tag

Get the ibutton pre/postfix.

### ***Prototype***

DWORD WINAPI LPU237\_tools\_msr\_get\_private\_tag(HANDLE hDev, DWORD dw\_track\_zero\_base, BYTE b\_prefix, BYTE\* s\_tag)

### ***parameters***

- hDev – [in] MSR handle( from LPU237\_TOOLS\_open() )
- b\_remove – [in] 1 – the pre/postfix of disconnecting ibutton.( only firmware v5.21 ,v3.22 or later ). 0 – the pre/postfix of connecting ibutton.
- b\_prefix – [in] 1 - prefix, 0 - postfix.
- s\_tag – [in/out] the buffer that will be saved the pre/postfix. If this value is zero, returns the size of buffer(byte unit). The save structure - see tag part of <https://blog.naver.com/elpusk/221987287359> . In this description, ignore '[' and ']'(these are separator.)

### ***return***

condition	value	etc
success	The size of s_tag	Unit byte
error	LPU237_TOOLS_RESULT_ERROR	

## LPU237\_tools\_msr\_set\_ibutton\_tag

Set the ibutton pre/postfix.

### ***Prototype***

DWORD WINAPI LPU237\_tools\_msr\_set\_ibutton\_tag(HANDLE hDev, BYTE b\_remove, BYTE b\_prefix, const BYTE\* s\_tag, DWORD dw\_tag)

### ***parameters***

- hDev – [in] MSR handle( from LPU237\_TOOLS\_open() )
- b\_remove – [in] 1 – the pre/postfix of disconnecting ibutton.( only firmware v5.21 ,v3.22 or later ). 0 – the pre/postfix of connecting ibutton.
- b\_prefix – [in] 1 - prefix, 0 - postfix.
- s\_tag – [in] the new pre/postfix, If this value is zero, pre/postfix is removed. s\_tag structure is <https://blog.naver.com/elpusk/221987287359> . In this description, ignore '[' and ']'(these are separator.)
- dw\_tag – [in] the size of s\_tag. If this value is zero, pre/postfix is removed. The max value is 14. (max key is 7).

### ***return***

condition	value	etc
success	LPU237_TOOLS_RESULT_SUCCESS	
error	LPU237_TOOLS_RESULT_ERROR	

## LPU237\_tools\_msr\_set\_default

Set to default.

### ***Prototype***

DWORD WINAPI LPU237\_tools\_msr\_set\_default(HANDLE hDev)

### ***parameters***

- hDev – [in] MSR handle( from LPU237\_TOOLS\_open() )

### ***return***

condition	value	etc
success	LPU237_TOOLS_RESULT_SUCCESS	
error	LPU237_TOOLS_RESULT_ERROR	

## LPU237\_tools\_msr\_is\_support\_msr

if or not support a magnetic card reading.

### ***Prototype***

DWORD WINAPI LPU237\_tools\_msr\_is\_support\_msr(HANDLE hDev, BYTE\* pc\_support)

### ***parameters***

- hDev – [in] MSR handle( from LPU237\_TOOLS\_open() )
- pc\_support – [in/out] one byte buffer

### ***return***

condition	value	etc
success	LPU237_TOOLS_RESULT_SUCCESS	pc_support is 0, not supports MSR pc_support is 1, supports MSR.
error	LPU237_TOOLS_RESULT_ERROR	



## LPU237\_tools\_msr\_is\_support\_ibutton

if or not support a ibutton reading.

### ***Prototype***

DWORD WINAPI LPU237\_tools\_msr\_is\_support\_ibutton(HANDLE hDev, BYTE\* pc\_support)

### ***parameters***

- hDev – [in] MSR handle( from LPU237\_TOOLS\_open() )
- pc\_ support – [in/out] one byte buffer

### ***return***

condition	value	etc
success	LPU237_TOOLS_RESULT_SUCCESS	pc_support is 0, not supports MSR pc_support is 1, supports MSR.
error	LPU237_TOOLS_RESULT_ERROR	

## Log file

API can generate the log file.

It generates in a special condition. tg\_lpu237\_tools.dll generates a new log file when is loaded into memory.

### ***Log file generation condition.***

1. In Logon user document folder, "Easyset\lpu230" folder exist
2. In Logon user document folder, "Easyset\lpu230\log" folder exist.
3. In "Easyset\lpu230" folder, lpu230\_tools.ini exist.
4. In lpu230\_tools.ini , [LogSetting] session exist.
5. In [LogSetting] session, the value of logenable key is 1 .

### ***the folder of Log file***

A log file is generated at "Easyset\lpu230\log". If tg\_lpu237\_tools.dll is loaded to memory at 2017/12/25, PM3H 45MIN 12SEC, log file name is tg\_lpu237\_tools\_071225154512.txt ( at version 1.0, tools071225154512.txt), log file format is text.

### ***lpu230\_tools.ini file***

For setting log-file.

<b>[LogSetting] session</b>	Log setting session
<b>logenable key</b>	1 – generation log. Else not be generated.
<b>loglevel key</b>	This value must be 3.
<b>logtimestemp key</b>	1 - [MM-dd hh:mm:ss] is added to each log line.
<b>logtimetick key</b>	1 – the systemtick is added to each log line.
<b>[control] session</b>	etc session.
<b>io 7 </b>	0(default) - If NDM is detected, a device-io operation is processed by NDM. Else, a device-io operation is processed by direct(windows API). 1 – a device-io operation is processed by direct(windows API). 2 - a device-io operation is processed by NDM.

## History

- 2022.11.25 – the first release. V1.0
- 2023.09.21 – release. Version 4.0. For mangement, V2x and 3.x is not used. Supporting lpu238 device.