esp_idf

User Manual

Reference:	TLT-XXX-MAN-esp_idf-esp_idf
Version:	1.1.1
Revision:	XXX

Confidentiality & Intellectual Property

Table of Contents

Chapter 1. File Documentation

1.1. com_espressif_esp_idf_esp_ota_ops.h File Reference

```
#include <stdint.h>
```

1.1.1. Functions

- int32_t Java_com_espressif_esp_1idf_esp_1ota_1ops_esp_1ota_1begin (int32_t partition, int32_t image_size)
- int32_t Java_com_espressif_esp_1idf_esp_1ota_1ops_esp_1ota_1end ( )
- int32_t Java_com_espressif_esp_1idf_esp_1ota_1ops_esp_1ota_1get_1boot_1partition (void)
- int32_t Java_com_espressif_esp_1idf_esp_1ota_1ops_esp_1ota_1get_1next_1update_1partition (int32_t start_from)
- int32_t Java_com_espressif_esp_1idf_esp_1ota_1ops_esp_1ota_1get_1running_1partition (void)
- int32_t Java_com_espressif_esp_1idf_esp_1ota_1ops_esp_1ota_1set_1boot_1partition (int32_t partition)
- int32_t Java_com_espressif_esp_1idf_esp_1ota_1ops_esp_1ota_1write (uint8_t * data, int32_t offset, int32_t size)
- int32_t Java_com_espressif_esp_1idf_esp_1ota_1ops_esp_1ota_1handle_1get (void)

Detailed Description

esp-idf low level API

Author: . MicroEJ Developer Team

Version: . 1.1.1

Date: . 13 November 2020

Definition in file /home/is2t/workspace/P0065_ESP32-Lib-c/com.espressif.esp_idf-c/target~/ccomponentWorking/bsp/espressif/inc/com_espressif_esp_idf_esp_ota_ops.h

1.2. com_espressif_esp_idf_esp_system.h File Reference

```
#include <sni.h>
```

1.2.1. Functions

- void Java_com_espressif_esp_1idf_esp_1system_esp_1restart (void)

Detailed Description

esp-idf low level API

Author: . MicroEJ Developer Team

Version: . 1.1.1

Date: . 13 November 2020

Definition in file /home/is2t/workspace/P0065_ESP32-Lib-c/com.espressif.esp_idf-c/target~/ccomponentWorking/bsp/espressif/inc/com_espressif_esp_idf_esp_system.h

1.3. com_espressif_esp_idf_nvs.h File Reference

```
#include <sni.h>
```

1.3.1. Functions

- void Java_com_espressif_esp_1idf_nvs_nvs_1close (jint arg0)
- void Java_com_espressif_esp_1idf_nvs_nvs_1commit (jint arg0)
- void Java_com_espressif_esp_1idf_nvs_nvs_1erase_1all (jint arg0)
- void Java_com_espressif_esp_1idf_nvs_nvs_1erase_1key (jint arg0, jbyte * arg1)
- jint Java_com_espressif_esp_1idf_nvs_nvs_1get_1blob (jint arg0, jbyte * arg1, jbyte * arg2, jint arg3)
- jshort Java_com_espressif_esp_1idf_nvs_nvs_1get_1i16 (jint arg0, jbyte * arg1)
- jint Java_com_espressif_esp_1idf_nvs_nvs_1get_1i32 (jint arg0, jbyte * arg1)
- jlong Java_com_espressif_esp_1idf_nvs_nvs_1get_1i64 (jint arg0, jbyte * arg1)

- jbyte Java_com_espressif_esp_1idf_nvs_nvs_1get_1i8 (jint arg0, jbyte * arg1)
- jint Java_com_espressif_esp_1idf_nvs_nvs_1get_1str (jint arg0, jbyte * arg1, jbyte * arg2, jint arg3)
- jshort Java_com_espressif_esp_1idf_nvs_nvs_1get_1u16 (jint arg0, jbyte * arg1)
- jint Java_com_espressif_esp_1idf_nvs_nvs_1get_1u32 (jint arg0, jbyte * arg1)
- jlong Java_com_espressif_esp_1idf_nvs_nvs_1get_1u64 (jint arg0, jbyte * arg1)
- jbyte Java_com_espressif_esp_1idf_nvs_nvs_1get_1u8 (jint arg0, jbyte * arg1)
- jlong Java_com_espressif_esp_1idf_nvs_nvs_1get_1used_1entry_1count (jint arg0)
- jint Java_com_espressif_esp_1idf_nvs_nvs_1open (jbyte * arg0, jint arg1)
- jint Java_com_espressif_esp_1idf_nvs_nvs_1open_1from_1partition (jbyte * arg0, jbyte * arg1, jint arg2)
- void Java_com_espressif_esp_1idf_nvs_nvs_1set_1blob (jint arg0, jbyte * arg1, jbyte * arg2, jint arg3)
- void Java_com_espressif_esp_1idf_nvs_nvs_1set_1i16 (jint arg0, jbyte * arg1, jshort arg2)
- void Java_com_espressif_esp_1idf_nvs_nvs_1set_1i32 (jint arg0, jbyte * arg1, jint arg2)
- void Java_com_espressif_esp_1idf_nvs_nvs_1set_1i64 (jint arg0, jbyte * arg1, jlong arg2)
- void Java_com_espressif_esp_1idf_nvs_nvs_1set_1i8 (jint arg0, jbyte * arg1, jbyte arg2)
- void Java_com_espressif_esp_1idf_nvs_nvs_1set_1str (jint arg0, jbyte * arg1, jbyte * arg2)
- void Java_com_espressif_esp_1idf_nvs_nvs_1set_1u16 (jint arg0, jbyte * arg1, jshort arg2)
- void Java_com_espressif_esp_1idf_nvs_nvs_1set_1u32 (jint arg0, jbyte * arg1, jint arg2)
- void Java_com_espressif_esp_1idf_nvs_nvs_1set_1u64 (jint arg0, jbyte * arg1, jlong arg2)
- void Java_com_espressif_esp_1idf_nvs_nvs_1set_1u8 (jint arg0, jbyte * arg1, jbyte arg2)

Detailed Description

esp-idf low level API

Author: . MicroEJ Developer Team

Version: . 1.1.1

Date: . 13 November 2020

Definition in file /home/is2t/workspace/P0065_ESP32-Lib-c/com.espressif.esp_idf-c/target~/ccomponentWorking/bsp/espressif/inc/com_espressif_esp_idf_nvs.h

1.4. com_espressif_esp_idf_esp_ota_ops.c File Reference

```
#include <stdint.h>
```

```
#include <stddef.h>
```

```
#include "com_espressif_esp_idf_esp_ota_ops.h"
```

```
#include "esp_err.h"
```

```
#include "esp_ota_ops.h"
```

1.4.1. Variables

- static esp_ota_handle_t handle

1.4.2. Functions

- int32_t Java_com_espressif_esp_1idf_esp_1ota_1ops_esp_1ota_1begin (int32_t partition, int32_t image_size)
- int32_t Java_com_espressif_esp_1idf_esp_1ota_1ops_esp_1ota_1end ( )
- int32_t Java_com_espressif_esp_1idf_esp_1ota_1ops_esp_1ota_1get_1boot_1partition (void)
- int32_t Java_com_espressif_esp_1idf_esp_1ota_1ops_esp_1ota_1get_1next_1update_1partition (int32_t start_from)
- int32_t Java_com_espressif_esp_1idf_esp_1ota_1ops_esp_1ota_1get_1running_1partition (void)
- int32_t Java_com_espressif_esp_1idf_esp_1ota_1ops_esp_1ota_1set_1boot_1partition (int32_t partition)
- int32_t Java_com_espressif_esp_1idf_esp_1ota_1ops_esp_1ota_1write (uint8_t * data, int32_t offset, int32_t size)

Detailed Description

esp-idf low level API implementation

Author: . MicroEJ Developer Team

Version: . 1.1.1

Date: . 13 November 2020

Definition in file /home/is2t/workspace/P0065_ESP32-Lib-c/com.espressif.esp_idf-c/target~/ccomponentWorking/bsp/espressif/src/com_espressif_esp_idf_esp_ota_ops.c

1.5. com_espressif_esp_idf_esp_system.c File Reference

```
#include "com_espressif_esp_idf_esp_system.h"
```

```
#include "esp_system.h"
```

1.5.1. Functions

- void Java_com_espressif_esp_1idf_esp_1system_esp_1restart (void)

Detailed Description

esp-idf low level API implementation

Author: . MicroEJ Developer Team

Version: . 1.1.1

Date: . 13 November 2020

Definition in file /home/is2t/workspace/P0065_ESP32-Lib-c/com.espressif.esp_idf-c/target~/ccomponentWorking/bsp/espressif/src/com_espressif_esp_idf_esp_system.c

1.6. com_espressif_esp_idf_nvs.c File Reference

```
#include "com_espressif_esp_idf_nvs.h"
```

```
#include "nvs.h"
```

1.6.1. Functions

- void Java_com_espressif_esp_1idf_nvs_nvs_1close (jint handle)

- void Java_com_espressif_esp_1idf_nvs_nvs_1commit (jint handle)
- void Java_com_espressif_esp_1idf_nvs_nvs_1erase_1all (jint handle)
- void Java_com_espressif_esp_1idf_nvs_nvs_1erase_1key (jint handle, jbyte * key)
- jint Java_com_espressif_esp_1idf_nvs_nvs_1get_1blob (jint handle, jbyte * key, jbyte * blob, jint length)
- jshort Java_com_espressif_esp_1idf_nvs_nvs_1get_1i16 (jint handle, jbyte * key)
- jint Java_com_espressif_esp_1idf_nvs_nvs_1get_1i32 (jint handle, jbyte * key)
- jlong Java_com_espressif_esp_1idf_nvs_nvs_1get_1i64 (jint handle, jbyte * key)
- jbyte Java_com_espressif_esp_1idf_nvs_nvs_1get_1i8 (jint handle, jbyte * key)
- jint Java_com_espressif_esp_1idf_nvs_nvs_1get_1str (jint handle, jbyte * key, jbyte * str, jint length)
- jshort Java_com_espressif_esp_1idf_nvs_nvs_1get_1u16 (jint handle, jbyte * key)
- jint Java_com_espressif_esp_1idf_nvs_nvs_1get_1u32 (jint handle, jbyte * key)
- jlong Java_com_espressif_esp_1idf_nvs_nvs_1get_1u64 (jint handle, jbyte * key)
- jbyte Java_com_espressif_esp_1idf_nvs_nvs_1get_1u8 (jint handle, jbyte * key)
- jlong Java_com_espressif_esp_1idf_nvs_nvs_1get_1used_1entry_1count (jint handle)
- jint Java_com_espressif_esp_1idf_nvs_nvs_1open (jbyte * name, jint open_mode)
- jint Java_com_espressif_esp_1idf_nvs_nvs_1open_1from_1partition (jbyte * part_name, jbyte * name, jint open_mode)
- void Java_com_espressif_esp_1idf_nvs_nvs_1set_1blob (jint handle, jbyte * key, jbyte * blob, jint length)
- void Java_com_espressif_esp_1idf_nvs_nvs_1set_1i16 (jint handle, jbyte * key, jshort value)
- void Java_com_espressif_esp_1idf_nvs_nvs_1set_1i32 (jint handle, jbyte * key, jint value)
- void Java_com_espressif_esp_1idf_nvs_nvs_1set_1i64 (jint handle, jbyte * key, jlong value)
- void Java_com_espressif_esp_1idf_nvs_nvs_1set_1i8 (jint handle, jbyte * key, jbyte value)
- void Java_com_espressif_esp_1idf_nvs_nvs_1set_1str (jint handle, jbyte * key, jbyte * value)
- void Java_com_espressif_esp_1idf_nvs_nvs_1set_1u16 (jint handle, jbyte * key, jshort value)
- void Java_com_espressif_esp_1idf_nvs_nvs_1set_1u32 (jint handle, jbyte * key, jint value)
- void Java_com_espressif_esp_1idf_nvs_nvs_1set_1u64 (jint handle, jbyte * key, jlong value)
- void Java_com_espressif_esp_1idf_nvs_nvs_1set_1u8 (jint handle, jbyte * key, jbyte value)

Detailed Description

esp-idf low level API implementation

Author: . MicroEJ Developer Team

Version: . 1.1.1

Date: . 13 November 2020

Definition in file /home/is2t/workspace/P0065_ESP32-Lib-c/com.espressif.esp_idf-c/target~/ccomponentWorking/bsp/espressif/src/com_espressif_esp_idf_nvs.c