

HEKR ESP8266 SDK 接口说明

v1.0.5 by xuefeng.zeng@hekr.me 2015/8/18 17:40:40

1-1 HekrConfig (Wi-Fi一键配置)

```
void start_hekr_config(hekr_config_event_cb_t event_cb, size_t timeout)
```

参数

- `event_cb` HekrConfig事件回调函数
- `timeout` 配置超时时间 (单位: 毫秒)

返回值

- 无

头文件

- `#include <module_wifi.h>`

参数类型定义

```
typedef enum
{
    EVENT_CONNECT_WIFI_FAIL = 0,    //配置失败, 未能连接上Wi-Fi
    EVENT_HEKR_CONFIG_TIMEOUT,      //配置超时
    EVENT_HEKR_CONFIG_FINISH        //配置成功, 将会自动去连接服务器
}config_event_t;

typedef void( *hekr_config_event_cb_t)(config_event_t event);
```

说明

- 上电的时候, 如果FLASH中没有Wi-Fi配置信息会自动进入配置模式。

1-2 取消Wi-Fi 一键配置

```
void stop_hekr_config(void)
```

参数

- 无

返回值

- 无

2-1 连接服务器

```
void connect_server(cloud_conn_event_cb_t cb)
```

参数

- `cb` 与服务器连接事件回调函数

返回值

- 无

头文件

- `#include <module_wifi.h>`

参数类型定义

```
typedef enum {  
    WIFI_DISCONNECTED = 0,      //Wi-Fi未连接  
    NO_ACCESSKEY,              //设备没有accesskey  
    DNS_ERROR,                 //无法获取服务器IP  
    DEVICE_LOGIN_ERROR,        //设备登入服务器失败  
    DEVICE_LOGIN_SUCCESS,      //设备登入服务器成功  
    DISCONNECTED_FROM_CLOUD    //设备与服务器连接断开  
}cloud_conn_event_t;  
  
typedef void(*cloud_conn_event_cb_t)(cloud_conn_event_t event);
```

2-2 断开与服务器的连接

```
void disconnected_from_server(void)
```

参数

- 无

返回值

- 无

2-3 向服务器发送数据

```
uint8_t send_data_to_cloud(void *src ,size_t size)
```

参数

- `src` 待发送的数据
- `size` 数据的大小 （单位：字节）

返回值

- 1 成功
- 0 失败

2-4 注册收到服务器数据的回调函数

```
void register_receive_server_data_callback(recicve_server_data_cb_t cb)
```

参数

- `cb` 收到服务器数据的回调函数

返回值

- 无

头文件

- `#include <conn_cloud.h>`

参数类型定义

```
typedef void(*recicve_server_data_cb_t)(void *data, size_t size);
```

3-1 注册串口收到数据的回调函数

```
void register_uart_data_received_callback(uart_data_received_cb_t *cb)
```

参数

- `cb` 串口回调函数

返回值

- 无

头文件

- `#include <uart.h>`

相关定义

```
typedef void(uart_data_received_cb_t)(uint8_t data);
```

4-1获取SDK版本号

```
char *get_hekr_sdk_version(void)
```

参数

- 无

返回值

- 类型 字符串
- 值 版本号

头文件

- `#include <device_info.h>`

4-2 系统日志

```
void system_log_set(log_port_t port)
```

参数

- `port` 设置输出端口，可选参数`PORT_UART0`、`PORT_UART1`、`PORT_NULL`

返回值

- 无

参数类型定义

```
typedef enum
{
    PORT_UART0 = 0, //通过uart0输出
    PORT_UART1,     //通过uart1输出
    PORT_NULL       //不输出
}log_port_t;
```

头文件

- `#include <log.h>`

说明

- 日志输出会占用资源，生成固件建议关闭

4-1 用户函数执行入口

```
void hekr_main(void)
```

说明

- 该函数在设备启动过程执行，初始化的函数放在此处执行。
- 执行时间长的任务不要放在此处执行

示例

```
void hekr_main(void)
{
    /*通过串口GPIO2输出信息*/
    /*波特率9600 8位数据位 1位停止位*/
    uart_init(BIT_RATE_9600, BIT_RATE_9600);
    os_printf("\n\nsystem run !! \n\n");
    os_printf("hekr sdk ver:%s\n", get_hekr_sdk_version());
}
```

参数

- 无

返回值

- 无

头文件

- 无 该函数用户自己定义

备注