

EXTI库

EXTI就是常说的外部中断，EXTI库就是关于单片机的外部中断的操作库。在使用本库之前先到ecbm_core.h里使能。双击打开ecbm_core.h文件，然后进入图形化配置界面，使能EXTI库。

| Option | Value |
|-----------|-------------------------------------|
| 单片机系统设置 | |
| 单片机外设库的选用 | |
| POWER库 | <input type="checkbox"/> |
| UART库 | <input type="checkbox"/> |
| EXTI库 | <input checked="" type="checkbox"/> |
| TIMER库 | <input type="checkbox"/> |
| EEPROM库 | <input type="checkbox"/> |

API

exti_init

函数原型：void exti_init(void);

描述

外部中断初始化函数。

输入

无

输出

无

返回值

无

参数配置

本函数初始化的参数都是由图形化配置界面来设置，双击打开exti.h文件，进入图形化配置界面。

| Option | Value |
|---|-------------------------------------|
| <input checked="" type="checkbox"/> 外部中断0 | <input checked="" type="checkbox"/> |
| 中断模式选择 | 只下降沿 |
| 初始化时就打开中断？ | <input checked="" type="checkbox"/> |
| <input checked="" type="checkbox"/> 外部中断1 | <input checked="" type="checkbox"/> |
| 中断模式选择 | 上升沿/下降沿 |
| 初始化时就打开中断？ | <input checked="" type="checkbox"/> |
| <input checked="" type="checkbox"/> 外部中断2 | <input checked="" type="checkbox"/> |
| 初始化时就打开中断？ | <input checked="" type="checkbox"/> |
| <input checked="" type="checkbox"/> 外部中断3 | <input checked="" type="checkbox"/> |
| 初始化时就打开中断？ | <input checked="" type="checkbox"/> |
| <input checked="" type="checkbox"/> 外部中断4 | <input checked="" type="checkbox"/> |
| 初始化时就打开中断？ | <input checked="" type="checkbox"/> |

设置说明如下：

外部中断0

- **外部中断0**：勾选这个才会编译外部中断0的初始化代码。
- **中断模式选择**：有【上升沿/下降沿】和【只下降沿】两个可以选择。
- **初始化时就打开中断**：勾选后外部中断0的中断使能会在初始化函数里打开。

外部中断1

- **外部中断1**：勾选这个才会编译外部中断1的初始化代码。
- **中断模式选择**：有【上升沿/下降沿】和【只下降沿】两个可以选择。
- **初始化时就打开中断**：勾选后外部中断1的中断使能会在初始化函数里打开。

外部中断2

- **外部中断2**：勾选这个才会编译外部中断2的初始化代码。
- **初始化时就打开中断**：勾选后外部中断2的中断使能会在初始化函数里打开。

外部中断3

- **外部中断3**：勾选这个才会编译外部中断3的初始化代码。
- **初始化时就打开中断**：勾选后外部中断3的中断使能会在初始化函数里打开。

外部中断4

- **外部中断4**：勾选这个才会编译外部中断4的初始化代码。
- **初始化时就打开中断**：勾选后外部中断4的中断使能会在初始化函数里打开。

小科普

- 这里的外部中断是和标准51类似工作原理的外部中断。STC8的最新型号里还有其他原理实现的外部中断，为防止混淆ECBM库会把那个新的中断称之为**IO中断**。
- STC8的中断模式是【上升沿/下降沿】和【只下降沿】。标准51是【低电平】和【下降沿】。

调用例程

```
#include "ecbm_core.h"//加载库函数的头文件。
void main(){//main函数，必须的。
    system_init();//系统初始化函数。
    exti_init();//初始化外部中断脚。
    while(1){
    }
}
```

注意事项

1. 只有外部中断0和外部中断1可以选择中断模式。
2. 【上升沿/下降沿】模式下，无论是上升沿还是下降沿都会触发中断，此时需要读一遍IO口的电平值来确认是什么边沿触发的中断。

exti_start

函数原型：void exti_start(u8 id);

描述

打开中断函数。

输入

- id：打开的外部中断编号，可输入范围是0~4。

输出

无

返回值

无

调用例程

```
if(control_mode==KEY_MODE){//如果操作模式切换成按键控制，
    exti_start(0); //打开按键所在的外部中断0。
}
```

注意事项

1. 如果在图形化配置界面选择了【初始化时就打开中断】，那么本函数将会在exti_init函数执行的时候自动调用，不需要额外执行了。反过来，假设没有选择【初始化时就打开中断】，那么一定得执行本函数，外部中断才能正常工作。

exti_stop

函数原型：void exti_stop(u8 id);

描述

关闭中断函数。

输入

- id: 关闭的外部中断编号，可输入范围是0~4。

输出

无

返回值

无

调用例程

```
if(control_mode==UART_MODE){//如果操作模式切换成串口控制，
    exti_stop(0); //关闭按键所在的外部中断0。
}
```

注意事项

无

exti_set_mode

函数原型: void exti_set_mode(u8 id,u8 mode);

描述

中断模式设置函数，受单片机框架原因只有外部中断0和外部中断1可用。

输入

- id: 设置的外部中断编号，0或者1。
- mode: 设置的中断模式。

输出

无

返回值

无

调用例程

工作的模式可以在图形化配置界面设置，也可以通过调用函数来修改，实现动态触发的效果。

参数的宏定义是：

- EXTI_MODE_UP_DOWN: 上升沿和下降沿都会触发中断。
- EXTI_MODE_DOWN: 只有下降沿才会触发中断。

```
exti_set_mode(0,EXTI_MODE_DOWN); //将外部中断0的中断模式设置成下降沿中断。
```

注意事项

1. 外部中断没有相应的标志显示是什么边沿触发的中断，在设置上升沿和下降沿模式之后，需要手动判断IO的电平来区分边沿的类型。比如触发中断后，IO是高电平，说明是上升沿触发。同理，触发

中断后，IO是低电平，说明是下降沿触发。

回调函数

在串口库那一章里有提到过回调函数的执行原理。目前为了统一管理中断，ECBM库已经定义好了中断处理函数，并开放了回调函数供大家使用。只要按着回调函数的名字定义一个一模一样的函数就行。

- exti0_it_callback
- exti1_it_callback
- exti2_it_callback
- exti3_it_callback
- exti4_it_callback

调用例程

```
#include "ecbm_core.h"//加载库函数的头文件。
void main(){//main函数，必须的。
    system_init();//系统初始化函数。
    exti_init();//初始化外部中断脚。
    while(1){
    }
}
void exti0_it_callback(void){//这是外部中断0的回调函数。
    LED=!LED; //当触发外部中断0中断时，取反LED的亮灭状态。
}
```

注意事项

1. 回调函数的名字一定得正确，因为中断里面会调用这些回调函数，如果名字不正确就调用不了。
2. 本函数不需要也不能在其他地方调用！只需要定义了即可。因为在外中断触发的时候，单片机的硬件会自动去调用的。

优化建议

本库内容比较简单，所以可优化的地方不多。基本保证没用到的功能不使能就行。