

算法接口说明

1 数据结构

```
typedef struct
{
    Uint32_t  steps;
    Uint32_t  cals;
    Uint32_t  dist;
    uint8_t   state;
}SportDataType;
steps -----步数
cals -----卡路里（单位：0.1 大卡）
dist -----里程（单位：0.1 米）
state-----保留
```

2 函数

(1)、void Sport_Init(void)

功能：初始化。

说明：该函数需要用户开机调用一次进行相关初始化工作。

(2)、void Set_Parameter(uint8_t height,uint8_t weight)

功能：设置用户信息（身高、体重）。

说明：该函数供用户使用设置用户信息，传入的参数 **height** 为身高，单位 **CM**；**weight** 为体重，单位 **KG**。

(3)、void Sport_Calculator(int32_t x,int32_t y,int32_t z)

功能：运动相关计算主函数。

说明：该函数需要定时调用（执行频率 **50HZ**，定时 **20MS** 调用一次）。传入的参数 **x,y,z**（32 位有符号整形）为用户在每次调用该函数前读取的加速度三轴

数据，单位 MG。

(4)、void Read_SportData(SportDataType *data)

功能：运动数据读取。

说明：该函数用于读取运动数据（步数、卡路里、里程），赋值给 SportDataType 型结构体，读取的步数、卡路里、里程数据是自用户上一次读取后开始计算的数据，如果是第一次读取则读取的是从开机运行开始计算的数据。

(5)、uint8_t Set_SportData(uint32_t step, uint32_t cal, uint32_t dis)

功能：设置运动数据。

说明：该函数用于设置运动数据（步数、卡路里、里程），与 SportDataType 结构体中的数据单位一致。成功返回 1，失败返回 0。

(6)、uint32_t Get_Version(void)

功能：获取版本信息。

说明：该函数用于获取版本的信息，例如 11，即当前版本为 1.1。

(7)、uint8_t Set_Step_Parameter(uint8_t pattern)

功能：设置计步模式。

说明：该函数用于设置计步模式，pattern 值为 1 时，当前为日常计步模式，pattern 值为 2 时，当前为跑步、健走模式。跑步、室内跑步、健走、室内健走设置为 2，日常计步、骑行、游泳等其他情况设置为 1。

(8)、uint8_t Set_Step_Stride(uint32_t stride)

功能：设置步幅。

说明：该函数用于设置高步频下的步幅，设置成功，返回 1，设置失败，返回 0，表示设置的步幅为异常值。恢复初始状态，则设置步幅为 0。

3 说明

算法共有 counter.lib 和 counter.h 两个文件，其中 h 文件里已经对供外部使用的数据结构、函数和状态定义做了声明，用户只需要在使用该数据的文件里包含 counter.h 即可，即 #include "counter.h"。