/\*

\获得板卡状态\*/

uint8\_t get\_board\_state(uint8\_t count);

返回值： 0 空闲状态 1 工作状态 -1 没有该板卡（若只有一块板卡 输入2）

第一块板卡状态 输入 1

第二块板卡状态 输入 2

第三块板卡状态 输入 3

以此类推

/\*

\获得板卡数量

\*/

uint8\_t get\_board\_count(void);

调用这个函数 可以获得板卡数量

返回值：

返回0 表示失败 没有板卡

>0 返回的板卡数量

/\*

\加载dll库 -- 打开板卡 复位板卡 申请资源

\*/

int Load\_library(void);

必须先调用这个函数，这个函数申请所需要的空间、资源等

返回值：

返回0 表示失败

>0 返回的板卡数量

/\*

\开始存数据

\*/

void Start\_receiving(uint8\_t index);

开启板卡存储

参数：启动哪块板卡，是按位来计算的，比如

传入 1 0x01 -- 表示开启第一块板卡

传入 2 0x10 -- 表示开启第二块板卡

传入 3 0x11 -- 表示开启第一块和第二块板卡

传入 15 0x1111-- 表示开启第一块到第四块板卡

/\*

\停止存数据

\*/

void Stop\_receiving(uint8\_t index);

关闭板卡 存储

参数：关闭哪块板卡存储，是按位来计算的，比如

传入 1 0x01 -- 表示关闭第一块板卡

传入 2 0x10 -- 表示关闭第二块板卡

传入 3 0x11 -- 表示关闭第一块和第二块板卡

传入 15 0x1111-- 表示关闭第一块到第四块板卡

/\*

\卸载dll库 关闭板卡 释放资源等

\*/

int Unload\_library(void);

卸载库 时候调用 释放资源