**广州数控25I系统库接口函数说明**

作者：李志波

时间：2017.03

说明：

25i网络库接口版本2.0

gsk25inetfun.dll

gsk25inetfun.h

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

错误码定义, 大于等于 0x8000 的错误码是 PC 端产生错误码

#define GSK25I\_OK 0x0000 // 成功，没有错误

#define GSK25I\_ERR\_HINST 0x8000 //实例不存在

#define GSK25I\_ERR\_REQ 0x8001 //命令发送或接收出错

#define GSK25I\_ERR\_AXIS 0x8002 //轴数超出范围

#define GSK25I\_ERR\_TYPE 0x8003 //不存在的类型

**连接管理接口:**

/\*

功 能：创建GSK数据通信实例并远程通信连接

参 数：cncIPAddr : CNC 的IP 地址

　　 type : 通讯模式 0为UDP,1为TCP/IP

返回值：成功返回通信实例句柄，出错返回NULL

备注： 988TD支持UDP 和 TCP/IP 2种方式

25i 目前支持 TCP/IP 1种方式\*/

GSK25I\_API HINSGSKRM GSKRM\_CreateInstance(unsigned char cncIPAddr[4], int type);

/\*

功 能：查询实例的连接状态

参 数：hInst: CNC远程诊断通信实例句柄

返回值：连接状态，-1:无效实例； 0:未连接; 1:已经连接 \*/

GSK25I\_API int GSKRM\_GetConnectState(HINSGSKRM hInst);

/\*

功 能：远程断开连接并关闭GSK数据通信实例

参 数：hInst: GSK数据通信实例句柄

返回值：无\*/

GSK25I\_API int GSKRM\_CloseInstance(HINSGSKRM hInst);

/\*

功 能： 设置通信超时时间

参 数： hInst: GSK数据通信实例句柄

overtime: 通信超时时间，单位ms

返回值：成功返回0, 否则返回错误码\*/

GSK25I\_API int GSKRM\_SetOvertime(HINSGSKRM hInst, unsigned int overtime);

/\*

功 能： 获取CNC型号名称

参 数： hInst: GSK数据通信实例句柄

typeName: 用于存放获得的CNC型号名称，数组大小应大于等于32

如：“25i”、“988TD”

返回值：成功返回0, 否则返回错误码\*/

GSK25I\_API int GSKRM\_GetCncTypeName(HINSGSKRM hInst, char typeName[32]);

**业务处理接口**

/\*

功 能： 获取当前系统的版本信息

参 数： hInst: GSK数据通信实例句柄

返回值：成功返回struct VERS\_INFO指针, 否则返回NULL\*/

struct VERS\_INFO

{

char sysVersion[32];//系统版本

char armVersion[32];//应用版本

char dspVersion[32];//插补版本

char FPGAVersion[32];//位控版本

char plcfileName[32];//梯图文件名

char hardVersion[32];//硬件版本 保留

char softWareNumber[32];//软件序号 保留

char hardWareNumber[32];//硬件序号 保留

};

GSK25I\_API struct VERS\_INFO\* GSKRM\_GetVersionInfo(HINSGSKRM hInst);

/\*

功 能： 获取当前系统的运行信息

参 数： hInst: GSK数据通信实例句柄

返回值：成功返回struct RUNSTAT\_INFO指针, 否则返回NULL\*/

struct RUNSTAT\_INFO

{

char OpMode; //操作模式

char RunMode; //运行模式

char EmergFlag; //急停信号输入

char currentalarmflag; //当前是否处于报警状态，报警个数

unsigned short currentAlarmNum; //当前报警号

unsigned short PartCount; //加工零件数

unsigned char Gmode[NET\_GGROUP\_NUM]; // 当前G代码模态

unsigned char GmodeNext[NET\_GGROUP\_NUM];//下段G代码模态

unsigned char SvCurrent[NET\_AXIS\_NUM]; //进给轴负载

unsigned char SpdCurrent[2]; //主轴负载

char progfileName[32]; //当前文件

unsigned int run\_time; // 运行时间

unsigned int process\_time; // 加工时间

unsigned int CNC\_time; //系统时间

unsigned int BlockNumber; //行号

int run\_time\_cnt; // 累计运行时间

int process\_time\_cnt; // 累计加工时间

int powerup\_times\_all; // 累计总次数

int powerup\_times\_month; // 累计本月次数

unsigned char loginLevel; //当前登陆等级

unsigned char opt\_error;

unsigned char custom\_error;

char reserver[64]; //保留

};

GSK25I\_API struct RUNSTAT\_INFO\* GSKRM\_GetRunInfo(HINSGSKRM hInst);

/\*

功 能： 获取当前系统的各轴坐标、倍率、转速信息

参 数： hInst: GSK数据通信实例句柄

返回值：成功返回struct AXIS\_INFO指针, 否则返回NULL\*/

struct AXIS\_INFO

{

double Abs\_Coord[NET\_AXIS\_NUM];//绝对坐标

double Rel\_Coord[NET\_AXIS\_NUM];//相对坐标

double Mac\_Coord[NET\_AXIS\_NUM];//机床坐标

double Addi\_Coord[NET\_AXIS\_NUM];//剩余距离

double Frd;

int Spd;

int F;//进给

int S;//转速!

int J;//快速倍率

double ActiveAxisSpeed[NET\_AXIS\_NUM]; //各轴实际分速度

};

GSK25I\_API struct AXIS\_INFO\* GSKRM\_GetAxisInfo(HINSGSKRM hInst);

/\*

功 能： 获取当前系统的报警履历

参 数： hInst: GSK数据通信实例句柄

返回值：成功返回struct ALARM\_INFO指针,最多160组

否则返回NULL\*/

struct ALARM\_INFO

{

int index;//索引号

int axisNo; //报警轴号或从站号

char ErrorNoStr[8];//报警号

char ErrorTime[16];//报警时间

char ErrorMessage[64];//报警信息

};

GSK25I\_API struct ALARM\_INFO\* GSKRM\_GetAlarmInfo(HINSGSKRM hInst,int \*retcnt);

/\*

功 能： 获取当前系统的加工履历

参 数： hInst: GSK数据通信实例句柄

返回值：成功返回struct PROGRAM\_INFO指针,最多160组

否则返回NULL\*/

struct PROGRAM\_INFO

{

int index;//索引号

char programName[32];//加工程序名

char programTime[32];//加工时间(结束时间+加工耗时)

};

GSK25I\_API struct PROGRAM\_INFO\* GSKRM\_GetProgramInfo(HINSGSKRM hInst,int \*retcnt);

/\*

功 能： 获取当前系统的操作履历

参 数： hInst: GSK数据通信实例句柄

返回值：成功返回struct OPERATE\_INFO指针,最多160组

否则返回NULL\*/

struct OPERATE\_INFO

{

int index;//索引号

char optstr[5][32];

//0//操作类型

//1//操作号或操作位

//2//操作前数值

//3//操作后数值

//4//操作时间

};

GSK25I\_API struct OPERATE\_INFO\* GSKRM\_GetOperateInfo(HINSGSKRM hInst,int \*retcnt);

/\*

写入CNC数据接口

功 能： 设置PLC寄存器，单一类型地址，

参 数： hInst: GSK数据通信实例句柄

type: 地址类型

index: 地址个数 起点, 偏移地址

bit: 0-7 读取寄存器的数量 (从起点开始的连续多个数据)

pValue: 0/1 当前数据缓冲区指针

返回值：成功返回0, 否则返回错误码\*/

extern "C" GSK25I\_API int GSKRM\_SetPlcDatabit(HINSGSKRM hInst, int type, int index, int bit, unsigned char Value);

/\*

写入CNC数据接口

功 能： 设置PLC寄存器，单一类型地址，从地址起点开始的连续多个数据的写入

参 数： hInst: GSK数据通信实例句柄

type: 地址类型

count: 地址个数 起点

num: 读取寄存器的数量 (从起点开始的连续多个数据)

pValue: 当前数据缓冲区指针

返回值：成功返回0, 否则返回错误码\*/

GSK25I\_API int GSKRM\_SetPlcData(HINSGSKRM hInst, int type, int count, int num, unsigned char dValue[]);

/\*

写入CNC数据接口

功 能： 设置工件系

参 数： hInst: GSK数据通信实例句柄

char coordType; //工件系类型, 0 : EXT G54 ~ G59; 1 : G54.1

char coordNo; //工件系编号

short reserver; //保留

char axisNo[4]; //轴号0-8

double value[4]; //值

返回值：成功返回0, 否则返回错误码

\*/

GSK25I\_API int GSKRM\_SetCoordSync(HINSGSKRM hInst, char coordType, char coordNo, char axisNo[], double value[]);

/\*

写入CNC数据接口

功 能： 设置工件系

参 数： hInst: GSK数据通信实例句柄

int toolNo;//刀具号

int type;//刀具中每一列的编号具体看宏的定义

double val; //单位mm

int optFlag; //0 正常写入 1 测量

返回值：成功返回0, 否则返回错误码

\*/

GSK25I\_API int GSKRM\_SetOfstSync(HINSGSKRM hInst, int toolNo, int type, int optFlag, double value);

/\*

写入CNC数据接口

功 能： 设置螺补

参 数： hInst: GSK数据通信实例句柄

int pitchNo;//螺补号

int pitchCnt;//个数

int direction; //1:正向/0:反向

int pitchVal; //螺补值

返回值：成功返回0, 否则返回错误码

\*/

GSK25I\_API int GSKRM\_SetPitchInfo(HINSGSKRM hInst, int pitchNo, int pitchCnt, int direction, int pitchVal);

struct PITCH\_ARRAY{

unsigned short cnt; //小于等于8

short direction; //0 反向 1 正向

short num[8];

short val[8];

};

/\*

读CNC数据接口

功 能： 读取螺补

参 数： hInst: GSK数据通信实例句柄

int flag;//1:正向/0:反向

int cnt;//个数

int pparamNum; //螺补号

返回值：成功返回0, 否则返回NULL

\*/

GSK25I\_API struct PITCH\_ARRAY\* GSKRM\_GetPitchInfo(HINSGSKRM hInst, int flag, int cnt, int \*pparamNum);

/\*

读CNC宏变量接口

功 能： 读取宏变量值

参 数： hInst: GSK数据通信实例句柄

int macroCnt;//宏变量个数

int \*macroNo;//害变量号

int \*macroVal; //宏变量值

返回值：成功返回0, 否则返回错误号

\*/

GSK25I\_API int GSKRM\_GetMacroInfo(HINSGSKRM hInst, int macroCnt, int \*macroNo, double \*macroVal);

/\*

设置CNC宏变量

功 能： 设置宏变量

参 数： hInst: GSK数据通信实例句柄

int macroCnt;//宏变量个数

int \*macroNo;//害变量号

int \*macroVal; //宏变量值

返回值：成功返回0, 否则返回错误号

\*/

GSK25I\_API int GSKRM\_SetMacroInfo(HINSGSKRM hInst, int macroCnt, int \*macroNo, double \*macroVal);

/\*

设置CNC参数

功 能： 设置参数

参 数： hInst: GSK数据通信实例句柄

int number;//参数号

int valcnt;//个数

int \*pval; //需要修改的参数值

返回值：成功返回0, 否则返回错误号

\*/

GSK25I\_API int GSKRM\_SetParamOpt(HINSGSKRM hInst, int number, int valcnt, double \*pval);

/\*

读CNC参数接口

功 能： 读取参数,位参以字节的十进制取值

参 数： hInst: GSK数据通信实例句柄

int number;//参数号

int valcnt;//个数

int \*pval; //存放参数值

返回值：成功返回0, 否则返回错误号

\*/

GSK25I\_API int GSKRM\_GetParamInfo(HINSGSKRM hInst, int number, int valcnt, double \*pval);

/\*

上传G代码文件,文件名后缀必须是大写NC, 如: 55.NC

功 能： 上传文件

参 数： hInst: GSK数据通信实例句柄

char \*filePath;//文件路径及文件名

bool isDNC; 1: DNC传输 0: 上传文件

返回值：成功返回0, 否则返回错误号

\*/

GSK25I\_API int GSKRM\_SetPutFile(HINSGSKRM hInst, char \*filePath, bool isDNC);