GOOSE/SV

软件底层开发包接口 API 说明文档

版本：V2.0

1.1 encodeGOOSE 编码 GOOSE 报文函数

函数声明	unsigned char *encodeGOOSE(GOOSE *gooseInfo, int *length);
功能	编码 goose 报文
参数	GOOSE *gooseInfo：需要编码的 goose 相关结构体指针 int *length ：需要传入一个 int 型指针，用以记录编码的数组长度
返回值	Unsigned char* ：编码成功后会返回一个 Unsigned char 数组指针，存放的是 goose 相关报文（无符号数组）,解码失败则返回 NULL。

示例	GOOSE *deom_p = malloc(sizeof (GOOSE));
	...
	int length = 0;
	unsigned char *arr = encodeGOOSE(deom_p,&length);

1.2 decodeGOOSE 解码 GOOSE 报文函数

函数声明	GOOSE *decodeGOOSE(const unsigned char *msg,int length);
功能	解码 goose 报文
参数	const unsigned char *msg：goose 报文存放数组； int length：goose 报文数组长度需给出；
返回值	解码成功返回一个 GOOSE *结构体指针,解码失败会返回 NULL。
示例	unsigned char arr[] ={...}; int length = X; GOOSE *demo_p2 = decodeGOOSE(arr ,length);

1.3 encodeSV 编码 SV 报文函数

函数	unsigned char *encodeSV(const SV *svInfo, int *length)

声明	
功能	编码 SV 报文
参数	const SV *svInfo：需要编码的 SV 结构体 int *length ：需要传入一个 int 型指针，用以记录编码的数组长度；
返回值	Unsigned char* ：编码成功后会返回一个 Unsigned char 数组指针，存放的是 sv 相关报文（无符号数组）,解码失败则返回 NULL。

示例	SV *demo2 = newSV();
	...
	int len = 0;
	unsigned char *arr2 = encodeSV(demo2 , &len);

1.4 decodeSV 解码 SV 报文函数

函数声明	SV *decodeSV(const unsigned char *msg, int length);
功能	解码 SV 报文
参数	const unsigned char *msg：SV 报文存放数组； int length：SV 报文数组长度需给出；
返回值	解码成功返回一个 SV*结构体指针,解码失败会返回 NULL。
示例	unsigned char arr[] ={...}; int length = X; SV *demo = decodeSV(arr ,length);

1.5 newDataByValue 函数

函数声明	Data * newDataByValue(int type, char *value);
功能	创建 IS0/IEC9506-2 中的 data 参数（用于 goose 编码中的 allData 结构赋值）
参数	int type：需要创建的 Data 的数据类型（如布尔型、浮点型和整数等） char *value：给指定的数据类型 Data 赋值

返回值	创建成功会返回相应 Data 指针

示例	GOOSE *p2 = malloc(sizeof (GOOSE));
	p2->allData[0] = *newDataByValue(GOOSE_FLOATING_POINT , "0");
	p2->allData[1] = *newDataByValue(GOOSE_BIT_STRING, "0000000000000");

1.6 printGoose 函数

函数声明	void printGoose(GOOSE *goose);
功能	打印相关 GOOSE 结构体信息
参数	GOOSE *goose：需要打印的 GOOSE 结构体指针
返回值	无
示例	GOOSE *p2 = malloc(sizeof (GOOSE)); ... printGoose(demo_p2);

1.7 newSV 函数

函数声明	SV *newSV()
功能	创建 SV 相关结构体
参数	无
返回值	SV* ：创建成功会返回一个指向 SV 结构提的指针
示例	SV *demo = newSV();

1.8 printf_SV 函数

函数声明	void printf_SV(SV *var)
功能	打印相关 SV 结构体信息
参数	SV *var：需要打印的 SV 结构体指针
返回值	无
示例	SV *demo = newSV(); ... printf_SV(demo3);

1.9 print_UnsignedCharArr 函数

函数声明	Void print_UnsignedCharArr(unsigned char *arr, int len)
功能	可以用于打印编码成功后产生 unsigned char 数组
参数	unsigned char *arr：需要打印的数组 int len：相关数组长度需要给出
返回值	无

示例	unsigned char arr[] ={...};
	int len = X;
	print_UnsignedCharArr(arr ,len);

1.10 getAllPcapCardInfo 函数

函数声明	char* getAllPcapCardInfo();
功	获取当前设备所有数据网口设备信息

能	
参数	无
返回值	char* ： 内容为所有数据网口设备信息[deviceName , deviceDesc]
示例	char* deviceInfo = getAllPcapCardInfo(); printf("device:%s \n" , deviceInfo);

1.11 getOnePcapCardInfoByIndex 函数

函数声明	char* getOnePcapCardInfoByIndex(int index)
功能	查看指定下标网口信息
参数	无
返回值	char* ： 内容为指定下标的数据网口设备信息[deviceName , deviceDesc]
示例	char* device = getOnePcapCardInfoByIndex(8); printf("device:%s \n" , device);

1.12 open_CurCard_By_Index 函数

函数声明	void open_CurCard_By_Index(int index);
功能	根据下标选择相应网口用以发送报文
参数	int index：指定下标（该参数可以结合 getAllPcapCardInfo()函数使用），工具之前的所有网口信息（下标从 0 开始）
返回值	无
示例	char* deviceInfo = getAllPcapCardInfo(); printf("device:%s \n" , deviceInfo);

	open_CurCard_By_Index(8);

1.13 StartSndGOOSETask 函数

函数声明	int StartSndGOOSETask(unsigned char *arr , int length , int sendNum);
功能	发送 GOOSE 报文
参数	unsigned char *arr：存放 goose 报文的 unsigned char 数组指针； int length：存放 goose 报文的 unsigned char 数组长度； int sendNum：报文发送次数；
返回值	int ：如果发送不成功，则返回-1；否则 0

示例	unsigned char arr[] ={...};
	int len = X;
	int res = StartSndGOOSETask (arr , len , 5);
	printf("res : %d\n" , res);

1.14 StartSndGOOSETaskMultiply 函数

函数声明	int StartSndGOOSETaskMultiply(GOOSE *gooseInfo , int sendNum);
功能	发送 GOOSE 报文
参数	GOOSE *gooseInfo：需要发送的 goose 结构体； int sendNum：报文发送次数；
返回值	int ：如果发送不成功，则返回-1；否则 0

示例	GOOSE *p = malloc(sizeof (GOOSE));
	...
	int res = StartSndGOOSETaskMultiply (p, 5);
	printf("res : %d\n" , res);

1.15 StartSndSVTask 函数

函数声明	int StartSndSVTask(unsigned char *arr , int length , int sendNum);
功能	发送 SV 报文
参数	unsigned char *arr：存放 SV 报文的 unsigned char 数组指针； int length：存放 SV 报文的 unsigned char 数组长度； int sendNum：报文发送次数；
返回值	int ：如果发送不成功，则返回-1；否则 0

示例	unsigned char arr[] ={...};
	int len = X;
	int res = StartSndSVTask(arr , len , 5);
	printf("res : %d\n" , res);

1.16 StartSndSVTaskMultiply 函数

函数声明	int StartSndSVTaskMultiply(SV *svInfo , int sendNum);
功能	发送 SV 报文
参数	SV *svInfo：需要发送的 SV 结构体； int sendNum：报文发送次数；
返回值	int ：如果发送不成功，则返回-1；否则 0

示例	SV *demo = newSV();
	...
	int res = StartSndSVTaskMultiply (demo, 5);
	printf("res : %d\n" , res);