Standard Specification

DeviceNet PC 主站库接口规范

DeviceNet 主站卡

SS01000000 V0.92 Date:2014/12/10 标准规范手册

类别	内容
关键词	DeviceNet 主站卡
摘　要	PCI/USB 接口的 DeviceNet 接口卡能使 PC 成为 DeviceNet 网络中的一个主控制节点，或者实现具体功能的节点，实现 DeviceNet 协议的通信功能

广州致远电子股份有限公司

修订历史

版本	日期	原因
V0.90	2008/08/04	创建文档
V0.91	2009/12/05	修改数据结构，增加错误代码
V0.92	2014/12/10	文档标准化

目 录

1. 函数库说明.....3
1.1 设备操作接口.....3
1.1.1 ZDNMA_Open.....3
1.1.2 ZDNMA_ReadBoardInfo.....3
1.1.3 ZDNMA_Init.....3
1.1.4 ZDNMA_Online.....4
1.1.5 ZDNMA_Offline.....4
1.1.6 ZDNMA_StartScan.....4
1.1.7 ZDNMA_StopScan.....5
1.1.8 ZDNMA_AddSlave.....5
1.1.9 ZDNMA_RemoveSlave.....5
1.1.10 ZDNMA_Uninit.....6
1.1.11 ZDNMA_Close.....6
1.2 设备服务接口.....7
1.2.1 ZDNMA_EMRequest.....7
1.3 设备数据接口.....8
1.3.1 ZDNMA_ReadSlaveIO.....8
1.3.2 ZDNMA_WriteSlaveIO.....8
1.4 设备网络管理接口.....9
1.4.1 ZDNMA_GetSlaveStatus.....9
1.5 数据结构.....10
1.5.1 ZDNMA_BOARDINFO.....10
1.5.2 ZDNMA_INITCFG.....10
1.5.3 ZDNMA_SLAVECONFIG.....11
1.6 调用流程.....14
1.7 错误代码总览.....15
1.7.1 主站卡错误代码表.....15
1.7.2 DeviceNet 通用错误代码表.....15
1.7.3 主从站设备当前状态.....16
2. 免责声明.....19

1. 函数库说明

1.1 设备操作接口

1.1.1 ZDNMA_Open

描述

调用此函数打开设备。

```
DWORD __stdcall ZDNMA_Open(DWORD DevType, DWORD DevIndex, DWORD Reserved);
```

参数

DevType: 设备型号，参见表 1.1

DevIndex: 设备索引号，对于同种型号的主站设备来说，最靠近 CPU 的设备索引号为 0，依此递增

Reserved: 保留，应填 0

返回值

返回 0 表示成功，否则为错误码，见表 1.5。

表 1.1 设备型号

设备型号列表	
名称	型号值
PCI-5010-D	1
USBCAN-E-D	101

1.1.2 ZDNMA_ReadBoardInfo

描述

调用此函数获取主站设备相关信息。

```
DWORD __stdcall ZDNMA_ReadBoardInfo(DWORD DevType, DWORD DevIndex, ZDNMA_BOARDINFO* pInfo);
```

参数

DevType: 设备型号

DevIndex: 设备索引号，对于同种型号的主站设备来说，最靠近 CPU 的设备索引号为 0，依此递增

pInfo: 存储设备信息缓冲区

返回值

返回 0 表示成功，否则为错误码，见表 1.5。

1.1.3 ZDNMA_Init

描述

调用此函数初始化主站通道。

```
DWORD __stdcall ZDNMA_Init(DWORD DevType, DWORD DevIndex, DWORD ChIndex, ZDNMA_INITCFG *Config, HANDLE* pOutHandle);
```

参数

DevType: 设备型号

DevIndex: 设备索引号，对于同种型号的主站设备来说，最靠近 CPU 的设备索引号为 0，依此递增

ChIndex: CAN 通道号，0 对应第一个通道

Config: 初始化参数

pOutHandle: 存储返回通道句柄

返回值

返回 0 表示成功，否则为错误码，见表 1.5。

1.1.4 ZDNMA_Online

描述

调用此函数使主站上线。

```
DWORD __stdcall ZDNMA_Online(HANDLE hdChannel);
```

参数

hdChannel: 通道句柄

返回值

返回 0 表示成功，否则为错误码，见表 1.5。

1.1.5 ZDNMA_Offline

描述

调用此函数使主站下线。

```
DWORD __stdcall ZDNMA_Offline(HANDLE hdChannel);
```

参数

hdChannel: 通道句柄

返回值

返回 0 表示成功，否则为错误码，见表 1.5。

1.1.6 ZDNMA_StartScan

描述

调用此函数扫描并连接从站。

```
DWORD __stdcall ZDNMA_StartScan(HANDLE hdChannel);
```

参数

hdChannel: 通道句柄

返回值

返回 0 表示成功，否则为错误码，见表 1.5。

1.1.7 ZDNMA_StopScan

描述

调用此函数停止扫描。

```
DWORD __stdcall ZDNMA_StopScan(HANDLE hdChannel);
```

参数

hdChannel: 通道句柄

返回值

返回 0 表示成功，否则为错误码，见表 1.5。

1.1.8 ZDNMA_AddSlave

描述

调用此函数添加从站。

```
DWORD __stdcall ZDNMA_AddSlave(HANDLE hdChannel, ZDNMA_SLAVECONFIG *Config);
```

参数

hdChannel: 通道句柄

Config: 从站配置参数

返回值

返回 0 表示成功，否则为错误码，见表 1.5。

1.1.9 ZDNMA_RemoveSlave

描述

调用此函数删除从站。

```
DWORD __stdcall ZDNMA_RemoveSlave(HANDLE hdChannel, DWORD dwSlaveID);
```

参数

hdChannel: 通道句柄

dwSlaveID: 从站地址，取值范围 0~63 或-1，指定-1 时尝试删除所有从站。

返回值

返回 0 表示成功，否则为错误码，见表 1.5。

1.1.10 ZDNMA_SaveCfg

描述

调用此函数保存当前配置，改变主站 MAC ID 和波特率，添加从站及删除从站后调用。

```
DWORD __stdcall ZDNMA_SaveCfg(HANDLE hdChannel);
```

参数

hdChannel: 通道句柄

返回值

返回 0 表示成功，否则为错误码，见表 1.5。

1.1.11 ZDNMA_Uninit

描述

调用此函数关闭通道。

```
DWORD __stdcall ZDNMA_Uninit(HANDLE hdChannel);
```

参数

hdChannel: 通道句柄

返回值

返回 0 表示成功，否则为错误码，见表 1.5。

1.1.12 ZDNMA_Close

描述

调用此函数关闭设备。

```
DWORD __stdcall ZDNMA_Close(DWORD DevType, DWORD DevIndex);
```

参数

DevType: 设备型号

DevIndex: 设备索引号，对于同种型号的主站设备来说，最靠近 CPU 的设备索引号为 0，依此递增

返回值

返回 0 表示成功，否则为错误码，见表 1.5。

1.2 设备服务接口

1.2.1 ZDNMA_EMRequest

描述

调用此函数向从站发送显示请求信息。

```
DWORD __stdcall ZDNMA_EMRequest(HANDLE hdChannel, DWORD dwSlaveID, DWORDdwService,DWORD dwClassID, DWORD dwInstanceID, BYTE *pServiceData, DWORD dwSize, BYTE *pReceiveData, DWORD *dwRecSize, DWORDdwWaitTime);
```

参数

hdChannel: 通道句柄

dwSlaveID: 从站地址，取值范围 0~63

dwService: DeivceNet 服务代码

dwClassID: 分类 ID

dwInstanceID: 分类的实例 ID

pServiceData: 指向服务所需的参数缓冲区的指针，参考 DeviceNet 规范中特定类或实例相关的显式请求信息的服务数据区

dwSize: 缓冲区 pServiceData 的长度

pReceiveData: 指向接收数据缓冲区的指针，参考 DeviceNet 规范中特定类或实例相关的显式应答信息的服务数据区

dwRecSize: 接收缓冲区的长度，函数返回时表示接收数据缓冲区中的有效数据长度

dwWaitTime: 等待超时时间，取值范围 100~30000，单位毫秒

返回值

返回 0 表示成功，否则为错误码，见表 1.5 和表 1.6。

1.3 设备数据接口

1.3.1 ZDNMA_ReadSlaveIO

描述

调用此函数获取指定从站的 I/O 输入数据。

```
DWORD __stdcall ZDNMA_ReadSlaveIO(HANDLE hdChannel, DWORD dwSlaveID, DWORD dwIOArea, BYTE* pBuffer,DWORD* dwBufSize,DWORD dwWaitTime);
```

参数

hdChannel: 通道句柄

dwSlaveID: 从站地址，取值范围 0~63

dwIOArea: 指定设备中被读取的 I/O 数据域，0=strobe,1=poll,2=cos,3=cyclic

pBuffer: 指向接收数据缓冲区的指针

dwBufSize: 接收缓冲区的长度，函数返回时表示接收数据缓冲区中的有效数据长度

dwWaitTime: 等待超时时间，取值范围 0~30000，单位毫秒

返回值

返回 0 表示成功，否则为错误码，见表 1.5 和表 1.6。

1.3.2 ZDNMA_WriteSlaveIO

描述

调用此函数向指定从站输出 I/O 数据。

```
DWORD __stdcall ZDNMA_WriteSlaveIO(HANDLE hdChannel, DWORD dwSlaveID, DWORD dwIOArea, BYTE* pBuffer,DWORD dwBufSize,DWORD dwWaitTime);
```

参数

hdChannel: 通道句柄

dwSlaveID: 从站地址，取值范围 0~63

dwIOArea: 指定设备中被写入的 I/O 数据域，0=strobe,1=poll,2=cos,3=cyclic

pBuffer: 指向发送数据缓冲区的指针

dwBufSize: 发送缓冲区的长度

dwWaitTime: 等待超时时间，取值范围 0~30000，单位毫秒

返回值

返回 0 表示成功，否则为错误码，见表 1.5 和表 1.6。

1.4 设备网络管理接口

1.4.1 ZDNMA_GetSlaveStatus

描述

调用此函数获取从站工作状态。

```
DWORD __stdcall ZDNMA_GetSlaveStatus(HANDLE hdChannel, DWORD dwSlaveID);
```

参数

hdChannel: 通道句柄

dwSlaveID: 从站地址，取值范围 0~63

返回值

返回从站工作状态，见表 1.7。

1.5 数据结构

1.5.1 ZDNMA_BOARDINFO

描述

此数据结构存储主站设备相关信息，调用函数：ZDNMA_ReadBoardInfo。

```
typedef struct _tagZDNMA_BoardInfo
{
    char szProdName[32];
    DWORD dwVendorID;
    DWORD dwDevType;
    DWORD dwMajorRev;
    DWORD dwMinorRev;
    char szSerialNum[16];
    DWORD dwReserved[2];
} ZDNMA_BOARDINFO, *LPZDNMA_BOARDINFO;
```

表 1.2ZDNMA_BOARDINFO 结构体参数

名称	范围	描述
szProdName	32 字节	产品名称
dwVendorID	1-65535	厂商号
dwDevType	1-65535	主设备类型
dwMajorRev	1-65535	主版本号
dwMinorRev	1-65535	次版本号
szSerialNum[16]		设备序列号
dwReserved[2]		

1.5.2 ZDNMA_INITCFG

描述

此数据结构存储主站设备初始化参数，调用函数：ZDNMA_Init。

```
typedef struct _tagZDNMA_InitCfg
{
    DWORD dwMacId;
    DWORD dwBaudRate;
    DWORD dwInterScanDelay;     //主站内扫描延时时间（默认 10ms）
    DWORD dwBackgroundPollRate;//后台轮询扫描速率（默认 1），
    DWORD dwGlobalEPR;          //默认为 75
```

```
    DWORD dwTxRetrycount;        //发送重试次数，默认为 1
    DWORD dwADREnable;           //使能设备自动替换（当前为保留值）
    DWORD dwAutoScanEnable;      //使能自动扫描（当前为保留值）
    DWORD dwAutoScanFixedMappingSize; //自动扫描下固定映射大小
    DWORD dwReserved[4];
} ZDNMA_INITCFG, *LPZDNMA_INITCFG;
```

表 1.31.5.2 ZDNMA_INITCFG 结构体参数

名称	范围	描述
dwMacId	0-63	主站地址
dwBaudrate	2 1 0	CAN 波特率，比如 2 表示 500Kbps，1 表示 250Kbps，0 表示 125Kbps
dwInterScanDelay	1-65535	主站内扫描延时时间（默认 10ms）
dwBackgroundPollRate	1-65535	后台轮询扫描速率（默认 1）从站使能后台轮询扫描时有效，为全局扫描的倍数。 如果从站设置本参数，当全局速率扫描 N 次之后，扫描该从站一次。
dwGlobalEPR	1-65535	默认为 75
dwTxRetrycount	1-10	发送重试次数，默认为 1
dwADREnable	0-1	使能设备自动替换（当前为保留值）
dwAutoScanEnable	0-1	使能自动扫描（当前为保留值）
dwAutoScanFixedMappingSize	1-32	自动扫描下固定映射大小（默认为 0=4 字节）
dwReserved[4]		保留

1.5.3 ZDNMA_SLAVECONFIG

描述

此数据结构存储从站设备初始化参数，调用函数：ZDNMA_AddSlave。

```
typedef struct _tagZDNMA_SlaveConfig
{
    DWORDdwMacID;
    DWORD dwVendorID;
    DWORD dwProductType;
```

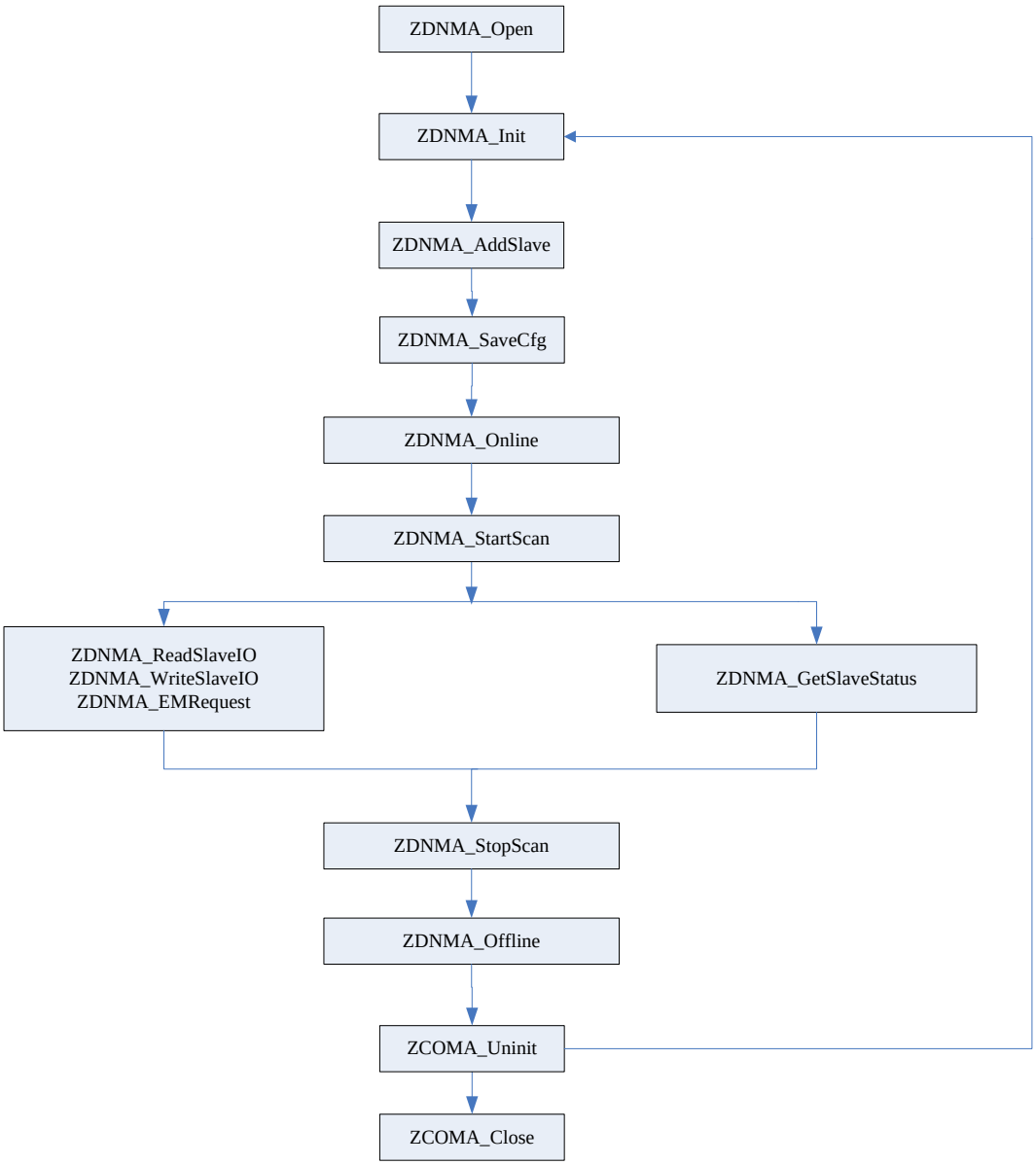
```
    DWORD dwProductCode;
    BYTE bScanType;
    BYTE bPollHz;
    BYTE bStrobeResponseLength;
    BYTE bStrobeCommandLength;
    BYTE bPollResponseLength;
    BYTE bCosCycResponseLength;
    BYTE bPollCommandLength;
    BYTE bCosCycCommandLength;
    DWORD dwpEprHeartbeat;
    DWORD dwAckTimer;
    DWORD dwInhibitTimer;
    DWORD dwReserved[4];
} ZDNMA_SLAVECONFIG, *LPZDNMA_SLAVECONFIG;
```

表 1.4ZDNMA_SLAVECONFIG 结构体参数

名称	范围	描述
bMacID	0-63	从站地址
dwVendorID	0-65535	从站厂商 ID
dwProductType	0-65535	产品类型
dwProductCode	0-65535	产品代码
bScanType		I/O 连接类型：参数如下 0000 0000 strobe 0000 0001 poll 0000 0010 strobe & poll 0000 0011 neitherstrobe nor poll 0000 1000 multicast poll 0000 0100 cos 0000 1000 cyclic 0001 0000 0 = ACK 1= UNACK 0010 0000 reserved
bPollHz	0-1	后台轮询使能
bStrobeResponseLength	0-255	位选通应答长度
bStrobeCommandLength	0-255	位选通命令长度
bPollResponseLength	0-255	轮询应答长度

bCosCycResponseLength	0-255	COS/CYC 应答长度
bPollCommandLength	0-255	轮询命令长度
bCosCycCommandLength	0-255	COS/CYC 命令长度
dwEprHeartbeat	1-65535	心跳报文时间(COS/CYC)
dwAckTimer	1-65535	应答超时时间(COS/CYC)
dwInhibitTimer	1-65535	报文生产时间(COS/CYC)
dwReserved[4]		保留

1.6 调用流程


ZDNMA_Open
ZDNMA_Init
ZDNMA_AddSlave
ZDNMA_SaveCfg
ZDNMA_Online
ZDNMA_StartScan
ZDNMA_ReadSlaveIO
ZDNMA_WriteSlaveIO
ZDNMA_EMRequest
ZDNMA_GetSlaveStatus
ZDNMA_StopScan
ZDNMA_Offline
ZCOMA_Uninit
ZCOMA_Close

1.7 错误代码总览

1.7.1 主站卡错误代码表

表 1.5 主站卡错误代码表

错误代码	错误描述	值
ERR_LOADLIB	装载 DLL 失败	41
ERR_GETPROC	获取函数地址失败	42
ERR_OPENED	设备已经被打开	43
ERR_NOTEXIST	设备不存在	44
ERR_INITDEV	初始化设备失败	45
ERR_NOTOPEN	设备没有打开	46
ERR_INVALIDPARAM	无效参数	47
ERR_INVALIDHANDLE	无效的设备句柄	48
ERR_CLOSEDEV	无法关闭设备	49
ERR_INSTALLDRIVER	驱动安装不正确	50
ERR_BUFFERTOOSMALL	用户指定的存储空间太小	52
ERR_CHANNELOPENED	通道已被打开	54
ERR_CHANNELNOTOPENED	通道未打开	55
ERR_NOMEMORY	内存不足	56
ERR_INTERNAL	其它未定义的内部错误	57
ERR_TIMEOUT	等待超时	59
ERR_SLAVEEXIST	从站已存在	60
ERR_SLAVENOTEXIST	从站不存在	61
ERR_SENDFAILED	发送数据失败	62
ERR_NODATA	没有数据	63
ERR_NOTIMPLEMENTED	该接口未实现（为以后板卡预留）	64
ERR_OFFLINE	主站没有上线	66
ERR_DISCONNECT	连接不存在	67

1.7.2 DeviceNet 通用错误代码表

表 1.6 DeviceNet 通用错误代码表

错误代码	错误描述	值
ERR_DN_NO_RESOURCE	资源不可用	2
ERR_DN_PATH_SEG_BAD	路径段错误	4
ERR_DN_NO_SVC	服务没有打开或者已经关闭	8
ERR_DN_INVALID_ATTRIB	无效的属性值	9
ERR_DN_IN_STATE	对象已处于服务请求的模式/状态	11
ERR_DN_OBJ_STATE_CONFLICT	对象在当前模式/状态下不能执行请求的服务	12
ERR_DN_NO_ATTR_SET	属性不可设	14
ERR_DN_DEV_STATE_CONFLICT	设备当前模式/状态禁止执行请求的服务	16
ERR_DN_REPLY_TOO_LARGE	应答数据太大	17
ERR_DN_DATA_UNDER	服务没有提供执行指定操作所需的足够数据	19
ERR_DN_NO_ATTR	不支持的属性	20
ERR_DN_DATA_OVER	数据太多	21
ERR_DN_NO_OBJ	对象不存在	22
ERR_DN_BAD_PARAM	参数无效	32
ERR_DN_BAD_MEMBER	无效的成员 ID	40

1.7.3 主从站设备当前状态

表 1.7 主从站设备当前状态

状态	描述	值
STASLAVE_ONLINE	从站在线	0
STAMASTER_AUTOSCAN_ACTIVE	自动扫描有效 【主站状态】	65
	保留	67
	保留	68
	保留	69
STAMASTER_DUP_MAC_FAILURE	重复 MAC ID 检测失败 【主站状态】	70

STASLAVE_SCANNER_CONFIG_ERROR	扫描列表中包含错误	71
STASLAVE_DEVICE_COMM_FAILURE	设备停止通信	72
STASLAVE_WRONG_DEV_TYPE	错误的设备类型	73
STASLAVE_PORT_OVERRUN_ERROR	端口超出（保留）	74
STAMASTER_NETWORK_FAILURE	网络通信失败 【主站状态】	75
STAMASTER_NO_MESSAGE_FOR_SCANNER	网络无数据，并且超时 【主站状态】	76
STASLAVE_WRONG_SIZE_DATA	数据长度不匹配	77
STASLAVE_NO_SUCH_DEVICE	设备在扫描列表中，但没有应答	78
STAMASTER_TRANSMIT_FAILURE	数据传输失败 【主站状态】	79
STAMASTER_IN_IDLE_MODE	设备处于空闲模式 【主站状态】	80
STAMASTER_IN_FAULT_MODE	处于错误模式 【主站状态】	81
STASLAVE_FRAGMENTATION_ERROR	分段报文顺序错误	82
STASLAVE_SLAVE_INIT_ERROR	设备初始化失败	83
STAMASTER_NOT_YET_INITIALIZED	设备尚未初始化 【主站状态】	84
STASLAVE_RUN_TIME_DATA_SIZE_ERR	连接包含错误的数据	85
STASLAVE_DEVICE_WENT_IDLE	设备自行进入空闲状态	86
STASLAVE_SHARED_MASTER_ERROR	共享的主站没有从站	87
STASLAVE_SHARED_CHOICE_ERROR	共享主站选择错误	88
STASLAVE_KEEPER_FAILED	保留	89
STAMASTER_CAN_PORT_DISABLED	用户禁止 CAN 接口 【主站状态】	90
STAMASTER_PORT_BUS_OFF	CAN 接口 BUS-OFF 【主站状态】	91
STAMASTER_PORT_POWER_OFF	CAN 传输无电源 【主站状态】	92
STAMASTER_CONFIGURATION_CRC_FAILURE	检测到一个或者多个配置功能块，CRC 错误 【主站状态】	93

	保留	94
STAMASTER_FLASH_UPDATE_IN_PROGRESS	FLASH 正在更新 【主站状态】	95
STASLAVE_IN_TEST_MODE	接口测试模式（保留）	96
STASLAVE_HALTED_BY_USER_COMMAND	用户命令停止扫描器	97
STAMASTER_FIRMWARE_FAILURE	固件错误 【主站状态】	98
STASLAVE_SYSTEM_FAILURE	系统错误看门狗超时	99

2. 免责声明

本文档提供有关广州致远电子产品的信息。本文档并未授予任何知识产权的许可，并未以明示或暗示，或以禁止发言或其它方式授予任何知识产权许可。除广州致远电子在其产品的销售条款和条件中声明的责任之外，广州致远电子概不承担任何其它责任。并且，广州致远电子对广州致远电子产品的销售和／或使用不作任何明示或暗示的担保，包括对产品的特定用途适用性、适销性或对任何专利权、版权或其它知识产权的侵权责任等，均不作担保。广州致远电子产品并非设计用于医疗、救生或维生等用途。致远电子可能随时对产品规格及产品描述做出修改，恕不另行通知。

销售与服务网络

广州致远电子股份有限公司

地址：广州市天河区车陂路黄洲工业区 7 栋 2 楼

邮编：510660

网址：www.zlg.cn

全国销售与服务电话：400-888-4005

全国服务电话：400-888-4005

销售与服务网络：

广州总公司

广州市天河区车陂路黄洲工业区 7 栋 2 楼

电话：(020)28267985　22644261

上海分公司：上海

上海市北京东路 668 号科技京城东楼 12E 室

电话：(021)53865521　53083451

北京分公司

北京市海淀区知春路 108 号豪景大厦 A 座 19 层

电话：(010)62536178　62635573

上海分公司：南京

南京市珠江路 280 号珠江大厦 1501 室

电话：(025)68123923　68123920

深圳分公司

深圳市福田区深南中路 2072 号电子大厦 12 楼

电话：(0755)83640169　83783155

上海分公司：杭州

杭州市天目山路 217 号江南电子大厦 502 室

电话：(0571)89719491　89719493

武汉分公司

武汉市洪山区广埠屯珞瑜路 158 号 12128 室（华中电脑数码市场）

电话：(027)87168497　87168397

重庆分公司

重庆市九龙坡区石桥铺科园一路二号大西洋国际大厦（赛格电子市场）2705 室

电话：(023)68796438　68797619

成都分公司

成都市一环路南二段 1 号数码科技大厦 403 室

电话：(028)85439836　85432683

西安办事处

西安市长安北路 54 号太平洋大厦 1201 室

电话：(029)87881295　87881296

请您用以上方式联系我们，我们会为您安排样机现场演示，感谢您对我公司产品的关注！