

类别	内容
关键词	EtherCAT主站接口库使用
摘要	

## 修订历史

版本	日期	原因
V0.01	2020/08/05	创建文档

## 目 录

1. 函数库简介.....	1
1.1 主站运行流程.....	1
2. 函数库数据结构以及接口说明.....	3
2.1 数据类型和结构定义.....	3
2.2 接口库函数说明.....	3
2.2.1 EcatMasterLibInit.....	3
2.2.2 EcatMasterLibExit.....	3
2.2.3 EcatCreateMaster.....	3
2.2.4 EcatFreeMaster.....	3
2.2.5 EcatConnectMaster.....	4
2.2.6 EcatDisconnectMaster.....	4
2.2.7 EcatResetStatistics.....	4
2.2.8 EcatLoadConfigFromFileA.....	4
2.2.9 EcatSetExternalCycleTime.....	5
2.2.10 EcatStartCyclicOperation.....	5
2.2.11 EcatStopCyclicOperation.....	5
2.2.12 EcatSetAutoRecoveryTimeout.....	6
2.2.13 EcatMasterServerCreate.....	6
2.2.14 EcatMasterServerStart.....	6
2.2.15 EcatMasterServerStop.....	7
2.2.16 EcatMasterServerDelete.....	7
2.2.17 EcatRequestMasterState.....	7
2.2.18 EcatGetMasterState.....	7
2.2.19 EcatSetExtCtrlTaskHandler.....	7
2.2.20 EcatResetExtCtrlTaskHandler.....	8
2.2.21 EcatCoeSDODownload.....	8
2.2.22 EcatCoeSDOUpload.....	8
2.2.23 EcatGetProcessImageSize.....	9
3. 免责声明.....	10

## 1. 函数库简介

EtherCAT 主站库以 Linux 下动态链接库(so)的方式提供，可以实现 EtherCAT 主站控制等功能。主站可以通过函数库启动主站 RPC 服务器，RPC 客户端可以通过网络连接到主站 RPC 服务器，控制主站完成 EtherCAT 相关的功能。

函数库包含 `zecm_zh.h` 头文件和 `libzecm.so` 动态链接库文件。`zecm_zh.h` 文件包含了 EtherCAT 主站的数据类型，数据结构以及函数声明，用户开发自定义主站需要配置交叉编译环境，将编译好的程序下载到主站上，在主站运行编译后的程序以完成 EtherCAT 的通信控制功能。

### 1.1 主站运行流程

主站可以运行用户编写的程序，主站程序运行一般遵循的流程如图 1.1 所示。

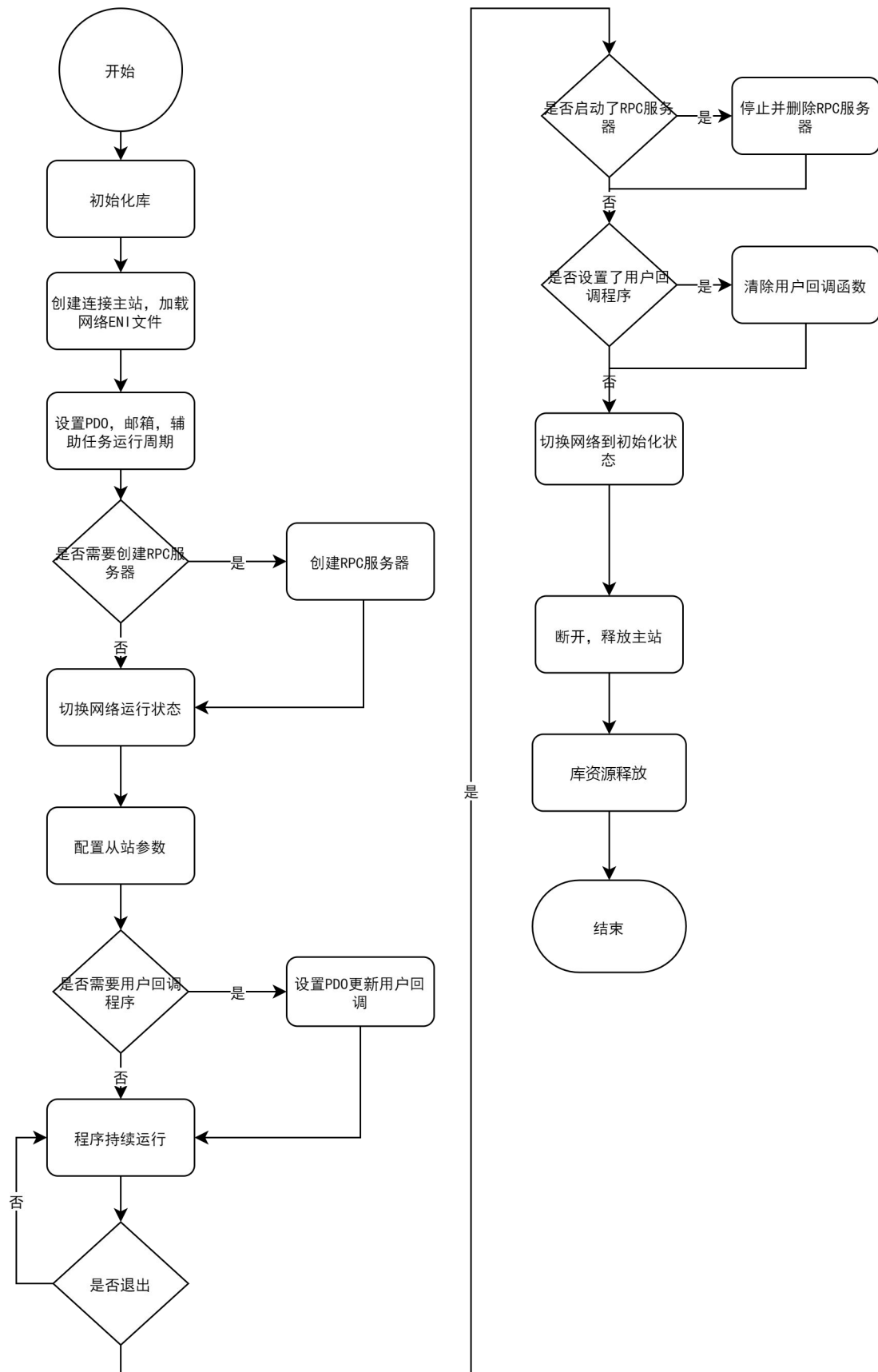


图 1.1 主站运行一般流程

## 2. 函数库数据结构以及接口说明

### 2.1 数据类型和结构定义

数据类型结构体相见提供主站函数库头文件中的说明。

### 2.2 接口库函数说明

#### 2.2.1 EcatMasterLibInit

该函数用于初始化主站接口库，调用接口库其他函数前须先调用此函数初始化接口库。

```
ECAT_RESULT EcatMasterLibInit(ECAT_CSTR pszIniFileName);
```

##### 参数

*pszIniFileName*: [IN]主站库配置文件(INI文件,非ENI文件)路径,当前使用ECAT\_NULL表示使用默认参数。

##### 返回值

返回ECAT\_S\_OK表示初始化成功,返回其他数值表示初始化失败,返回值为错误码,错误码含义见头文件中相关定义。

#### 2.2.2 EcatMasterLibExit

该函数用于主站库资源释放,所有调用库相关的API结束后需调用此函数进行库释放操作。

```
ECAT_RESULT EcatMasterLibExit();
```

##### 参数

无

##### 返回值

返回ECAT\_S\_OK表示成功,返回其他值表示出现错误,范围值为错误码,错误码含义见头文件中相关定义。

#### 2.2.3 EcatCreateMaster

该函数用于创建主站对象,创建的主站通过主站句柄供后续操作使用。

```
ECAT_RESULT EcatCreateMaster( ECAT_HANDLE* phMaster);
```

##### 参数

*phMaster*: [OUT]指向返回主站句柄变量的指针。

##### 返回值

返回ECAT\_S\_OK表示成功,返回其他值表示出现错误,范围值为错误码,错误码含义见头文件中相关定义。

#### 2.2.4 EcatFreeMaster

该函数用于释放主站资源,和EcatCreateMaster()配对使用。

```
ECAT_RESULT EcatFreeMaster(ECAT_HANDLE hMaster);
```

##### 参数

*hMaster*: [IN]主站句柄。

### 返回值

返回 ECAT\_S\_OK 表示成功，返回其他值表示出现错误，范围值为错误码，错误码含义见头文件中相关定义。

## 2.2.5 EcatConnectMaster

该函数用于连接主站设备，协议栈和驱动建立连接。

```
ECAT_RESULT EcatConnectMaster(ECAT_HANDLE hMaster, ECAT_DWORD dwType, const ECAT_WCHAR* pwszParams1, const ECAT_WCHAR* pwszParams2, EcatDriver* pDriver);
```

### 参数

*hMaster*: [IN]主站句柄。

*dwType*: [IN]设备类型，可选类型为 ECAT\_MASTER\_REDUNDANCY\_TYPE\_NORMAL 和 ECAT\_MASTER\_REDUNDANCY\_TYPE\_REDUNDANCY。

*pwszParams1*: [IN]主网卡名称。

*pwszParams2*: [IN]冗余网卡名称，当 *dwType* 为 ECAT\_MASTER\_REDUNDANCY\_TYPE\_NORMAL 时此参数必须为 ECAT\_NULL。

*pDriver*: [IN]指向驱动的指针，如果为空使用默认驱动。

### 返回值

返回 ECAT\_S\_OK 表示成功，返回其他值表示出现错误，范围值为错误码，错误码含义见头文件中相关定义。

## 2.2.6 EcatDisconnectMaster

该函数用于主站协议栈断开和驱动的连接，和 EcatConnectMaster() 配对使用。

```
ECAT_RESULT EcatDisconnectMaster(ECAT_HANDLE hMaster);
```

### 参数

*hMaster*: [IN]主站句柄。

### 返回值

返回 ECAT\_S\_OK 表示成功，返回其他值表示出现错误，范围值为错误码，错误码含义见头文件中相关定义。

## 2.2.7 EcatResetStatistics

该函数用于复位主站的网络统计信息。

```
ECAT_RESULT EcatResetStatistics(ECAT_HANDLE hMaster);
```

### 参数

*hMaster*: [IN]主站句柄。

### 返回值

返回 ECAT\_S\_OK 表示成功，返回其他值表示出现错误，范围值为错误码，错误码含义见头文件中相关定义。

## 2.2.8 EcatLoadConfigFromFileA

该函数使用 ENI 文件加载主站配置，路径使用 ANSI 编码。

```
ECAT_RESULT EcatLoadConfigFromFileA(ECAT_HANDLE hMaster, const ECAT_CHAR* pszFilePath);
```

### 参数

*hMaster*: [IN]主站句柄。

*pszFilePath*: [IN] ENI 文件路径(ANSI 编码)。

### 返回值

返回 ECAT\_S\_OK 表示成功, 返回其他值表示出现错误, 范围值为错误码, 错误码含义见头文件中相关定义。

## 2.2.9 EcatSetExternalCycleTime

该函数用于配置外部循环时间, 在以同步模式 2 启动主机之前, 必须配置外部循环。在其他情况下, 外部循环的值被忽略

```
ECAT_RESULT EcatSetExternalCycleTime(ECAT_HANDLE hMaster, ECAT_TIME_NS nsExternalCycleTime);
```

### 参数

*hMaster*: [IN]主站句柄。

*nsExternalCycleTime*: [IN]外部循环时间, 单位 ns。

### 返回值

返回 ECAT\_S\_OK 表示成功, 返回其他值表示出现错误, 范围值为错误码, 错误码含义见头文件中相关定义。

## 2.2.10 EcatStartCyclicOperation

该函数用于启动主从站之间周期数据交换。

```
ECAT_RESULT EcatStartCyclicOperation(ECAT_HANDLE hMaster, ECAT_DWORD dwCycleTimeoutMicrosec, ECAT_DWORD dwSubCycleTimeoutMicrosec, ECAT_DWORD dwBasePriority);
```

### 参数

*hMaster*: [IN]主站句柄。

*dwCycleTimeoutMicrosec*: [IN]PDO 更新周期时间, 单位微秒(us)。

*dwSubCycleTimeoutMicrosec*: [IN]邮箱数据更新周期时间, 单位微秒(us)。

*dwBasePriority*: [IN]此参数已不再使用。

### 返回值

返回 ECAT\_S\_OK 表示成功, 返回其他值表示出现错误, 范围值为错误码, 错误码含义见头文件中相关定义。

## 2.2.11 EcatStopCyclicOperation

该函数用于停止主从站之间首期数据交换。

```
ECAT_RESULT EcatStopCyclicOperation(ECAT_HANDLE hMaster);
```

### 参数

*hMaster*: [IN]主站句柄。

### 返回值

返回 ECAT\_S\_OK 表示成功, 返回其他值表示出现错误, 范围值为错误码, 错误码含义



见头文件中相关定义。

### 2.2.12 EcatSetAutoRecoveryTimeout

该函数用于设置辅助任务(如扫描总线上从站数量等)时间。

```
ECAT_RESULT EcatSetAutoRecoveryTimeout(ECAT_HANDLE hMaster, ECAT_DWORD dwTimeout);
```

#### 参数

*hMaster*: [IN]主站句柄。

*dwTimeout*: [IN]超时时间(毫秒)。

#### 返回值

返回 ECAT\_S\_OK 表示成功, 返回其他值表示出现错误, 范围值为错误码, 错误码含义见头文件中相关定义。

### 2.2.13 EcatMasterServerCreate

该函数创建一个 RPC(远程调用服务)服务器, 供客户端连接。用户可以通过 RPC 客户端连接到 RPC 服务器, 对主站进行远程控制。

```
ECAT_RESULT EcatMasterServerCreate(ECAT_HANDLE* phServer, ECAT_WORD wType, ECAT_BOOL bAutostartServer, ECAT_UINT32 nPort, ECAT_BOOL bUseOverUdpMailbox, ECAT_HANDLE hMaster, ECAT_HANDLE hDriver);
```

#### 参数

*phServer*: [OUT]指向返回 RPC 服务器句柄的指针。

*wType*: [IN]创建的服务器类型, 参考 EcatMasterServerType。

*bAutostartServer*: [IN]该参数目前忽略, 用于创建完后是否启动服务, ECAT\_TRUE: 是; ECAT\_FALSE: 否。

*nPort*: [IN]端口号。

*bUseOverUdpMailbox*: [IN]ECAT\_TRUE:使用 OverUdpMailBox, ECAT\_FALSE 表示不使用。

*hMaster*: [IN]主站句柄。

*hDriver*: [IN]驱动句柄, 如果为空, 使用默认驱动

#### 返回值

返回 ECAT\_S\_OK 表示成功, 返回其他值表示出现错误, 范围值为错误码, 错误码含义见头文件中相关定义。

### 2.2.14 EcatMasterServerStart

该函数用于启动 RPC 服务器。

```
ECAT_RESULT EcatMasterServerStart(ECAT_HANDLE hServer);
```

#### 参数

*hServer*: [IN]RPC 服务器句柄。

#### 返回值

返回 ECAT\_S\_OK 表示成功, 返回其他值表示出现错误, 范围值为错误码, 错误码含义见头文件中相关定义。

## 2.2.15 EcatMasterServerStop

该函数用于停止 RPC 服务器。

```
ECAT_RESULT EcatMasterServerStop(ECAT_HANDLE hServer);
```

### 参数

*hServer*: [IN]RPC 服务器句柄。

### 返回值

返回 ECAT\_S\_OK 表示成功，返回其他值表示出现错误，范围值为错误码，错误码含义见头文件中相关定义。

## 2.2.16 EcatMasterServerDelete

该函数用于删除主站 RPC 服务器。

```
ECAT_RESULT EcatMasterServerDelete(ECAT_HANDLE hServer);
```

### 参数

*hServer*: [IN]RPC 服务器句柄。

### 返回值

返回 ECAT\_S\_OK 表示成功，返回其他值表示出现错误，范围值为错误码，错误码含义见头文件中相关定义。

## 2.2.17 EcatRequestMasterState

该函数用于请求主机切换状态，主机会切换所在网络上主站和所有从站的状态。

```
ECAT_RESULT EcatRequestMasterState(ECAT_HANDLE hMaster, ECAT_BYTE byState);
```

### 参数

*hMaster*: [IN]主站句柄。

*byState*: [IN]请求状态，详见 EcatState。

### 返回值

返回 ECAT\_S\_OK 表示成功，返回其他值表示出现错误，范围值为错误码，错误码含义见头文件中相关定义。

## 2.2.18 EcatGetMasterState

该函数用于获取主站当前的状态。

```
ECAT_RESULT EcatGetMasterState(ECAT_HANDLE hMaster, ECAT_BYTE* pbyState);
```

### 参数

*hMaster*: [IN]主站句柄。

*pbyState*: [OUT]指向主站当前状态变量的指针，详见 EcatState。

### 返回值

返回 ECAT\_S\_OK 表示成功，返回其他值表示出现错误，范围值为错误码，错误码含义见头文件中相关定义。

## 2.2.19 EcatSetExtCtrlTaskHandler

该函数用于设置用户处理程序，每次过程数据更新时调用用户设置的处理程序。

```
ECAT_RESULT EcatSetExtCtrlTaskHandler(ECAT_HANDLE hMaster, EcatExtCtrlTaskHandler pfnHandler,  
ECAT_LPVOID pvCtrlTaskArg);
```

## 参数

*hMaster*: [IN]主站句柄。

*pfnHandler*: [IN]用户的回调函数指针。

*pvCtrlTaskArg*: [IN]用户回调函数参数。

## 返回值

返回 ECAT\_S\_OK 表示成功，返回其他值表示出现错误，范围值为错误码，错误码含义见头文件中相关定义。

### 2.2.20 EcatResetExtCtrlTaskHandler

该函数用于复位外部任务处理程序的执行并设置为默认处理程序（即空处理程序）。

```
ECAT_RESULT EcatResetExtCtrlTaskHandler(ECAT_HANDLE hMaster);
```

## 参数

*hMaster*: [IN]主站句柄。

## 返回值

返回 ECAT\_S\_OK 表示成功，返回其他值表示出现错误，范围值为错误码，错误码含义见头文件中相关定义。

### 2.2.21 EcatCoeSDODownload

该函数用于使用 COE 进行从站 SDO 下载操作，即写入从站的对象字典。

```
ECAT_RESULT EcatCoeSDODownload(ECAT_HANDLE hMaster, ECAT_WORD wSlaveId, ECAT_WORD  
wIndex, ECAT_BYTE bySubIndex, ECAT_BOOL bCompleteAccess, ECAT_DWORD* pdwDataLen,  
ECAT_BYTE* pbyData, ECAT_DWORD dwMaxDataLen, ECAT_DWORD dwTimeOut);
```

## 参数

*hMaster*: [IN]主站句柄。

*wSlaveId*: [IN]从节点位置索引。

*wIndex*: [IN] SDO 索引。

*bySubIndex*: [IN] SDO 子索引。

*bCompleteAccess*: [IN] SDO 完全访问，需要从站支持。

*pdwDataLen*: [IN] 指向要写入数据长度的指针。

*pbyData*: [IN] 要写入数据的地址。

*dwMaxDataLen*: [IN]最大数据长度。

*dwTimeOut*: [IN] 超时。

## 返回值

返回 ECAT\_S\_OK 表示成功，返回其他值表示出现错误，范围值为错误码，错误码含义见头文件中相关定义。

### 2.2.22 EcatCoeSDOUpload

该函数用于使用 COE 进行从站 SDO 上传操作，即读取从站的对象字典。

```
ECAT_RESULT EcatCoeSDOUpload (ECAT_HANDLE hMaster, ECAT_WORD wSlaveId, ECAT_WORD wIndex, ECAT_BYTE bySubIndex, ECAT_BOOL bCompleteAccess, ECAT_DWORD* pdwDataLen, ECAT_BYTE* pbyData, ECAT_DWORD dwMaxDataLen, ECAT_DWORD dwTimeOut);
```

## 参数

*hMaster*: [IN]主站句柄。

*wSlaveId*: [IN]从节点位置索引。

*wIndex*: [IN] SDO 索引。

*bySubIndex*: [IN] SDO 子索引。

*bCompleteAccess*: [IN] SDO 完全访问，需要从站支持。

*pdwDataLen*: [IN][OUT] 读取数据长度的指针。

*pbyData*: [OUT] 存放读取数据的地址。

*dwMaxDataLen*: [IN]最大数据长度。

*dwTimeOut*: [IN] 超时。

## 返回值

返回 ECAT\_S\_OK 表示成功，返回其他值表示出现错误，范围值为错误码，错误码含义见头文件中相关定义。

### 2.2.23 EcatGetProcessImageSize

该函数用于获取主站过程映像大小。

```
ECAT_RESULT EcatGetProcessImageSize(ECAT_HANDLE hMaster, ECAT_WORD* pwInputsSize, ECAT_WORD* pwOutputsSize);
```

## 参数

*hMaster*: [IN]主站句柄。

*pwInputsSize*: [OUT]指向保存输入过程映像长度变量的指针。

*pwOutputsSize*: [OUT]指向保存输出过程映像长度变量的指针。

## 返回值

返回 ECAT\_S\_OK 表示成功，返回其他值表示出现错误，范围值为错误码，错误码含义见头文件中相关定义。

## 3. 免责声明

本着为用户提供更好服务的原则，广州致远电子股份有限公司（下称“致远电子”）在本手册中将尽可能地向用户呈现详实、准确的产品信息。但鉴于本手册的内容具有一定的时效性，致远电子不能完全保证该文档在任何时段的时效性与适用性。致远电子有权在没有通知的情况下对本手册上的内容进行更新，恕不另行通知。为了得到最新版本的信息，请尊敬的用户定时访问致远电子官方网站或者与致远电子工作人员联系。感谢您的包容与支持！