

IGH EtherCAT

码龄3年 暂无认证

28

83万+

158万+

5万+

原创

周排名

总排名

访问

等级

920

114

46

161

374

积分

粉丝

获赞

评论

收藏

私信

关注

搜博主文章

- 热门文章
- stm32f103介绍

9261
- Linux下IGH Ethercat Master安装

3505
- igh ethercat master及简单介绍

3402
- 修改Ubuntu内核版本

3169
- 在树莓派4.x内核下安装Igh EtherCAT master主站

3036

- 最新评论
- IGH EtherCAT应用层控制电机代码 qq_36126487: 状态字是0怎么解决的啊
- IGH EtherCAT应用层控制电机代码 qq_36126487: 你们的可以进入OP模式了吗，怎么解决的啊前辈们
- Linux下IGH Ethercat Master安装 IGH EtherCAT: 这个没遇到过，你可以百度一下，这种错误应该是有答案的
- Linux下IGH Ethercat Master安装 IGH EtherCAT: 都可以的，我都试过，用虚拟机的话需要设置桥接到物理网卡
- Linux下IGH Ethercat Master安装 IGH EtherCAT: 你configure的时候有没有加选项--enable-e1000

您愿意向朋友推荐“博客详情页”吗？

强烈不推荐

不推荐

一般般

推荐

强烈推荐

- 最新文章
- U盘变小恢复方法
- igh etherlab主站介绍

IGH EtherCAT应用层控制电机代码

原创 IGH EtherCAT 于 2020-05-15 10:47:33 发布 2907 收藏 39

分类专栏: ethercat 文章标签: c语言 ethercat

ethercat

专栏收录该内容

12 订阅 13 篇文章 订阅专栏

在主站配置好之后，连接从站，我用的是雷赛的伺服，对于大部分来说改个pid，vid，应该就可以用，这个是用的pv模式应该还是容易懂的，我把一些用不到的代码给阉割了

```
1  /**
2   * compile : gcc test.c -o test -I/opt/etherlab/include -L/opt/etherlab/lib -lethercat
3   */
4  #include <errno.h>
5  #include <signal.h>
6  #include <stdio.h>
7  #include <string.h>
8  #include <sys/resource.h>
9  #include <sys/time.h>
10 #include <sys/types.h>
11 #include <unistd.h>
12 #include <sys/mman.h>
13
14 /*****
15
16 #include "ecrt.h"
17
18 /*****
19
20 /*Application Parameters*/
21 #define TASK_FREQUENCY      10 /*Hz*/
22 #define TARGET_VELOCITY    110000 /*target velocity*/
23 #define PROFILE_VELOCITY   3 /*Operation mode for 0x6060:0*/
24
25 /*****
26
27 /*EtherCAT*/
28 static ec_master_t *master = NULL;
29 static ec_master_state_t master_state = {};
30
31 static ec_domain_t *domain1 = NULL;
32 static ec_domain_state_t domain1_state = {};
33
34 static ec_slave_config_t *sc;
35 static ec_slave_config_state_t sc_state = {};
36
37 /*****
38
39 /*Process Data*/
40 static uint8_t *domain1_pd = NULL;
41
42 #define DM3E      0,0 /*EtherCAT address on the bus*/
43
44 #define VID_PID    0x00004321,0x00008100 /*Vendor ID, product code*/
```

- 分类专栏
- ethercat

13篇
- linux

4篇
- 其他

1篇
- c 语言基础

4篇
- stm32

5篇
- c++

2篇

```
46 /*Offsets for PDO entries*/
47 static struct{
48     unsigned int operation_mode;
49     unsigned int ctrl_word;
50     unsigned int target_velocity;
51     unsigned int status_word;
52     unsigned int current_velocity;
53     unsigned int current_position;
54 }offset;
55
56 const static ec_pdo_entry_reg_t domain1_regs[] = {
57     {DM3E, VID_PID, 0x6040, 0, &offset.ctrl_word},
58     {DM3E, VID_PID, 0x6060, 0, &offset.operation_mode },
59     {DM3E, VID_PID, 0x60FF, 0, &offset.target_velocity},
60     {DM3E, VID_PID, 0x6041, 0, &offset.status_word},
61     {DM3E, VID_PID, 0x606C, 0, &offset.current_velocity},
62     {DM3E, VID_PID, 0x6064, 0, &offset.current_position},
63     {}
64 };
65
66 /*****
67  *Config PDOs*/
68 static ec_pdo_entry_info_t device_pdo_entries[] = {
69     /*RxPdo 0x1600*/
70     {0x6040, 0x00, 16},
71     {0x6060, 0x00, 8 },
72     {0x60FF, 0x00, 32},
73     /*TxPdo 0x1A00*/
74     {0x6041, 0x00, 16},
75     {0x606C, 0x00, 32},
76     {0x6064, 0x00, 32}
77 };
78
79 static ec_pdo_info_t device_pdos[] = {
80     //RxPdo
81     {0x1600, 3, device_pdo_entries + 0 },
82     //TxPdo
83     {0x1A00, 3, device_pdo_entries + 3 }
84 };
85
86 static ec_sync_info_t device_syncs[] = {
87     { 0, EC_DIR_OUTPUT, 0, NULL, EC_WD_DISABLE },
88     { 1, EC_DIR_INPUT, 0, NULL, EC_WD_DISABLE },
89     { 2, EC_DIR_OUTPUT, 1, device_pdos + 0, EC_WD_DISABLE },
90     { 3, EC_DIR_INPUT, 1, device_pdos + 1, EC_WD_DISABLE },
91     { 0xFF}
92 };
93
94 /*****
95
96
97
98 void check_domain1_state(void)
99 {
100     ec_domain_state_t ds;
101     ecrt_domain_state(domain1, &ds);
102     if (ds.working_counter != domain1_state.working_counter)
103     {
104         printf("Domain1: WC %u.\n", ds.working_counter);
105     }
```

```
106     if (ds.wc_state != domain1_state.wc_state)
107     {
108         printf("Domain1: State %u.\n", ds.wc_state);
109     }
110     domain1_state = ds;
111 }
112
113 void check_master_state(void)
114 {
115     ec_master_state_t ms;
116     ecrt_master_state(master, &ms);
117     if (ms.slaves_responding != master_state.slaves_responding)
118     {
119         printf("%u slave(s).\n", ms.slaves_responding);
120     }
121     if (ms.al_states != master_state.al_states)
122     {
123         printf("AL states: 0x%02X.\n", ms.al_states);
124     }
125     if (ms.link_up != master_state.link_up)
126     {
127         printf("Link is %s.\n", ms.link_up ? "up" : "down");
128     }
129     master_state = ms;
130 }
131
132 /*****/
133
134 void check_slave_config_states(void)
135 {
136     ec_slave_config_state_t s;
137     ecrt_slave_config_state(sc, &s);
138     if (s.al_state != sc_state.al_state)
139     {
140         printf("slave: State 0x%02X.\n", s.al_state);
141     }
142     if (s.online != sc_state.online)
143     {
144         printf("slave: %s.\n", s.online ? "online" : "offline");
145     }
146     if (s.operational != sc_state.operational)
147     {
148         printf("slave: %soperational.\n", s.operational ? "" : "Not ");
149     }
150     sc_state = s;
151 }
152
153 /*****/
154
155 void cyclic_task()
156 {
157     static uint16_t command=0x004F; //用来帮助判断状态字的值
158     uint16_t status;
159     /*Receive process data*/
160     ecrt_master_receive(master);
161     ecrt_domain_process(domain1);
162     /*Check process data state(optional)*/
163     check_domain1_state();
164
165     //Check for master state
166     check_master_state();
```



```

167 //Check for slave configuration state(s)
168 check_slave_config_states();
169 /*Read state*/
170 status = EC_READ_U16(domain1_pd + offset.status_word); //读取状态字
171
172 //DS402 CANOpen over EtherCAT status machine
173 if( (status & command) == 0x0040 )
174 {
175     EC_WRITE_U16(domain1_pd + offset.ctrl_word, 0x0006 );
176     EC_WRITE_S8(domain1_pd + offset.operation_mode, PROFILE_VELOCITY);
177 //设置控制模式
178     command = 0x006F;
179 }
180
181 else if( (status & command) == 0x0021)
182 {
183     EC_WRITE_U16(domain1_pd + offset.ctrl_word, 0x0007 );
184     command = 0x006F;
185 }
186
187 else if( (status & command) == 0x0023)
188 {
189     EC_WRITE_U16(domain1_pd + offset.ctrl_word, 0x000f );
190     command = 0x006F;
191 }
192 //operation enabled
193
194 else if( (status & command) == 0x0027)
195 {
196     EC_WRITE_S32(domain1_pd + offset.target_velocity, TARGET_VELOCITY);
197     EC_WRITE_U16(domain1_pd + offset.ctrl_word, 0x001f );
198 }
199
200 /*Send process data*/
201 ecrt_domain_queue(domain1);
202 ecrt_master_send(master);
203 }
204
205 /*****
206
207 int main(int argc, char **argv)
208 {
209     printf("Requesting master...\n");
210     master = ecrt_request_master(0);
211     if (!master)
212     {
213         exit(EXIT_FAILURE);
214     }
215
216     domain1 = ecrt_master_create_domain(master);
217     if (!domain1)
218     {
219         exit(EXIT_FAILURE);
220     }
221     if (!(sc = ecrt_master_slave_config(master, DM3E, VID_PID)))
222     {
223         fprintf(stderr, "Failed to get slave configuration for slave!\n");
224         exit(EXIT_FAILURE);
225     }
226     printf("Configuring PDOs...\n");
227     if (ecrt_slave_config_pdos(sc, EC_END, device_syncs))

```

```
228     {
229         fprintf(stderr, "Failed to configure slave PDOs!\n");
230         exit(EXIT_FAILURE);
231     }
232     else
233     {
234         printf("**Success to configuring slave PDOs*\n");
235     }
236
237     if (ecrt_domain_reg_pdo_entry_list(domain1, domain1_regs))
238     {
239         fprintf(stderr, "PDO entry registration failed!\n");
240         exit(EXIT_FAILURE);
241     }
242
243     printf("Activating master...\n");
244     if (ecrt_master_activate(master)) {
245         exit(EXIT_FAILURE);
246     }
247     else
248     {
249         printf("**Master activated*\n");
250     }
251     if (!(domain1_pd = ecrt_domain_data(domain1))) {
252         exit(EXIT_FAILURE);
253     }
254
255     printf("**It's working now*\n");
256
257     while (1)
258     {
259         usleep(100000/TASK_FREQUENCY);
260         cyclic_task();
261     }
262     return EXIT_SUCCESS;
263 }
264
```

EtherCAT igh主站**控制**3个台dasdaa2伺服转圈圈 12-03
EtherCAT igh主站**控制**3个台dasdaa2伺服转圈圈。包括伺服使能、**控制**3个**电机**转圈圈、每秒读取**电机**的实际位置、运行90秒自动退出程序。。需要...

igh ethercat 台达**电机控制代码** 10-28
ethercat主站资料较少，相关**应用**就更少了。好不容易找到一段**控制**台达**电机**的**代码**，亲自测试过能驱动点击标准的ds402协议

评论 23 您还未登录，请先 [登录](#) 后发表或查看评论

基于EtherCAT实时通信的**电机驱动控制_ethercat控制电机-其它代码...** 3-8
为了将EtherCAT快速**应用**到**电机驱动控制**系统中,采用IntervalZero公司的KingStar Motion软件,设计了一种基于EtherCAT实时通信的**电机驱动控制**方案,并...

ethercat电机,mfc简单连接程序_c++ethercat-C++**代码**类资源-CSDN文库 3-14
为了将EtherCAT快速**应用**到**电机驱动控制**系统中,采用IntervalZero公司的KingStar Motion软件,设计了一种基于EtherCAT实时通信的**电机驱动控制**方案,并...

igh实时**控制电机**动起来的一个小项目，包括编译环境，及**代码**。 12-04
IGH, xenoma**控制电机**，下载解压后，需要修改在mian.c里面把自己**电机**的ID换一下就能运行了。这里面是视频演示（ https://www.bilibili.com/video/BV1r...

igh_test.rar 09-29
目前ethercat主站资料比较少，大部分主站**代码**收费，开源igh驱动**电机应用**在国内网站几乎没有资料，好不容易找到**控制**台达ethercat驱动器**代码**，本人...

【Ethercat CSP**控制**模式下**电机**卡顿原因分析及解决方法... 3-25
国外商业化的ethercat主站价格昂贵,国内尚未有全自主的ethercat主站。德国开源**IGH ethercat**主站成为近几年国内研究**应用**的主流方向。大多数**IGH**主站...

ighethercat控制台达电机(ds402)_DS402-C代码类资源-CSDN文库	3-12
为了将EtherCAT快速应用到电机驱动控制系统中,采用IntervalZero公司的KingStar Motion软件,设计了一种基于EtherCAT实时通信的电机驱动控制方案,并...	
EtherCAT (学习笔记) 热门推荐	pwl999的博客 3万+
文章目录1. 简介1.1 运动控制1.2 实时以太网1.3 EtherCAT2. EtherCAT原理介绍2.1 实时性2.2 端口管理2.3 EtherCAT网络拓扑2.4 EtherCAT网络协议栈2....	
EtherCAT igh主站控制埃斯顿伺服 (csp模式)	cln512的博客 3547
完整代码 算了、干脆直接贴代码了。最近都在搞EtherCAT主站、从站的应用, 过段时间再分享一些项目中基础的东西。驱动伺服主要还是参考厂商提供的...	
在CODESYS中通过EtherCAT总线驱动单个电机_澜澜爱睡觉...	3-25
本文讲述了手上有支持EtherCAT总线的驱动器和步进电机的情况下,如何通过CODESYS添加EtherCAT的主站从站,控制电机转动。 本文所用设备:普通P...	
SOEM-EtherCAT-Cia402电机控制_一蓑烟雨任平生	3-9
1.环境配置 SOEM主站环境搭建及连接板子测试:https://blog.csdn.net/zhandouhu/article/details/108424677 EtherCat开源主站SOEM在windows下工程配...	
Ethercat系列---IGH DC模式测试电机程序 (Preempt-RT)	中华田园巨龙 2698
先上代码 #include <errno.h> #include <signal.h> #include <stdio.h> #include <string.h> #include <sys/resource.h> #include <sys/time.h> #include <sy...	
igh ethercat应用层代码分析	qq_43530144的博客 1780
igh ethercat应用层代码分析 igh ethercat可以自己编写应用层代码与从站进行数据交互, 本身主站给了例子, 在example文件夹下, 根据自己平台选择看...	
IGH 控制伺服的一个例子 (验证中) 最新发布	xiaoding_ding的专栏 79
介绍了一个伺服控制的程序例子 (使用子线程通讯交换数据), 主线程与用户交互。还在测试中。	
EtherCAT igh主站控制埃斯顿伺服回零	cln512的博客 1144
完整代码 #include <errno.h> #include <signal.h> #include <stdio.h> #include <string.h> #include <sys/resource.h> #include <sys/time.h> #include <sy...	
igh EtherCAT Master 的使用	qq_46136275的博客 1123
igh EtherCAT Master 的使用EtherCAT Mater 的启动ethercat masterethercat slavesethercat cstruct二级目录三级目录 EtherCAT Mater 的启动 输入命令...	
基于LinuxCNC下的EtherCAT的伺服电机控制《二》	思泽之思 4018
第二阶段, 伺服电机动起来了, 接下来的连接HAL, ini xml 文件, 实现界面控制, 不再是命令行 继续更新中	
igh-Ethercat-master:基于igh EtherCAT主控松下和安川伺服电机	05-18
igh-Ethercat大师 EtherCAT Master源代码 基于igh EtherCAT主控松下和安川伺服电机	
基于AM335x的Ethercat实现	07-23
关于使用AM335x的相关开发平台实现EtherCAT通信的案例	
基于kithara实现的伺服电机控制代码-VisualStudio-ethercat	07-20
基于kithara实现的伺服电机控制代码-VisualStudio-ethercat 基于kithara实现的伺服电机控制代码-VisualStudio-ethercat	
SOEM控制伺服电机	qq_43530144的博客 2653
我只完成了pv模式, 对于csp模式我不知道是哪里出现了问题, 有知道的可以在下方评论 这个代码我的pv模式可以正常运行和控制电机, csp模式可以使能...	
IGH_Master主站配置驱动伺服电机和变频器总结	A-CAT的博客 1454
IGH_Master主站配置驱动伺服电机和变频器总结 Ethercat是倍福公司提出的一种工业现场总线协议, 具有很好的实时性, IGH是一种开源的Ethercat主站...	
记一次坑爹的ioremap bug	中华田园巨龙 165
在进行IGH协议栈修改时, 遇到这样一个宏定义 #define EC_WRITE_U16(DATA, VAL) \do { \if((uint16_t)(DATA) % 2 ==0) \{ *((uint16_t *) (DATA)) = ...	
ethercat移植至ARM	Tanya 1080
需求:移植ethercat至imx6q板, 使用内核版本为4.1.15 准备工作: 1.在官网 http://www.etherlab.org/en/ethercat/ 下载ethercat-1.5.2.tar.bz2以及ethercat-1....	
linux5.4内核搭建igh主站第二次尝试	ze3000的博客 211
想了一下, 还是重来一遍。 tar -vxvf ethercat-1.5.2.tar.bz2 cd ethercat-1.5.2/ ./configure --enable-8139too=no make sudo make modules 目录 1. netdev...	
“相关推荐”对你有帮助？	
<div><div></div>非常没帮助</div> <div><div></div>没帮助</div> <div><div></div>一般</div> <div><div></div>有帮助</div> <div><div></div>非常有帮助</div>	
©2022 CSDN 皮肤主题: 1024 设计师: 我叫白小胖 返回首页	





IGH EtherCAT

关注



1



23



39



专栏目录