

IgH Etherlab EtherCAT master

Linux

author: yc

IgH Etherlab 是建立在linux下的一个开源的EtherCAT通信主站，虽然近年来更新频率较低，但有后续开源社区的维护。Synapticon驱动器官方也提供了IgH Etherlab EtherCAT master的相关维护与优化，详细内容可以在[Github/Synapticon](#)中查看。

IgH 在codeforge上提供的最新版本是stable-1.5,支持4.x内核，实际测试在ubuntu20.04上也可以正常安装。能否正常使用待验证。

Synapticon提供的修改版IgH基于的版本是1.5.2，相比官方版本更早。

目前实现了控制代码的版本是在ubuntu16.04LTS下测试的，以Synapticon提供的IgH为核心。

对于IgH主站的配置，Synapticon提供了官方的快捷安装脚本文件、

```
$ sudo sncn_installer/install.sh eth0
```

注意这里指向的eth0是为了在脚本中实现对主站Mac地址的获取，所以对于不同的Ubuntu版本，网口名称需要通过 **ethtool** 获得，例如在ubuntu14.04下网卡命名形式为 **eth_x** ,但是从ubuntu16.04开始命名的形式更改为 **enp_x**

在脚本中自动安装了相关的依赖库，以及对 **.conf** 文件的配置、符号链接的配置。

本文为了更清楚所执行的操作，以原生安装方式为例进行具体描述。如果只需要了解如何配置，按照上面的命令执行 **sh 文件即可。**

安装依赖库

```
$ sudo apt install autoconf automake libtool net-tools
$ sudo apt-get install linux-headers-$(uname -r)
$ sudo apt-get install build-essential cmake
```

注意到如果事前对系统打了rt补丁，可能在 **linux-header-\$(uname -r)** 的位置报错，即找不到合适的头文件，这里的问题可以略过。

编译

在解压/clone后的etherlab文件夹目录下进行命令操作

```
$ ./bootstrap
$ ./configure --enable-cycles --enable-sii-assign --enable-hrtimer --enable-8139too=no --prefix="/opt/etherlab"
```

对于 `configure` 参数, 这里提供简单参考

Table 9.1: Configuration options

Option/Switch	Description	Default
<code>--prefix</code>	Installation prefix	<code>/opt/etherlab</code>
<code>--with-linux-dir</code>	Linux kernel sources	Use running kernel
<code>--with-module-dir</code>	Subdirectory in the kernel module tree, where the EtherCAT kernel modules shall be installed.	<code>ethercat</code>
<code>--enable-generic</code>	Build the generic Ethernet driver (see section 4.3).	yes
<code>--enable-8139too</code>	Build the 8139too driver	yes
<code>--with-8139too-kernel</code>	8139too kernel	†
<code>--enable-e100</code>	Build the e100 driver	no
<code>--with-e100-kernel</code>	e100 kernel	†
<code>--enable-e1000</code>	Enable e1000 driver	no
<code>--with-e1000-kernel</code>	e1000 kernel	†
<code>--enable-e1000e</code>	Enable e1000e driver	no
<code>--with-e1000e-kernel</code>	e1000e kernel	†
<code>--enable-r8169</code>	Enable r8169 driver	no
<code>--with-r8169-kernel</code>	r8169 kernel	†
<code>--enable-ccat</code>	Enable ccat driver (independent of kernel version)	no
<code>--enable-igb</code>	Enable igb driver	no
<code>--with-igb-kernel</code>	igb kernel	†
<code>--enable-kernel</code>	Build the master kernel modules	yes
<code>--enable-rtdm</code>	Create the RTDM interface (RTAI or Xenomai directory needed, see below)	no
<code>--with-rtai-dir</code>	RTAI path (for RTAI examples and RTDM interface)	
<code>--with-xenomai-dir</code>	Xenomai path (for Xenomai examples and RTDM interface)	
<code>--with-devices</code>	Number of Ethernet devices for redundant operation (> 1 switches redundancy on)	1

Option/Switch	Description	Default
<code>--enable-debug-if</code>	Create a debug interface for each master	no
<code>--enable-debug-ring</code>	Create a debug ring to record frames	no
<code>--enable-eoe</code>	Enable EoE support	yes
<code>--enable-cycles</code>	Use CPU timestamp counter. Enable this on Intel architecture to get finer timing calculation.	no
<code>--enable-hrtimer</code>	Use high-resolution timer to let the master state machine sleep between sending frames.	no
<code>--enable-regalias</code>	Read alias address from register	no
<code>--enable-tool</code>	Build the command-line tool “ethercat” (see section 7.1)	yes
<code>--enable-userlib</code>	Build the userspace library	yes
<code>--enable-tty</code>	Build the TTY driver	no
<code>--enable-wildcards</code>	Enable <code>0xffffffff</code> to be wildcards for vendor ID and product code	no
<code>--enable-sii-assign</code>	Enable assigning SII access to the PDI layer during slave configuration	no
<code>--enable-rt-syslog</code>	Enable syslog statements in real-time context	yes

https://blog.csdn.net/qc_49530144

上述代码中应用的 `configure` 参数是Synapticon官方建议的。

`configure` 完成后进行编译

```
$ sudo make
$ sudo make modules
```

安装

```
$ sudo make modules_install install
$ sudo ldconfig
$ sudo depmod
```

检查

在 `/opt` 下查看是否存在 `etherlab` 文件夹，检查是否安装。

至此如果没有出现报错，安装结束，之后需要进行一系列配置工作保证主站的正常使用

配置

配置工作需要完成对服务的链接、二进制文件的链接以及接口的配置

Service

```
$ sudo ln -s /opt/etherlab/etc/init.d/ethercat /etc/init.d/ethercat
```

Binary

```
$ sudo ln -s /opt/etherlab/bin/ethercat /usr/bin/ethercat
```

Interface

1. 添加udev rule

```
$ echo KERNEL=="EtherCAT[0-9]*", MODE=="0664" > /etc/udev/rules.d/99-EtherCAT.rules
```

2. 链接sysconfig

在 `/etc` 下创建 `/sysconfig` 目录后

```
$ sudo ln -s /opt/etherlab/etc/sysconfig/ethercat /etc/sysconfig/ethercat
```

3. 配置conf

在 `/etc/sysconfig/ethercat` 文件中配置需要绑定的网口MAC与驱动模块

```
MASTER0_DEVICE="${MAC}"  
DEVICE_MODULES="generic"
```

保存退出后配置结束。

验证

```
~$ ethercat
```

如果正确安装，这里可以看到显示了ethercat命令的所有用法
几个比较常用的命令是：

```
$ ethercat master  
$ ethercat slave
```

```
$ ethercat pdos
```

对于主站的状态控制，通过下面的代码实现

```
$ sudo /etc/init.d/ethercat status  
$ sudo /etc/init.d/ethercat start  
$ sudo /etc/init.d/ethercat stop  
$ sudo /etc/init.d/ethercat restart
```

如果需要系统自启动 **ethercat** 服务，通过 **sysv-rc-conf** 管理
如果需要配置实时权限：

```
$ sudo vim /etc/security/limits.conf
```

在文件中添加

```
<username> hard rtprio 99
```

参考

<https://blog.csdn.net/Allforyouz/article/details/78721760> (实时权限)

<https://blog.csdn.net/scyhk/article/details/51672136> (服务自启动)

<https://blog.csdn.net/lslfox/article/details/106985609>

<https://blog.csdn.net/cln512/article/details/106728954>

https://blog.csdn.net/weixin_40293570/article/details/107610445

https://blog.csdn.net/Beliven/article/details/79661386?depth_1-