

# User guide to IPC/DEBIAN60A-v2.1 - a Debian Linux by Syslogic



## Version 2.1

Version des Dokuments	Datum	Autor	Modifikation
1.0	24.06.2013	Ivo Trajkovic	

## Content

<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1.	Abstract	1
1.2.	Sources of information	1
1.3.	Conventions	1
<b>2</b>	<b>Getting started quickly</b>	<b>2</b>
2.1.	System preparation	2
2.2.	Booting	2
2.3.	Kernel arguments and GRUB	2
2.4.	Serial port console	2
2.5.	Login, User, and password configuration	3
2.6.	Console keyboard configuration	3
2.7.	Time setup via the network time protocol (NTP)	3
2.8.	USB storage devices	3
2.9.	Overview of the installed software	3
<b>3</b>	<b>Kernel sources, manuals, sample tools</b>	<b>4</b>
<b>4</b>	<b>Virtual computer</b>	<b>4</b>
4.1.	General	4
4.2.	Installation	4
4.3.	USB performance	5
4.4.	Shared folders	5
<b>5</b>	<b>Debian Software Management</b>	<b>5</b>
5.1.	dpkg	5
5.2.	aptitude	5
5.3.	Optional software packages	6
5.3.1.	MySQL Server	6
5.3.2.	Tomcat6	6
5.4.	Language	6
<b>6</b>	<b>Network setup</b>	<b>6</b>
<b>7</b>	<b>Window manager / graphical desktop environment</b>	<b>8</b>
7.1.	Graphics drivers and 2D acceleration	8
7.2.	Graphical framebuffer console	8
7.3.	Graphical desktop – the X server	8
7.4.	Default window manager fluxbox	8
7.5.	Optional full desktop environments GNOME and KDE	9
7.6.	console-mode only	9

<b>8</b>	<b>Serial port configuration</b>	<b>10</b>
8.1.	Kernel configuration for more than 4 serial ports	10
8.2.	Console on a serial port	10
<b>9</b>	<b>CAN port configuration</b>	<b>11</b>
9.1.	Getting started	11
9.2.	lincan	11
9.3.	SocketCAN	11
<b>10</b>	<b>Syslogic netX boards</b>	<b>13</b>
10.1.	Firmware and configuration files	13
10.2.	Using the driver	13
<b>11</b>	<b>EtherCAT Master</b>	<b>14</b>
11.1.	General remarks	14
11.2.	Driver configuration	14
11.3.	Starting the master	14
<b>12</b>	<b>Audio</b>	<b>15</b>
12.1.	Driver configuration	15
12.2.	Installing and using ALSA	15
<b>13</b>	<b>Network filesystems</b>	<b>16</b>
13.1.	Accessing files on WINDOWS from DEBIAN-60A	16
13.2.	Accessing files on DEBIAN-60A from Windows	16
13.3.	Exchange files with SCP	17
13.4.	Remote shell with SSH	17
<b>14</b>	<b>Linux system deployment</b>	<b>18</b>
14.1.	Backup	18
14.2.	Backup of the master boot record	18
14.3.	Deploying a system image in archive format	18
	Step 1: Preparing the compact flash card	19
	Step 2: Create filesystem and initialise swap disk	19
	Step 3: Copy base image	19
	Step 4: Install bootloader	19
	Step 5: Finishing	19
<b>15</b>	<b>Making a program running faster / priorities</b>	<b>19</b>
<b>16</b>	<b>Linux kernel</b>	<b>20</b>
16.1.	Building a kernel - the Debian way	20
16.2.	Building and installing the Syslogic modules	20
	<b>Contact Information / Disclaimer</b>	<b>21</b>

# 1 Introduction

## 1.1. Abstract

This guide describes how to use the IPC/DEBIAN-60A v2.1 GNU/Linux Distribution.

The distribution has been configured and optimised by Syslog to run on the x86 industrial computers by Syslogic.

Syslogic has chosen this Linux distribution because of the quality of its releases and the easy-to-use package management system (especially APT). Moreover, Debian uses an open development and testing process. It is developed by volunteers from around the world and supported by donations through Software in the Public Interest, Inc., a non-profit umbrella organization for free software projects.

## 1.2. Sources of information

This guide focuses on the specific parts of IPC/DEBIAN-60A v2.1 for the Syslogic products; the “Debian Linux Anwenderhandbuch” <http://debiananwenderhandbuch.de/> (only German version available) covers more Debian-specific details.

Tables 1 and 2 list the Syslogic hardware supported by this Linux distribution.

Platform / architecture	Syslogic product(s)
Vortex86DX	IPC/NETIPC-4x
Vortex86DX2	IPC/BM41xxx-xxxx
AMD Geode	IPC/NETIPC-6x IPC/NETSBC-6x
Intel Atom Z510/530	IPC/NETSBC-7x
Intel Atom E6xx	IPC/BL71-xxxx

**Table 1 – Supported CPU boards**

Description	Syslogic product(s)
Relais board	IPC/REL12-1Ax
Digital IO board	IPC/DIO32-1Ax
CAN boards	IPC/COMPACGP-1Fx FBC/CANCORE-2x
NetX100 boards	IPC/NETX100-xx IPC/NETXGP-xx
4-chan serial port board	IPC/SIC4(T)-1xx
Board for wireless communication: GPS, UMTS, WLAN, Bluetooth	IPC/WIRELESS

**Table 2 – Supported expansion boards**

## 1.3. Conventions

In this document, the console input on the industrial computer is represented by

```
netipc:/#
```

The console input on the development system (usually a commercial PC) is represented by

```
host:/#
```

Device drivers included in the Linux kernel are called “modules” or “kernel modules”.

## 2 Getting started quickly

### 2.1. System preparation

If the DEBIAN-60A compact flash card has been delivered to you separately, open your industrial computer and plug the flash card into the corresponding socket on the CPU board. Connect a keyboard, a VGA monitor, and a mouse (optional).

### 2.2. Booting

The Linux loader GRUB starts after the Bios screen. GRUB will list the installed kernel versions in a blue menu, one per line. After a couple of seconds of keyboard inactivity, the first / default kernel in the list will start.

### 2.3. Kernel arguments and GRUB

Kernel arguments enable / disable kernel features. The default kernel arguments should do for most configurations; to add arguments, hit “e” in the GRUB menu and navigate the cursor with the arrow keys to the end of the line “*linux /boot/vmlinuz...*” where a kernel argument is entered. Some useful kernel arguments are listed in Table 3.

Kernel argument	Description
8250.nr_uarts=x	x is the number of UARTs (if more than 4), see Section 8
console=x	Put system console on serial port, e.g.: console=ttyS0,57600,8,n,1

**Table 3 – Some useful kernel arguments**

Kernel arguments are active per default when added to the file */etc/default/grub*:

```
netipc:/# nano /etc/default/grub
```

Add your kernel argument to the line

```
GRUB_CMDLINE_LINUX="new kernel argument"
```

Now invoke GRUB’s configuration update with

```
netipc:/# update-grub
```

### 2.4. Serial port console

DEBIAN-60A v2.1 exports by default a console on the first serial port, i.e. */dev/ttyS0*, with symbol rate 115200 baud, 8 data bits, 1 stop bit, no parity, no flow control.

To login from a remote Windows computer to the computer running DEBIAN-60A v2.1, (i) connect the latter’s serial port “COM1” over a “RX/TXcrossed” / “AT-Link” RS232 cable to a serial port of the Windows computer, (ii) inside Windows use e.g. the tool “putty” (can be found under */tools* on the product DVD) to open a serial session, and (iii) boot the DEBIAN-

60A computer. The DEBIAN-60A login should appear inside the putty window after the booting has finished.

**Note:**

The kernel messages can be viewed inside the serial console by using the commands “dmesg” and “cat /var/log/kern.log”.

More details about the configuration of the serial console can be found in Section 8.2.

## 2.5. Login, User, and password configuration

DEBIAN-60A v2.1 automatically logs in the user root on terminals tty1, ..., tty6. This behavior can be changed by editing */etc/inittab*.

The Syslogic Linux release is configured with the following main accounts:

Username	Password	Description
netipc	netipc	Standard / normal user account
root	netipc	Root / administrator account

**Table 4 – Preconfigured user accounts**

The password of the actually logged in user is changed with

```
netipc:/# passwd
```

## 2.6. Console keyboard configuration

To change the keyboard layout,

```
netipc:/# dpkg-reconfigure keyboard-configuration
```

## 2.7. Time setup via the network time protocol (NTP)

In DEBIAN-60A v2.1, the system clock and the hardware clock are configured automatically via NTP on boot, if the system is connected to the internet. This is done via the “ifup”-network script */etc/network/if-up.d/ntpdate*.

## 2.8. USB storage devices

USB storage devices are mounted under */media/usb\** automatically after being attached.

All users are able to read from and write to the mounted device.

All users can and should unmount a USB storage device after reading from or writing to it with

```
netipc:/# pumount /dev/sdb1
```

It might be necessary to adapt “/dev/sdb1” (the console command “dmesg” may be helpful).

## 2.9. Overview of the installed software

The Debian Linux image on the Compact Flash comprises:

- the Linux kernel “3.6.11-rt25-syslogic-486” with embedded lincan 0.3.5, EtherCAT 1.5.1, Intel EMGD (support for Atom E6xx-based) modules and modules for Syslogic on-board peripherals and add-on boards

- Low-latency Linux kernel “2.6.38.8-syslogic-lowlatency-486” with embedded lincan 0.3.5 and EtherCAT 1.5.1 modules and modules for Syslogic on-board peripherals and add-on boards
- Debian’s generic / universal kernel “2.6.32-5-486”
- Binutils and GNU Compiler Tools
- DHCP client
- “Fluxbox” graphical window manager
- Textual file manager “Midnight Commander (mc)”
- Iceweasel (Firefox) Web-Browser
- Network servers: Samba (Windows file sharing) / Apache (WWW) / FTP / Telnet / SSH / Cups (Network Printing)
- A tool for automatic mounting of USB storage devices: usbmount
- NTP (Network Time Protocol) client “ntpdate”
- Java™ SE runtime environment – build 1.7.0\_21-b11 (installed under */usr/local/jre1.7.0\_21/*)

### 3 Kernel sources, manuals, sample tools

On the DEBIAN-60A v2.1 root partition, various sample tools for add-on boards and features by Syslog are under */SYSLOGIC/tests\_demos*, manuals for the kernel modules supporting the add-on boards are on the product DVD under */doc*, kernel sources and installable binary packages are on the product DVD under */sources*.

## 4 Virtual computer

### 4.1. General

DEBIAN-60A v2.1 may serve as a development environment on the industrial computer. Nevertheless, an additional PC running DEBIAN-60A v2.1 or another Linux, natively or virtually, may be handy for testing purposes or for compiling the kernel sources.

### 4.2. Installation

To run the virtual computer, download and install the freely available VirtualBox (<https://www.virtualbox.org/>).

Next, copy the “*DEBIAN60\_v2.1.vdi*” file from the “*/virtualbox*” directory on the product DVD to your PC (requires ~1.5 GB of free HDD space); the write permission on the file must be re-enabled afterwards. After starting the VirtualBox, click on the “new”-button on the left top to create a new virtual computer. Choose the “*DEBIAN60\_v2.1.vdi*” file as a virtual SATA disk. Now launch the virtual computer by clicking on the “*Show*” button.

**Note:**

You must click into the virtual computer window in order to use the mouse inside it. To free the trapped mouse pointer, press the right **Ctrl** button. To avoid the mouse trapping, install the so called “VirtualBox guest additions” on the virtual computer.

### 4.3. USB performance

If you encounter poor USB performance inside the virtual DEBIAN60A,

1. install the “Virtualbox Extension Pack” from <https://www.virtualbox.org/>,
2. in the Virtualbox window, click through “File → Preferences → Extensions“ and finally press the „add package“ button ,
3. choose the extension package you just downloaded.

This procedure will enable virtual USB 2.0 ports.

### 4.4. Shared folders

Inside the virtual DEBIAN60A, “shared folders” enable you to mount folders from the computer hosting the virtual computer. To use this feature, install the “VirtualBox guest additions” from <https://www.virtualbox.org/>. Finally, add the shared folders inside the “Shared Folders” section in the Virtualbox window of your virtual computer.

## 5 Debian Software Management

DEBIAN-60A’s basic packet manager is “dpkg”, whereas “aptitude” works on top of dpkg and resolves package dependencies.

### 5.1. dpkg

<code>dpkg -l</code>	List installed packages
<code>dpkg -l "*"</code>	List all (available and installed packages)

The resulting package list shows the version number and the installation status of the packages:

```
rc hotplug          0.0.20040329-26 Linux Hotplug Scripts
ii iamerican        3.1.20.0-4.3   An American English dictionary for ispell
...
```

The first row is the paket action:

*u=unknown, i=install, r=remove (delete without config), p=purge (delete with config)*

The second row is the current paket status:

*n=not installed, i=installed, c=not installed anymore (but configfiles ok), u=unpacked, but unconfigured, f=failed configuration, h=half installed*

<code>dpkg -i packet.deb</code>	Install packet.deb
<code>dpkg -r packet.deb</code>	Remove packet.deb, keep configuration files
<code>dpkg -P packet.deb</code>	Remove packet.deb, also delete configuration files
<code>dpkg-reconfigure packet.deb</code>	Reconfigure packet.deb
<code>dpkg -S 'filename'</code>	Search for package that contains 'filename'
<code>dpkg -s packet</code>	Show packet status
<code>dpkg -L packet</code>	Show files of a packet

### 5.2. aptitude

Aptitude can be used to install or remove single packages or package groups.

<code>aptitude</code>	Runs the text menu based aptitude front end
<code>aptitude search keyword</code>	Search in all available packages for a keyword
<code>aptitude show package</code>	Show all package infos



<code>aptitude update</code>	Update the package list
<code>aptitude install package</code>	Install a package
<code>aptitude remove package</code>	Remove a package (without config. files)
<code>aptitude purge package</code>	Remove a package (with all config. files)
<code>aptitude clean</code>	Clears the package cache (/var/cache/apt/archives)
<code>aptitude autoclean</code>	Clears old packets from the package cache

The following keys are useful inside the text menu based aptitude front.

<code>u</code>	update the package lists
<code>/</code>	search for packages
<code>n</code>	continues searching
<code>+</code>	install / update the selected package
<code>-</code>	remove the selected package
<code>q</code>	quit

### 5.3. Optional software packages

Use internet search machines or aptitude to look for specific software. The installation of some popular packages are described below in this section.

#### 5.3.1. MySQL Server

Install latest version of MySQL Server for Debian (approx. 120MB):

```
netipc:/# aptitude install mysql-server
```

#### 5.3.2. Tomcat6

Install version 6 of Tomcat for Debian (approx. 100MB):

```
netipc:/# aptitude install tomcat6
```

### 5.4. Language

To change the language of DEBIAN-60A, use

```
netipc:/# dpkg-reconfigure locales
```

## 6 Network setup

In DEBIAN-60A v2.1, all ethernet interfaces are preconfigured for automatic IP address assignment based on DHCP. Changes to this configuration, for example assigning static IP addresses, are made through the file `/etc/network/interfaces`.

#### Example: /etc/network/interfaces

```
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

# The loopback network interface
auto lo
iface lo inet loopback

auto eth0 eth1

#####
# automatic network configuration (DHCP)
# uncomment to enable, comment out to disable
#####

iface eth0 inet dhcp
iface eth1 inet dhcp

#####
# static network configuration
# uncomment to enable, comment out to disable
#####

#iface eth0 inet static
#   address    192.168.1.40
#   netmask    255.255.255.0
#   network    192.168.1.0
#   broadcast  192.168.1.255
#   gateway    192.168.1.1

#iface eth1 inet static
#   address    192.168.2.41
#   netmask    255.255.255.0
#   network    192.168.2.0
#   broadcast  192.168.2.255
#   gateway    192.168.2.1
```

After being connected to a network, an “ifdown” then an „ifup“ command is ran automatically configuring the particular ethernet port. On disconnecting, an “ifdown” command configures the port for no IP. These mechanisms are implemented by the “ifplugd” daemon.

## 7 Window manager / graphical desktop environment

### 7.1. Graphics drivers and 2D acceleration

Through the script `/etc/syslogic/init_scripts/syslogic` (out of `/etc/rc.local`), the modules for accelerated graphics suitable for your Syslogic computer are loaded on boot.

### 7.2. Graphical framebuffer console

You don't need this section if you intend to only run a X server based graphical desktop.

To enable the graphical framebuffer console, first load "fbcon" module with

```
netipc:/# modprobe fbcon
```

Use the following table to find out the name of the appropriate framebuffer module:

Platform / architecture	Framebuffer module
Vortex86DX based	sm7xx
AMD Geode based	lxfb
Intel Atom Z5xx based	gma500_gfx
Vortex86DX2 based	currently not supported
Intel Atom E6xx based	currently not supported

**Table 5 - Syslogic hardware platforms and the appropriate framebuffer modules**

The framebuffer module can now be loaded with modprobe, e.g.

```
netipc:/# modprobe gma500_gfx
```

The resolution of the framebuffer console can be modified with the "fbset" command, e.g.

```
netipc:/# fbset -a -g 1024 768 1024 768 16
```

The option "-a" invokes change in all consoles; the first two values given with the option "-g" represent the visible resolution, the second two values represent the virtual resolution, 16 is the number of bits a pixel colour value is represented by.

#### Note:

Especially "sm7xx" and "gma500\_gfx" are in development stage; therefore not all resolutions and colour depths are supported, and you might get a black screen after using the "fbset" command. Also running the X server from a graphical console might produce a black screen. In this case, the easiest is to press the CTRL+ALT+F1, then the CTRL+ALT+DEL key combinations which will reboot the system.

### 7.3. Graphical desktop – the X server

Start the graphical environment, also "X server" with

```
netipc:/# startx
```

### 7.4. Default window manager fluxbox

The window manager works on the top of the X server and implements features for graphical operation similar to Windows or OSX.

As an embedded computer manufacturer, we have pre-installed the light-weight window manager “fluxbox”.

### **7.5. Optional full desktop environments GNOME and KDE**

To install a full desktop environment, simply

```
netipc:/# aptitude install gnome
```

or

```
netipc:/# aptitude install kde
```

To uninstall any of the desktop environments, just re-run aptitude with the option “remove” instead of “install”.

### **7.6. console-mode only**

To get rid of the window manager and X server altogether, remove the packages with:

```
netipc:/# aptitude remove fluxbox  
netipc:/# aptitude remove xserver-xorg
```

## 8 Serial port configuration

The Kernels of DEBIAN-60A v2.1 automatically detect and configure the serial ports. This can be overruled by the “setserial” tool through the /etc/serial.conf file.

### Note:

When serial ports share interrupt lines, communication may hang. To solve this issue, in the BIOS setup, configure the affected serial ports to use an interrupt line which is not used by any other device (check this in DEBIAN-60A with the command “`cat /proc/interrupts`”)

### 8.1. Kernel configuration for more than 4 serial ports

If your Syslogic computer features more than 4 serial ports, the kernel argument “`8250.nr_uaarts=x`” must be active (see Section 2.3), where *x* states the total number of ports.

### Note:

The “serial.pdf” document (under /doc on the product DVD) details how to test and get operational with the serial ports.

### 8.2. Console on a serial port

In DEBIAN-60A v2.1, a console on the serial port is preconfigured through the following line in the /etc/inittab file:

```
T0:23:respawn:/sbin/getty -L ttyS0 115200 vt100
```

This provides system access through COM1 with a terminal program from a remote computer at a rate of 115200 baud.

## 9 CAN port configuration

### 9.1. Getting started

Some Syslogic CPU boards are equipped with up to 2 Philips/NXP SJA1000 CAN controller which are supported by “lincan” and “SocketCAN”. The “3.6.11-rt25-syslogic-486” and “2.6.38.8-syslogic-lowlatency-486” kernels both contain both drivers. The lincan driver has been ported from the project “lincan” 0.3.5 and has been modified by Syslogic to improve the initialisation of the SJA1000 controller; the original SocketCAN drivers, as part of the kernel tree, have stayed untouched.

#### Note:

Before loading any CAN modules, make sure that:

1. you have chosen one IRQ number for both CAN controllers, e.g. 11
2. you have configured the hardware to use this IRQ according to the user manual of your Syslog industrial computer
3. you have reserved the chosen IRQ in the BIOS settings somewhere under “PnP / PCI Configurations → IRQ Resources” or similar, which means that the BIOS will not automatically assign this IRQ to any devices
4. the chosen IRQ number is not listed by “cat /proc/interrupts” under Linux
5. the chosen interrupt line is level-triggered (done with “cd /SYSLOGIC/tests\_demos/lincan; ./prog\_irq x level” under Linux, where *x* is your chosen IRQ number, e.g. 11)
6. the CAN bus is terminated correctly

### 9.2. lincan

Assuming that both CAN controllers are configured for IRQ 11, the lincan module can be loaded with

```
netipc:/# modprobe lincan hw=gensja1000io,gensja1000io irq=11,11  
io=0x7600,0x7700
```

or the line

```
lincan hw=gensja1000io,gensja1000io irq=11,11 io=0x7600,0x7700
```

can be added to the file “/etc/modules”.

The PeliCAN mode is activated per default. The lincan utilities “rxtx”, “sendburst”, “readburst”, “send”, and “can-proxy” are under “/SYSLOGIC/tests\_demos/lincan”.

Further details on the setup of the CAN hardware and the lincan driver can be found on the product DVD under */doc/lincan.pdf*.

### 9.3. SocketCAN

Assuming that both CAN controllers are configured for IRQ 11, the SocketCAN module can be loaded with

```
netipc:/# modprobe sja1000_isa port=0x7600,0x7700 irq=11,11
```

The bitrates of the SocketCAN interfaces “can0” and “can1” are set to e.g. 500 kbit/s by

```
netipc:/# ip link set can0 type can bitrate 500000 restart-ms 1000
```

```
netipc:/# ip link set can1 type can bitrate 500000 restart-ms 1000
```

#### Activate the interfaces with

```
netipc:/# ifconfig can0 up
netipc:/# ifconfig can1 up
```

Note that, the above commands can also be added to */etc/rc.local* to be automatically executed on startup, e.g.

```
#!/bin/sh -e
#
# rc.local
#
# This script is executed at the end of each multiuser runlevel.
# Make sure that the script will "exit 0" on success or any other
# value on error.
#
# In order to enable or disable this script just change the execution
# bits.
#
# By default this script does nothing.

/etc/syslogic/init_scripts/syslogic

# Program IRQ 11 to be level-triggered
cd /opt/SocketCAN
prog_irq 11 level

# Load SocketCAN module
modprobe sjal1000 isa port=0x7600,0x7700 irq=11,11
ip link set can0 type can bitrate 500000 restart-ms 1000
ip link set can1 type can bitrate 500000 restart-ms 1000
ifconfig can0 up
ifconfig can1 up

exit 0
```

A sample application for SocketCAN in C is under */SYSLOGIC/tests\_demos/socket\_can*. To compile it, change to that directory and run

```
netipc:/SYSLOGIC/tests_demos/socket_can# make
```

## 10 Syslogic netX boards

### 10.1. Firmware and configuration files

The netX boards require a strictly defined folder structure (located under `/opt/cifx`) where the bootloader, firmware and warmstart parameters are stored:

```
netipc:/# tree /opt/cifx
.
├── cifX0.log
├── deviceconfig
│   ├── 9059011
│   │   ├── 21
│   │   │   ├── channel0
│   │   │   └── device.conf
│   │   └── 28
│   │       ├── channel0
│   │       └── device.conf
├── nxgp_bsl.bin
└── README
```

The top-level directory `/opt/cifx` contains the bootloader `nxgp_bsl.bin`. Below that, the device configurations are stored, each in a separate folder structure, composed of the board function number (e.g. 9059011) and the serial number (e.g. 21). Under `channel0`, the firmware for the protocol stack is kept. The example shown above is contained on your DEBIAN-60A image under `/opt/cifx`, and must be individually adopted for your specific board. Several device definitions can co-exist. There is no need to delete device definitions for cards which are not actually present in the system.

### 10.2. Using the driver

A sample application “netxlinuxsample” under `/SYSLOGIC/tests_demos/netx` (for netX devices) detects a card and initialises it. Before running `netxlinuxsample`, please configure the I/O base address of the netx submodule on the IPC/COMPACGP card according to the latter’s documentation.

Then adapt the

```
#define ISA_DPM_ADDR 0xD8000
```

line inside the `netxlinuxsample.c` file by replacing “0xD8000” by the correct I/O base address.

Now, compile and run the application:

```
netipc:/SYSLOGIC/tests_demos/socket_can# make
netipc:/SYSLOGIC/tests_demos/socket_can# ./netxlinuxsample
```

For further information regarding programming and usage of netx cards, please contact our support.



## 11 EtherCAT Master

### 11.1. General remarks

The IgH EtherCAT-Master for Linux allows to use a generic Ethernet controller as a EtherCAT master. The project is maintained by the Ingenieurgemeinschaft IgH and hosts the sources and documentation under **Fehler! Verweisquelle konnte nicht gefunden werden.**

### 11.2. Driver configuration

The user has to adapt the MAC address of the ethernet interface in the `/etc/ethercat` file. The MAC address can be identified using the “ifconfig” command or by the sticker attached nearby the interface port.

```
netipc:/# vim /etc/ethercat.conf
. . .
MASTER0_DEVICE="00:a0:10:00:87:fa"
. . .
```

In the same configuration file, the module can be selected. Default is “ec\_generic”.

### 11.3. Starting the master

The master can be started and stopped via the ethercat init script (see below). Check with `dmesg` which device name the ethercat master was assigned to. This is needed for the subsequent “ip” command. The Ethercat master status can be obtained with the tool `ethercat_test` under `/SYSLOGIC/tests_demos/ethercat` (“ethercat -h” generates a help screen after the Ethercat master has been started)

```
netipc:/# /etc/init.d/ethercat start
Starting EtherCAT master devel done
netipc:/# dmesg
. . .
[ 2174.47] EtherCAT: Master driver devel unknown
[ 2174.47] EtherCAT: 1 master waiting for devices.
[ 2174.48] ec_generic: EtherCAT master generic Ethernet device module devel
unknown
[ 2174.49] EtherCAT: Accepting device 00:A0:10:00:87:FA for master 0.
[ 2174.49] ec_generic: Binding socket to interface 2 (eth0).
[ 2174.49] EtherCAT 0: Starting EtherCAT-IDLE thread.
. . .
netipc:/# ip link eth0 up
```

## 12 Audio

### 12.1. Driver configuration

Certain variants of IPC/NETIPC6 and IPC/NETSBC6 are equipped with audio features. In DEBIAN-60A, audio support for those boards has to be explicitly enabled by either loading the module

```
netipc:/# modprobe snd-cs5535audio
```

or, by removing/commenting out the entry in the modprobe blacklist:

```
netipc:/# vim /etc/modprobe.d/blacklist
. . .
# blacklist snd-cs5535audio
```

### 12.2. Installing and using ALSA

Further, you will need to install basic ALSA tools with:

```
netipc:/# apt-get install alsa-base alsa-utils
```

Now you may use tools like alsamixer:

```
netipc:/# alsamixer
```

## 13 Network filesystems

### 13.1. Accessing files on WINDOWS from DEBIAN-60A

To e.g. copy files from a Windows computer into the DEBIAN-60A filesystem, setup a “shared directory” inside Windows by right-clicking on the folder and choosing “Properties”, then the “Sharing” tab.

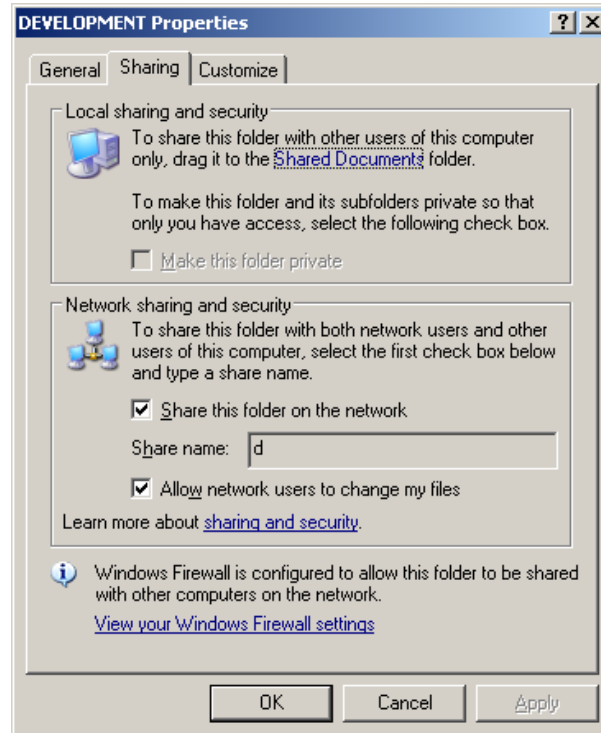


Figure 1 – Setting up a Windows Shared folder

Inside DEBIAN-60A, you can bind the share to the directory `/mnt/winhost` with

```
netipc:/# mount -t cifs //192.168.1.153/d /mnt/winhost -o  
username=Administrator,password=
```

Note that, 192.168.1.153 has to be replaced by the IP address of the Windows computer.

### 13.2. Accessing files on DEBIAN-60A from Windows

The so-called “Samba” server is pre-installed and pre-configured on DEBIAN-60A. It enables full access to DEBIAN-60A’s file system from a Windows computer. To do this, open the link [\\netipc\\syslogic\\_root](netipc://syslogic_root) inside a Windows explorer, type the username „root“ and the password „netipc“. Instead [\\netipc](netipc://), you can also use IP addresses, e.g. [\\192.168.1.214\\syslogic\\_root](192.168.1.214://syslogic_root).

The Samba server is configured through the `/etc/samba/smb.conf` file.

### 13.3. Exchange files with SCP

To copy a file from DEBIAN-60A to another Linux system with IP address e.g.192.168.1.228 and user root,

```
netipc:/# scp /filename root@192.168.1.228:/filename
```

### 13.4. Remote shell with SSH

The secure shell (SSH) is a network protocol that allows encrypted data exchange. SSH is typically used for executing shell commands on remote computers.

Assuming that your industrial computer has the IP 192.168.1.228, remotely login onto it with

```
host:/# ssh root@192.168.1.228
```

and password “netipc”.

The output of a graphical application, e.g. “xcalc”, on the remote industrial computer can be forwarded to the local computer with the option “-X”:

```
host:/# ssh -X 192.168.1.228  
host:/# xcalc
```

## 14 Linux system deployment

### 14.1. Backup

To create a backup of a compact flash card, (i) start your virtual DEBIAN-60A (see chapter 4), (ii) insert the flash card into the reader, (iii) click on / activate the USB device (corresponding to the flash card reader) in the lower right corner of the VirtualBox window to bind it as e.g. `/dev/sdb` to the virtual DEBIAN-60A and (iv) compress `/dev/sdb` into a file inside the virtual DEBIAN-60A with

```
host:/# dd if=/dev/sdb | gzip -9 > myimage.gz
```

The “-9” option means maximum compression.

As the free disk space on the flash card may have randomly assigned bits, the compression result may be poor. You can fill up the free disk space with zeros “0” with

```
netipc:/# dd if=/dev/zero of=myzerofile.bin
```

before creating a backup; this will lead to better compression ratios.

To write the image file to another flash card,

```
host:/# dd if=myimage.gz | gunzip | dd of=/dev/sdb
```

---

#### Note

This procedure will not work when trying to clone a system to a different compact flash card, as the real sizes of the cards might slightly differ from the advertised size.

---

### 14.2. Backup of the master boot record

Before applying changes to a flash card, e.g. updating the GRUB bootloader, you might want to backup the master boot record (MBR) of the card. The following command reads the MBR and writes it into the file `bootsector.bin`:

```
netipc:/# dd if=/dev/sdb of=bootsector.bin bs=1 count=446
```

To write back the MBR,

```
Netipc:/# dd if=bootsector.bin of=/dev/sdb bs=1 count=446
```

Note that, the MBR size is 512 bytes. It suffices to consider only the first 446 bytes, since the remaining bytes represent the partition table, and the latter may differ between flash cards.

### 14.3. Deploying a system image in archive format

---

#### Note:

Please double-rethink before acting, since any mistake in the commands below and above might ruin your operating system or data!

---

Syslogic provides system images in archived format (TAR compressed with bzip2). Such an image contains the complete DEBIAN-60A system including the kernels.

In the installation guide below, substitute all occurrences with the actual device node for the compact flash card reader on your system. Further, it is assumed that the first partition `/dev/sdb1` becomes the root partition. Also create an empty directory for mounting partitions, e.g. `/mnt/compactflash`.

### Step 1: Preparing the compact flash card

Partition the flash card with one of the following commands,

```
host:/# fdisk /dev/sdb
host:/# parted /dev/sdb
```

If your application consumes lots of system memory, it is recommended to create a swap partition; the latter's size can be chosen to be equal to the amount of system memory.

Follow the on-screen instructions. Choose type "Linux" (type 83h) for the root partition and **set its bootable flag**; choose type "Linux Swap" (type 82h) for the swap partition.

### Step 2: Create filesystem and initialise swap disk

Initialise (format) the system partition and initialise the swap partition with

```
host:/# mkfs.ext3 /dev/sdb1
host:/# mkswap /dev/sdb2
```

Now disable the time interval checks of the EXT3 filesystem with

```
host:/# tune2fs -i 0 /dev/sdb1
```

On Debian Linux, this can prevent confusing behaviour during boot if the hardware clock previously has lost its state.

### Step 3: Copy base image

Mount the system partition of the compact flash card with

```
host:/# mount /dev/sdb1 /mnt/compactflash
```

Mount the product DVD (assumed device node */dev/sdc*) with

```
host:/# mount /dev/sdc /media/cdrom
```

Unpack the base image */flash\_image/DEBIAN60\_v2.tar.bz2* on the product DVD with

```
host:/# tar xvf /media/cdrom/flash_image/DEBIAN60_v2.1.tar.bz2 -C
/mnt/compactflash
```

### Step 4: Install bootloader

Install the bootloader

```
host:/# grub-install --root-directory=/mnt/compactflash /dev/sdb
host:/# chroot /mnt/compactflash update-grub
```

The message "grub-probe: sending ioctl 1261 to a partition!" can be ignored.

### Step 5: Finishing

Unmount the compact flash and the product DVD

```
host:/# umount /mnt/compactflash
host:/# umount /media/cdrom
```

Now, the compact flash and the DVD can be safely removed from their drives.

## 15 Making a program running faster / priorities

To make a program use more processor cycles, i.e. increase its priority level, use

```
netipc:/# nice -n -10 myprog
```

with priority level "-10". The default priority level is 0, 19 is the lowest, -20 is the highest.

Only root may change the priority of already running programs with

```
netipc:/# renice -10 506
```

where “506” is the PID of the running program.

## 16 Linux kernel

### 16.1. Building a kernel - the Debian way

First become “root” with

```
host:/# su
```

Install the necessary packages

```
host:/# aptitude install build-essential bzip2 fakeroot initramfs-tools  
kernel-package libncurses5-dev
```

Extract the archive of the kernel source to `/usr/src/`, then configure the kernel

```
host:/# cd /usr/src/my_kernel  
host:/usr/src/my_kernel# make menuconfig
```

adapting thereby “my\_kernel” accordingly. A kernel configuration dialog will appear. Give your kernel a unique name through the “Local version – append to kernel release” item in the “General Setup” submenu.

As user “root”, start to build the kernel with

```
host:/# LOCAL_VERSION="" make-kpkg --initrd kernel_image
```

Depending on the configuration you just made, this command might take an hour or two on an industrial computer. This time can be drastically reduced by building on a (multi-core) Desktop computer, even when done inside a virtual DEBIAN-60A!

The new kernel image will appear as a Debian package, i.e. a file with ending “.deb”. This package is deployed on the industrial computer with

```
host:/# dpkg -i /path/to/the/new/kernel/package/linux-image-xyz.deb
```

where the path “/path/to/the/new/kernel/package/linux-image-xyz.deb” must be adapted accordingly.

In case the package system asks whether to stop the installation since there is a similarly called kernel already installed, answer “no”. The installer will then setup the kernel, resolve the module dependencies, build the initial ramdisk and finally update the entries in the GRUB bootloader configuration.

### 16.2. Building and installing the Syslogic modules

All modules driving Syslogic peripheral devices are integrated into the kernel tree. In the linux kernel configuration dialog, they can be enabled / disabled under “Device Drivers” → “SYSLOGIC support”. Building and installing is done implicitly when compiling the kernel. On the next boot, the modules will load according to the modutils configuration file `/etc/modules`. In this file, loading of certain modules can be enabled, disabled and configured.

## Contact Information / Disclaimer

Our distributors and system integrators will gladly give you any information about our products and their use. If you want to contact the manufacturer directly, please send a fax or email message containing a short description of your application and your request to the following address or use one of the information or technical support request forms on our internet homepage. Syslogic is grateful for any help referring to errors or suggestions for improvements.

The following registered trademarks/licences are used:

- GPL    General Public Licence
- GNU    GNU is Not Unix, Free Software Foundation

The content and presentation of this document has been carefully checked. No responsibility is accepted for any errors or omissions in the documentation. Note that this application note is constantly revised and improved. The right to change this documentation at any time without notice is therefore reserved.

Syslogic Datentechnik AG  
Täferstrasse 28  
CH-5405 Baden-Dättwil / Switzerland

Email: [support@syslogic.ch](mailto:support@syslogic.ch)  
www: <http://www.syslogic.ch>  
Tel: +41 56 200 90 40  
Fax: +41 56 200 90 50