User guide to IPC/DEBIAN-7A-32 - a 32-bit Debian Linux by Syslogic

product release 1.0

Version des Dokuments	Datum	Autor	Modifikation
1.0	10.07.2014	Ivo Trajkovic	IPC/DEBIAN-7A-32 v1.0beta
1.1	02.03.2015	Ivo Trajkovic	IPC/DEBIAN-7A-32 v1.0

Content

1 Introduction

1.1. Abstract

This guide describes how to use the IPC/DEBIAN-7A-32 GNU/Linux.

This Linux has been compiled, configured and optimised by Syslog to run on Intel Atom Exxx and DM&P Vortex86 based 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-7A-32 for the Syslogic products; the "Debian Linux Anwenderhandbuch" http://debiananwenderhandbuch.de/ (only German version available) covers more Debian-specific details.

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

Table 1 – Supported expansion boards

2 Getting started quickly

2.1. System preparation

If the IPC/DEBIAN-7A-32 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. Press the return key to launch the default kernel. During the first boot, IPC/DEBIAN-7A-32 configures itself uniquely for the hardware platform it is ran on.

Note:
On Intel Atom Exxx platforms, during the first boot GRUB is reconfigured for another default kernel; this causes an immediate reboot.

2.3. Login, User, and password configuration

IPC/DEBIAN-7A-32 automatically logs in the user root on terminals tty1, …, tty6. This behaviour can be changed by editing */etc/inittab*.

In IPC/DEBIAN-7A-32 the following main accounts are preconfigured:

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

Table 3 – Preconfigured user accounts

2.4. Console keyboard configuration

To change the keyboard layout,

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

2.5. Time setup via the network time protocol (NTP)

In IPC/DEBIAN-7A-32, 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*.

2.6. 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 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).

2.7. Overview of the installed software

The Debian Linux image on the Compact Flash comprises:

- the Linux kernel “3.14.3-rt4-syslogic-atom-svnrevX” optimised for and only runnable on Intel Atom processors; this kernel i. is RT-PREEMPT patched, ii. supports SMP with up to 4 processor cores, iii. Supports PAE for ≥4Gb of RAM, iv. comprises lincan 0.3.5, EtherCAT 1.5.1, Intel i915 (Intel Atom E38xx-based graphics), Intel EMGD (Intel Atom E6xx-based graphics) and Syslog-specific driver modules for the add-on hardware .
- the Linux kernel “3.14.3-rt4-syslogic-vortex86-svnrevX” optimised for i486; this kernel i. supports only one processor core, ii. supports only up to 1Gb of RAM, iii. is RT-PREEMPT patched, iv. comprises lincan 0.3.5, EtherCAT 1.5.1 and Syslog-specific driver modules for the add-on hardware.
- Linux kernel “3.6.11-rt25-syslogic-486” from the previous IPC/DEBIAN60A release
- Debian’s generic / universal kernel
- from scratch compiled minimal Xorg Server 1.9.0 to support the recent Intel Atom E38xx-based graphics
- Binutils and GNU Compiler Tools
- DHCP client
- “Fluxbox” graphical window manager
- “idesk” icon manager
- Textual file manager “Midnight Commander (mc)”
- Iceweasel 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 1.8.0_31-b13 (installed under */usr/local/jre/*)

2.8. Kernel sources, manuals, sample tools

On the IPC/DEBIAN-7A-32 root partition, various sample tools for add-on hardware and features by Syslogic are under */SYSLOGIC/tests_demos*.

On the IPC/DEBIAN-7A-32 medium:

- manuals for the kernel modules supporting the add-on boards are under */doc*
- kernel sources and kernel binary packages are under */kernels*
- Xorg sources are under */xorg-syslogic*.

3 Managing Debian

IPC/DEBIAN-7A-32's basic packet manager is "dpkg", whereas "aptitude" works on top of dpkg and resolves package dependencies.

3.1. dpkg

dpkg –l	List installed packages
dpkg –l "*"	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*

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

3.2. aptitude

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

aptitude	Runs the text menu based aptitude front end
aptitude search *keyword*	Search in all available packages for a keyword
aptitude show package	Show all package infos
aptitude update	Update the package list
aptitude install package	Install a package
aptitude remove package	Remove a package (without config. files)
aptitude purge package	Remove a package (with all config. files)
aptitude clean	Clears the package cache (/var/cache/apt/archives)
aptitude autoclean	Clears old packets from the package cache

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

u	update the package lists
/	search for packages
n	continues searching
+	install / update the selected package
-	remove the selected package

q	quit

3.3. Optional software packages

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

3.3.1. MySQL Server

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

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

3.3.2. Tomcat6

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

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

3.4. Language

To change the language, use

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

4 Kernel arguments and GRUB

Kernel arguments enable / disable kernel features. To add arguments, choose a kernel in the GRUB menu, hit “e”, 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 the table below.

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

Table 2 – Some useful kernel arguments

Kernel arguments are permanently active 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
```

5 Network setup

In IPC/DEBIAN-7A-32, 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 connecting a network to an RJ45 ethernet connector, the “ifplugd” daemon runs automatically the “ifdown”, then the “ifup” command, thereby automatically configuring the particular ethernet port. After unplugging, ifplugd runs “ifdown” configuring thereby the port for no IP.

6 Graphical desktop environment

Syslogic has compiled a minimal Xorg Server 1.9.0 from scratch which supports the recent Intel Atom E38xx-based graphics.

6.1. Graphics drivers

The script */usr/local/xorg-syslogic/bin/startx* loads the drivers (kernel modules) for accelerated graphics on your Syslogic computer and starts the Xorg server. Adapt this script to change Xorg parameters, e.g. screen resolution, screen rate, etc. Platform-dependent Xorg configuration files are under */usr/local/xorg-syslogic/etc/X11/*.

6.2. 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.

IPC/DEBIAN-7A-32 features the light-weight window manager “fluxbox” and the desktop icon manager “idesk”. Their configuration files are located under */root/.fluxbox* and */root/.idesktop*.

6.3. Building Xorg

To build your own Xorg Server, extract the sources from */xorg-syslogic* on the IPC/DEBIAN-7A-32 medium into your development environment. Compile everything with:

```
host:/usr/src/xorg# ./syslogic_make_install.bash all
```

This fails the very first time, if perquisites are missing. In this case, the compilation logs under *./compilation_log* are helpful.

The configuration options of Xorg-modules can be modified inside the data structure declared in the beginning of *syslogic_make_install.bash*.

After a successful compilation, the resulting binaries are copied to */usr/local/xorg-syslogic*. Another target directory can be set up inside *./syslogic_lib.bash* (variable „TARGET_DIR“). Use the *./syslogic_sync_target.bash* script to copy the resulting binaries to the target.

7 Serial port configuration

The Kernels of IPC/DEBIAN-7A-32 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, configure in the BIOS setup the affected serial ports to use an interrupt line which is not used by any other device (check this with the command “`cat /proc/interrupts`” and in the user manual of your Syslogic computer).

7.1. Kernel configuration for more than 4 serial ports

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

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

7.2. Console on a serial port

In IPC/DEBIAN-7A-32, uncommenting the following line in the */etc/inittab* file

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

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

8 CAN port configuration

8.1. Getting started

Some Syslogic computers are equipped with one or two Philips/NXP SJA1000 CAN controllers which are supported by the “lincan” and “SocketCAN” kernel drivers.

Note:

Before loading any CAN kernel drivers, make sure that:

1. your Syslogic industrial computer is configured for a certain IRQ according to the user manual
2. you have reserved the 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 another device
3. the IRQ is not listed by “cat /proc/interrupts” under Linux
4. set the IRQ to level-triggered with
   `/SYSLOGIC/tests_demos/lincan/prog_irq XX level`
   with XX = IRQ number
5. the CAN bus is terminated correctly

8.2. lincan

Assuming that both CAN controllers are configured for IRQ 15, the lincan module can be loaded with

```
netipc:/# modprobe lincan hw=gensja1000io,gensja1000io irq=15,15
io=0x7600,0x7700
```

or the line

```
lincan hw=gensja1000io,gensja1000io irq=15,15 io=0x7600,0x7700
```

can be added to the file “/etc/modules”. The PeliCAN mode is activated per default.

Use the utilities under */SYSLOGIC/tests_demos/lincan* to communicate.

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

8.3. SocketCAN

Similarly the SocketCAN kernel module can be loaded with

```
netipc:/# modprobe sja1000_isa port=0x7600,0x7700 irq=15,15
```

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 15 level
# Load SocketCAN module
modprobe sja1000_isa port=0x7600,0x7700 irq=15,15
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
```

9 EtherCAT Master

9.1. 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.

9.2. Driver configuration

First adapt the MAC address of the ethernet interface in */etc/ethercat*. 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".

9.3. Starting the master

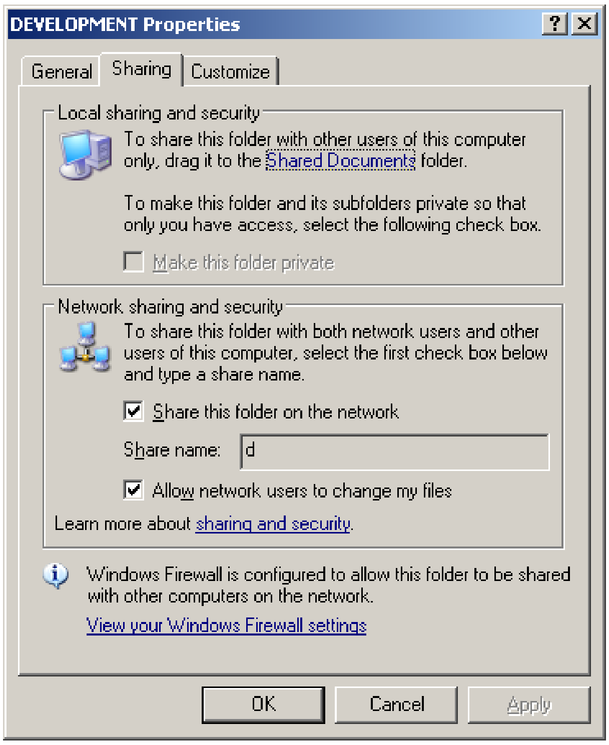
The master can be started and stopped via the ethercat init script (see below). Check with *dmesg* which device name the ethercat master was assigned to. This is needed for the subsequent "ip" command. The Ethercat master status can be obtained with the tool *ethercat_test* under */SYSLOGIC/tests_demos/ethercat* ("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
```

10 TCP/IP networking

10.1. Accessing files on WINDOWS from IPC/DEBIAN-7A-32

To e.g. copy files from a Windows computer into the IPC/DEBIAN-7A-32 filesystem, setup a “shared directory” inside Windows by right-clicking on the folder and choosing “Properties”, then the “Sharing” tab.



Local sharing and security

To share this folder with other users of this computer only, drag it to the Shared Documents folder.

To make this folder and its subfolders private so that only you have access, select the following check box.

□ Make this folder private

Network sharing and security

To share this folder with both network users and other users of this computer, select the first check box below and type a share name.

☑ Share this folder on the network

Share name: d

☑ Allow network users to change my files

Learn more about sharing and security.

Windows Firewall is configured to allow this folder to be shared with other computers on the network.

View your Windows Firewall settings

Figure 1 – Setting up a Windows Shared folder

Inside IPC/DEBIAN-7A-32, 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.

10.2. Accessing files on IPC/DEBIAN-7A-32 from Windows

The so-called “Samba” server is pre-installed and pre-configured on IPC/DEBIAN-7A-32. It enables full access to IPC/DEBIAN-7A-32’s file system from a Windows computer. To do this, open the link \\netipc\syslogic_root inside a Windows explorer, type the username „root“ and the password „netipc“. Instead \\netipc, you can also use IP addresses, e.g. \\192.168.1.214\syslogic_root.

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

10.3. Exchange files with SCP

To copy a file from IPC/DEBIAN-7A-32 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
```

10.4. 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. “iceweasel”, on the remote industrial computer can be forwarded to the local computer with the option “-X”:

```
host:/# ssh –X 192.168.1.228
host:/# iceweasel
```

11 Virtual host

A multicore desktop-PC running a (virtual) IPC/DEBIAN-7A-32 may be handy for compiling tasks.

11.1. Installation

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

Next, copy the "*DEBIAN-7A-xxx.vdi*" file from the "*/virtualbox*" directory on the product medium 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 .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.

11.2. USB performance

If you encounter poor USB performance inside the virtual IPC/DEBIAN-7A-32,

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.

11.3. Shared folders

Inside the virtual IPC/DEBIAN-7A-32, "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.

12 Linux system deployment

12.1. Backup

To create a backup of a compact flash card,

1. start your virtual IPC/DEBIAN-7A-32
2. insert the flash card into the reader
3. 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 IPC/DEBIAN-7A-32
4. compress */dev/sdb* into a file inside the virtual IPC/DEBIAN-7A-32 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 “0x00” with

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

before using “dd” above; this will lead to better compression ratios.

To write the image file to another flash card with the same or larger storage space,

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

12.2. 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*:

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

To write back the MBR,

```
host:/# 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.

12.3. 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 (compressed tar). Such an image contains the complete IPC/DEBIAN-7A-32 system.

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 runtime memory, it is recommended to create a swap partition; the latter's size can be chosen to be equal to the amount of RAM.

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 medium (assumed device node */dev/sdc*) with

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

Unpack the base image with

```
host:/# tar xvf /media/cdrom/flash_image/DEBIAN7A-32_v1.0.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 medium

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

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

13 Linux kernel

13.1. Building a kernel - the Debian way

The packages

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

must be installed.

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. Make your configuration, save it and exit the configuration dialog.

As user “root”, build the kernel with

```
host:/# make-kpkg --initrd kernel_image
```

Depending on your configuration, this might take an hour or two on an industrial computer. Building on a (multi-core) desktop computer, even when done inside a virtualised Linux, is several times faster. The new kernel image will appear as a Debian binary .deb-file. This file is deployed on the industrial computer with

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

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

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.

13.2. Building IPC/DEBIAN-7A-32’s kernel

Extract */kernels/linux-3.14.3-rt4-syslogic.tar.xz* on your IPC/DEBIAN-7A-32 medium to /usr/src/. Change to the */usr/src/linux-3.14.3-rt4-syslogic* directory.

Two configurations, one for Intel Atom Exxx and one for DM&P Vortex86, are saved in the files *config_atom* and *config_vortex86* inside */usr/src/linux-3.14.3-rt4-syslogic.*

To build both kernel configurations, run

```
host:/usr/src/linux-3.14.3-rt4-syslogic# ./syslogic_make_kernel_deb_pkg.bash
```

The two .deb binary images will appear unter */usr/src*.

13.3. Building and installing the Syslogic modules

All modules driving Syslogic peripheral devices are integrated into the Syslog kernel tree under *drivers/syslogic*. Inside the configuration dialog, they can be enabled / disabled under “Device Drivers” → “SYSLOGIC support”.

14 Real-time

All Syslogic Kernels are RT-PREEMPT patched; however, this does not simply imply that your system consisting of a Syslogic industrial computer running IPC/DEBIAN-7A-32 will meet all latency constraints in your specific application. A careful planning, measuring and testing of the whole application scenario is crucial.

The page https://rt.wiki.kernel.org/index.php/RT_PREEMPT_HOWTO describes how make use of RT-PREEMPT.

Inside IPC/DEBIAN-7A-32, the “cyclictest” tool is under */SYSLOG/tests_demos/rt-tests*.

15 Hints and notes

15.1. LVDS brightness on Intel Atom E38xx

On Intel Atom E38xx platforms, the brightness of an LVDS display is controllable through */sys/devices/pci0000:00/0000:00:02.0/backlight/acpi_video0/brightness* after the Xorg server has been started

15.2. Link up hangings on R6040 ethernet

On DM&P Vortex86 platforms, sporadic link up hangigs on the R6040 ethernet controller during Linux startup can be overcome with:

```
netipc:/# ethtool –s eth2 autoneg off; ethtool –s eth2 autoneg on
```

In IPC/DEBIAN-7A-32, this has been automised inside */etc/syslogic/init_scripts/syslogic*.

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
Täfernstrasse 28
CH-5405 Baden-Dättwil / Switzerland

Email:	support@syslogic.ch
www:	http://www.syslogic.ch
Tel:	+41 56 200 90 40
Fax:	+41 56 200 90 50