

QuickStart Instructions
phyCORE-AM3517 Rapid Development Kit for
Linux

Document No: L-765e_2
Edition: August 3, 2011

In this QuickStart are descriptions for copyrighted products that are not explicitly indicated as such. The absence of the trademark (™) and copyright (©) symbols does not imply that a product is not protected. Additionally, registered patents and trademarks are similarly not expressly indicated in this manual.

The information in this document has been carefully checked and is believed to be entirely reliable. However, PHYTEC America LLC assumes no responsibility for any inaccuracies. PHYTEC America LLC neither gives any guarantee nor accepts any liability whatsoever for consequential damages resulting from the use of this manual or its associated product. PHYTEC America LLC reserves the right to alter the information contained herein without prior notification and accepts no responsibility for any damages which might result.

Additionally, PHYTEC America LLC offers no guarantee nor accepts any liability for damages arising from the improper usage or improper installation of the hardware or software. PHYTEC America LLC further reserves the right to alter the layout and/or design of the hardware without prior notification and accepts no liability for doing so.

© Copyright 2011 PHYTEC America LLC, Bainbridge Island, WA.

Rights - including those of translation, reprint, broadcast, photomechanical or similar reproduction and storage or processing in computer systems, in whole or in part - are reserved. No reproduction may occur without the express written consent from PHYTEC America LLC.

	EUROPE	NORTH AMERICA
Address:	PHYTEC Technologie Holding AG Robert-Koch-Str. 39 D-55129 Mainz GERMANY	PHYTEC America LLC 203 Parfitt Way SW, Suite G100 Bainbridge Island, WA 98110 USA
Ordering Information:	+49 (800) 0749832 order@phytec.de	1 (800) 278-9913 sales@phytec.com
Technical Support:	+49 (6131) 9221-31 support@phytec.de	1 (800) 278-9913 support@phytec.com
Fax:	+49 (6131) 9221-33	1 (206) 780-9135
Website:	http://www.phytec.de	http://www.phytec.com

Table of Contents

Table of Contents	i
List of Figures	ii
1 Introduction	1
1.1 Rapid Development Kit Documentation	1
1.2 Professional Support Packages Available	1
1.3 System Requirements	1
1.3.1 Hardware Requirements:	1
1.3.2 Software Requirements:	1
1.4 Conventions used in this QuickStart	2
1.5 Kit Contents	3
2 Host Setup	4
2.1 Installing the phyCORE-AM3517 Kit CD	4
2.2 Server-side Support	5
2.3 TFTP	5
2.3.1 NFS	6
2.3.2 Samba	7
2.4 PTXdist	8
2.4.1 Extracting Sources	8
2.4.2 Pre-Requisites	9
2.4.3 Setup	9
2.5 Toolchains	10
2.5.1 Existing Toolchains	10
2.5.2 Building OSELAS Toolchains	10
2.5.2.1 Protecting Toolchains	11
3 Board Setup	12
3.1 Basic Settings	13
3.2 Selecting Boot Modes	14
3.3 Stand-Alone Booting	15
3.4 Remote Booting	15
3.5 Other Booting Options	15
3.6 Display Settings	16
4 Building a BSP	17
4.1 Managing Configurations	17
4.1.1 Platform	17
4.1.1.1 Enabling Graphics SDK	18
4.2 Kernel	18
4.2.1 Root Filesystem	18
4.3 Building Images with PTXdist	18
5 Flashing Images	19
5.1 X-Loader	19
5.2 U-Boot	19
5.2.1 Restore default U-Boot settings	19
5.3 Kernel	20
5.4 Root Filesystem	20
Appendix A: Helpful Hints and Tips	21
A.1 Where to Find More Information	21

List of Figures

List of Figures	ii
Fig. 2-1. File Directory	4
Fig. 3-1. NAND Switch Setting	14
Fig. 3-2. MMC Switch Setting	14

1 Introduction

This document describes how to install and work with the Linux Board Support Package (BSP) for the phyCORE-AM3517 platform. This BSP provides a fundamental software platform for development, deployment and execution on the phyCORE-AM3517.

The QuickStart contains instructions for:

- Host Setup
- Boot Options
- Building a BSP (Platform, Kernel, Root Filesystem)
- Flashing images to NAND (X-Loader, U-Boot, Kernel, Root Filesystem)

Please refer to the phyCORE-AM3517 Hardware Manual for specific information on board-level features such as jumper configuration, memory mapping and pin layout for the phyCORE-AM3517 System on Module and Carrier Board.

1.1 Rapid Development Kit Documentation

This rapid development kit (RDK) includes the following electronic documentation on the enclosed phyCORE-AM3517 Rapid Development Kit CD:

- The PHYTEC phyCORE®-AM3517 Hardware Manual
- Controller User's Manuals and Datasheets
- This QuickStart Instruction guide

1.2 Professional Support Packages Available

PHYTEC backs up our Rapid Development Kits with a Start-Up Guarantee. We invite you to make use of our free Technical Support concerning installation and setup of QuickStart demos until any kit start-up problem you might encounter is resolved. For more in-depth questions, we offer technical support packages for purchase. Please contact our sales team at sales@phytec.com to discuss the appropriate support option if professional support beyond installation and setup is required.

1.3 System Requirements

The following system requirements are necessary to complete this QuickStart without errors and problems. Deviations from these requirements may suffice, or may have other work-arounds.

1.3.1 Hardware Requirements:

The following hardware components are included in the phyCORE-AM4317 Linux Rapid Development Kit (part number KPCM-048-L) and are necessary for completing the instructions in this QuickStart

- phyCORE-AM3517 System on Module (Part No. PCM-048-5533ER110-1A)
- phyCORE-AM3517 Carrier Board (Part# PCM-961)
- AC adapter supplying 5 VDC /3.2A adapter, center positive
- Straight Ethernet cable
- Serial cable (RS-232)
- USB Standard A to mini-B cable

1.3.2 Software Requirements:

- Ubuntu 10.04 LTS

1.4 Conventions used in this QuickStart

The following is a list of the typographical conventions used in this book.

Bold	Commands or other text that should be typed literally by the user
<i>Bold Italic</i>	File and directory names
<i>Italic</i>	Field and window names or titles, menu items, and other terms that correspond to items on the PC desktop
<u>Underline Blue</u>	Hyperlinks to documents, webpage URLs, or FTPs
<bracket red>	Replace these entries with the text applicable to your system. For instance this may be a directory name, or IP address. The entire entry, including brackets should be replaced by the applicable text.
Fixed Width	Source code listings and examples

Pay special attention to the notes set apart from the text with the following icons:



SUCCESS

Completion of an important part of this QuickStart.



TIP

Useful supplementary information and tips about the topic.



TIME

Estimated completion time for the following chapter.



CAUTION

This warning will help you avoid potential problems.



RESOLVE

Helpful information for troubleshooting and resolving potential problems.

1.5 Kit Contents

The following hardware components are included in the phyCORE-AM4317 Linux Rapid Development Kit (part number KPCM-048-L) and are necessary for completing the instructions in this QuickStart:

- phyCORE-AM3517 System on Module (Part # PCM-048-5533ER110-1A)
- phyCORE-AM3517 Carrier Board (Part # PCM-961)
- Bare PCB Expansion Board (PCM-988) (optional)
- AC adapter supplying 5 VDC /3.2A adapter, center positive
- Straight Ethernet cable
- Serial cable (RS-232)
- USB Standard A to mini-B cable
- SOM extraction tool
- PHYTEC Kit CD

2 Host Setup

The phyCORE-AM3517 (PCM-048) has been developed and tested with Ubuntu 10.04 LTS Lucid Lynx. Although it is possible to use a different OS, some setup information will contain OS-specific commands and paths for settings.

Update repositories and upgrade installed packages.

```
sudo apt-get update
sudo apt-get upgrade
```

2.1 Installing the phyCORE-AM3517 Kit CD

The phyCORE-AM3517 Linux Kit CD includes datasheets, the phyCORE-AM3517 hardware manual and quickstart, PTXdist guides in addition to Linux BSP images, source, and toolchain.



Linux BSP contents can also be downloaded from the PHYTEC America FTP: ftp://ftp.phytec.com/products/PCM-048_phyCORE-AM35xx/Linux/PD11.1.0/

Installation process:

1. Insert CD into disc drive.
2. Navigate to /media/SO-492-## directory (refer to the kit CD version and match "##" accordingly).
3. Execute install.sh:
 - If using the terminal, type the following command: `./install.sh` then enter the root password.
 - If using the file browser, double-click and select Run in Terminal and enter the root password.
4. Follow the installer prompts, select a directory, and continue the installation through completion.
5. After a successful installation, the complete CD contents will be in the designated directory.

Use the following directory structure to navigate the contents of the Kit CD or the installed contents:

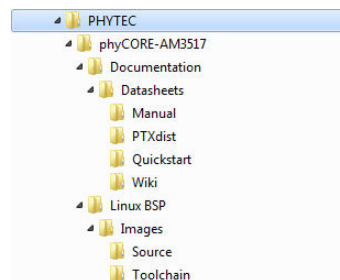


Fig. 2-1. File Directory

2.2 Server-side Support

Support for installing and setting up TFTP, NFS, and Samba server settings to enable communication between multiple systems and the target.

2.3 TFTP

TFTP allows files to be downloaded from one machine to another. With most embedded Linux devices, TFTP is an efficient way to boot the kernel during development so that the user does not have to flash a new kernel every time it is modified. It is also helpful when updating images in flash from U-Boot. First, start by installing the TFTP server.

```
sudo apt-get install tftpd-hpa
```

Next, files can be accessed from another machine on the same network by simply using the IP address of the host. You must specify a folder where the files will reside on the host by replacing the folder path for TFTP_DIRECTORY with whatever folder you wish to use as your TFTP file storage location, or leave the folder as the default.

```
sudo gedit /etc/default/tftpd-hpa
```

```
# /etc/default/tftpd-hpa
```

```
TFTP_USERNAME="tftp"  
TFTP_DIRECTORY="/var/lib/tftpboot"  
TFTP_ADDRESS="0.0.0.0:69"  
TFTP_OPTIONS="--secure"
```

If you made any changes to the settings of the TFTP server, you need to restart it for them to take effect.

```
sudo restart tftpd-hpa
```

Lastly, if you would like to grant every user on the system permission to place files in the TFTP directory, use the following command, replacing **<TFTP_DIRECTORY>** with your chosen location.

```
sudo chmod ugo+rwx <TFTP_DIRECTORY>
```

2.3.1 NFS

A network file-system (NFS) server gives other systems the ability to mount a file-system stored on the host and exported over the network. In embedded development, this is essential for quickly testing applications and evaluating different file-system setups.

To begin the installation process use the following command:

```
sudo apt-get install nfs-kernel-server
```

Exported filesystems are designated in the `/etc/exports` file and allow you to choose both the directory to be exported and many settings for accessing the exports. Below is an example for exporting a folder called `nfs_export-ex` located in a user's home directory.

```
sudo gedit /etc/exports
```

```
# /etc/exports
```

```
/home/<user>/nfs_export-ex *(rw, sync, no_root_squash, no_subtree_check)
```

The options (`rw`, `sync`, `no_root_squash`, `no_subtree_check`) for this folder are essential in setting up the NFS export correctly. For more information on additional options, refer to the man page for 'exports'.

- `rw` enables
Read and write access when the directory is mounted
- `sync`
Tells the file-system to handle local access calls before remote access
- `no_root_squash`
Allows root access when mounting the file-system
- `no_subtree_check`
Reduces the number of checks the server must make to ensure that an exported sub-directory is within an exported tree and also enables access to root files in conjunction with `no_root_squash`

After modifying this file, in order to mount the directories as an NFS, you must force the NFS server to export all of the directories listed in `/etc/exports`.

```
sudo /usr/sbin/exportfs -va
```

2.3.2 Samba

Samba servers are an excellent way to access a Linux file-system on a Windows machine via a network connection. Using a Samba server, it is quick and easy to transfer files between systems. To install a Samba server, use the following command:

```
sudo apt-get install samba
```

Before the Samba share can be mounted on another machine it's necessary to modify the configuration file to allow write access and access to home directories. Start by editing the `/etc/samba/smb.conf` file.

```
sudo gedit /etc/samba/smb.conf
```

Inside this file there are four specific things that need to be un-commented (remove the '#' at the beginning of the line) to enable the sharing of home folders and write access. Below is the section that must be modified:

```
##### Share Definitions #####

# Un-comment the following (and tweak the other settings below to suit)
# to enable the default home directory shares. This will share each
# user's home directory as \\server\username
;[homes]
;  comment = Home Directories
;  browseable = yes

# By default, the home directories are exported read-only. Change the
# next parameter to 'no' if you want to be able to write to them.
;  read only = no
```

The outcomes after the changes are made follow:

```
##### Share Definitions #####

# Un-comment the following (and tweak the other settings below to suit)
# to enable the default home directory shares. This will share each
# user's home directory as \\server\username
[homes]
  comment = Home Directories
  browseable = yes

# By default, the home directories are exported read-only. Change the
# next parameter to 'no' if you want to be able to write to them.
  read only = no
```



TIP

NOTE: It might also be necessary to change the "workgroup = " line to match the workgroup for your machine.

To apply the changes, the next step is to restart all Samba-related processes.

```
sudo restart smbd
sudo restart nmbd
```

Lastly, each user needs to have a password enabled to be able to use the Samba server. There are no rules for this password. The simplest method for choosing this password is to make it the same as the UNIX user's password, but it is not a requirement. After typing in the command below, you will be prompted to enter the password for the specified user.

```
sudo smbpasswd -a <user>
```

As mentioned in the configuration file, the samba share can be connected by accessing `\\<host machine ip>\\<user>` by either mounting a network share or using Windows explorer to navigate to it.

2.4 PTXdist

PTXdist is a set of tools created by Pengutronix to help manage all of the BSP sources, from x-loader to the filesystem and its applications. It makes it easier for a user to manage everything for specific platforms and toolchains without manually repeating relatively complex commands every time it is necessary to build a new image.

2.4.1 Extracting Sources

Visit the AM3517's BSP page phyCORE-AM3517 Linux BSP-PD11.1.0 http://www.phytec.com/wiki/index.php?title=PhyCORE-AM3517_Linux_BSP-PD11.1.0 for the correct BSP tools and images.

To install PTXdist you need to extract the archive with the PTXdist software *ptxdist-2010.11.1.tgz*.

The PTXdist packet has to be extracted in order to be built before the installation. In this example, we assume that the *ptxdist-2010.11.1.tgz* file has been placed in the `~/(<home/><user>)` directory.

```
~$ tar -zxvf ptxdist-2010.11.1.tgz
~$ cd ptxdist-2010.11.1
```

Now that the source has been extracted, the next step is to configure it for building.

2.4.2 Pre-Requisites

PTXdist checks for specific packages that must be installed before it can be successfully built. The following command begins a script that uses GNU autotools to help set up the environment for building the distribution.

```
~/ptxdist-2010.11.1$ ./configure
```

This command will automatically stop if it is missing a package, give a reason for the termination, and list which package to install to continue with the initial setup. After successfully running the configure script, PTXdist is ready to be built and installed.

```
~/ptxdist-2010.11.1$ make
~/ptxdist-2010.11.1$ sudo make install
```

The install location is **/usr/local** by default, which is why the **make install** command must be run with root privileges. If another directory is preferred for the install, use the `--prefix` option when installing, but be sure to add the **bin/** directory of the installed tools in this new folder to your `$PATH` by modifying and sourcing `~/.bashrc`.

```
source ~/.bashrc
```

Now that the install is complete, the PTXdist folder in the `~ (/home/<user>)` directory can be removed, as well as the original archive, but this is not necessary.

Optional:

```
~/ptxdist-2010.11.1$ cd ..
~$ rm -rf ptxdist-2010.11.1*
```

2.4.3 Setup

The first time PTXdist is used, there are some setup properties that have to be configured. To run PTXdist's setup, use the following command:

```
~$ ptxdist setup
```

Once in the ptxdist setup, the only settings that should be modified are the *User->Name* and *User->eMail*. This is mainly for general logging purposes. If you wish to modify any other settings, please refer to the Getting Help section for a link to PTXdist documentation.

Since PTXdist works with sources only, it needs to grab source archives from the web using `wget` as it advances through its setup if they do not exist already.

2.5 Toolchains

In order to build images or applications for an embedded device, it is necessary to have a cross toolchain that will perform the necessary operations to compile code for a specified processor and system setup.

Each toolchain will have a modified GNU Compiler Collection (gcc) designed for the desired setup.

For example, the following are all toolchains contained in the OSELAS Toolchain-1.99.3.7 distribution:

- arm-cortexa8-linux-gnueabi-gcc
- arm-1136jfs-linux-gnueabi-gcc
- arm-v5te-linux-gnueabi-gcc
- powerpc-603e-linux-gnu-gcc

2.5.1 Existing Toolchains

If a working toolchain is already installed for a given architecture, it is possible to use this instead of building an OSELAS Toolchain. Do note that since external toolchains have not been tested it is possible that they may not work properly across all environments.

An extra step needs to be taken if the plan is to use an external toolchain. By default, the software package will be set to check for the vendor toolchain that it was compiled with. In order to change this, use the following command inside the root directory of the BSP:

```
ptxdist platformconfig
```

Navigate to *architecture->toolchain* and clear the "*check for specific toolchain vendor*" field.

2.5.2 Building OSELAS Toolchains

Download OSELAS.Toolchain-1.99.3.7 <http://www.pengutronix.de/oselas/toolchain/download/OSELAS.Toolchain-1.99.3.7.tar.bz2> and place it in the `~ (/home/<user>)` directory then perform the following commands:

```
~$ tar -jxvf OSELAS.Toolchain-1.99.3.7.tar.bz2
~$ cd OSELAS.Toolchain-1.99.3.7
~/OSELAS.Toolchain-1.99.3.7$ ptxdist select ptxconfigs/arm-cortexa8-linux-
gnueabi_gcc-4.3.2_glibc-2.8_binutils-2.18_kernel-2.6.27-sanitized.ptxconfig
~/OSELAS.Toolchain-1.99.3.7$ ptxdist go
```

Optional:

```
~/OSELAS.Toolchain-1.99.3.7$ cd ..
~$ rm -rf OSELAS.Toolchain-1.99.3.7*
```

The toolchain is now built and installed in `/opt/OSELAS.Toolchain-1.99.3/arm-cortexa8-linux-gnueabi` and ready to be used for building the BSP.

2.5.2.1 Protecting Toolchains

It is recommended, although completely optional, to set the `/opt/OSELAS.Toolchain-1.99.3/arm-cortexa8-linux-gnueabi` directory and its contents as read-only to prevent accidental modifications to the toolchain.

3 Board Setup

The phyCORE-AM3517 comes pre-flashed with a recent version of x-load, u-boot, linux kernel, and jffs2 filesystem.

- After the device is out of the box and setup, make sure that there is a DB9/DB25 cable attached between the host machine and UART3 on the AM3517 carrier board to enable RS-232 serial communication.
- In a terminal on the host, access minicom and set it up, allowing console access over the serial port.

```
/* If minicom is not installed */  
sudo apt-get install minicom  
minicom -c on -s
```
- Navigate to *Serial port setup* in minicom and modify line "A - Serial Device : " to read `/dev/ttyS0`.



TIP

NOTE: The serial device is dependent on what COM port you are connected to on your system, so `/dev/ttyS0` is merely an example.

- Next, modify "E - Bps/Par/Bits : " to have a speed of 115200 and 8-N-1 (8N1) for the stop bits.
- Return to the main menu of minicom and select *Save setup as dfl* to make this the default setup anytime minicom is loaded, meaning `minicom -c on` is all that needs to be done in the future for this machine to be able to communicate with the kit.

3.1 Basic Settings

The settings most necessary for operation at this point are environment variables in U-Boot following the initial setup of minicom (or any other serial monitor):

```
PCM048 # print
PCM048 # setenv ethaddr ##:##:##:##:##:##
PCM048 # setenv ipaddr ###.###.###.###
PCM048 # setenv serverip ###.###.###.###
PCM048 # setenv gatewayip ###.###.###.###
PCM048 # setenv netmask ###.###.###.###
PCM048 # setenv nfshost ###.###.###.###
PCM048 # setenv rootpath /<nfs_mount_location>
PCM048 # save
```

- **ethaddr**
MAC address for the device. If this is not already defined in the list of environment variables, it will be located on a small sticker on the SOM.
- **ipaddr**
A dedicated IP address for the SOM. This is crucial if TFTP will be used for updating the device's images at any point.
- **serverip**
IP address of the host or another machine where the TFTP directory, if it exists, is located.
- **gatewayip**
Gateway IP for the network. This is only necessary if the TFTP directory is located on another network.
- **netmask**
Netmask for the network: typically 255.255.255.0. This is only necessary if the TFTP directory is located on another network.
- **nfshost** (required for NFS)
IP address of the NFS host.
- **rootpath** (required for NFS)
Location of the path to the NFS directory on the host system. e.g.
/home/<user>/nfs-export



NOTE: **help** is a useful tool in u-boot to show available commands and usage.

3.2 Selecting Boot Modes

The boot mode is selected using switches S9 and S10 on the carrier board. This selection identifies the location from where the X-Loader and U-Boot binaries are loaded for execution.



Be careful when changing these switches as the S9 and S10 packages are rotated 180° off of each other so the switches will be on opposite sides for both OFF and ON and count 1-4 in different directions.

To boot (X-Loader and U-Boot) from NAND, use this switch setting:

S9-1 OFF S9-2 OFF S9-3 OFF S9-4 OFF

S10-1 OFF S10-2 OFF S10-3 OFF S10-4 OFF

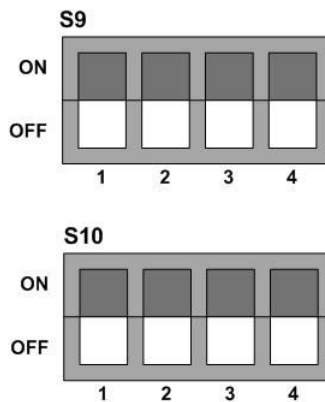


Fig. 3-1. NAND Switch Setting

To boot (X-Loader and U-Boot) from MMC, use this switch setting:

S9-1 OFF S9-2 ON S9-3 OFF S9-4 OFF

S10-1 ON S10-2 OFF S10-3 OFF S10-4 OFF

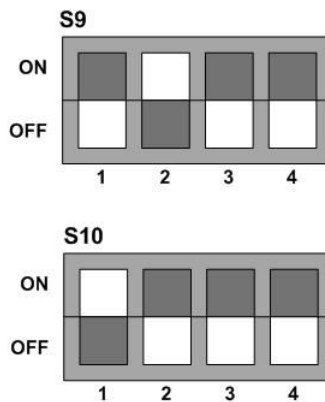


Fig. 3-2. MMC Switch Setting

3.3 Stand-Alone Booting

By default, the kit comes setup to boot from the provided images loaded on the NAND flash. Executing the following commands in U-Boot tells the system to boot using the linux kernel and filesystem located in NAND.

```
PCM048 # setenv bootcmd 'run boot_nand'  
PCM048 # save
```

3.4 Remote Booting

For development it may be beneficial to modify the boot settings to allow the kernel to be loaded from TFTP, and/or mount a network file-system (NFS) so that updates to the kernel can be tested quickly using custom applications hosted on the NFS.

Boot Linux Kernel via NAND with NFS

```
PCM048 # setenv bootcmd 'run boot_nand_nfs'  
PCM048 # save
```

Boot Linux Kernel via TFTP with NFS

```
PCM048 # setenv bootcmd 'run boot_tftp_nfs'  
PCM048 # save
```

Boot Linux Kernel via TFTP with Root Filesystem in NAND

```
PCM048 # setenv bootcmd 'run boot_tftp_nand'  
PCM048 # save
```

3.5 Other Booting Options

There are additional options for booting from MMC that may be handy as well.

Boot Linux Kernel via MMC with NFS

```
PCM048 # setenv bootcmd 'run boot_mmc_nfs'  
PCM048 # save
```

Boot Linux Kernel via MMC with Root Filesystem in NAND

```
PCM048 # setenv bootcmd 'run boot_mmc_nand'  
PCM048 # save
```

Boot Linux Kernel via MMC with Root Filesystem on MMC

```
PCM048 # setenv bootcmd 'run boot_mmc'  
PCM048 # save
```

3.6 Display Settings

By default the system is set to support the 5" VGA LVDS TFT-LCD display with integrated touch and adapter board (part# LCD-014). Listed below are settings to modify the bootargs in U-Boot for alternate display options.

Use DVI

```
PCM048 # setenv disp '"dvi"'
PCM048 # setenv fbm 'dvi:1440x900MR@60'
PCM048 # save
```

Use LCD-014

```
PCM048 # setenv disp '"lcd"'
PCM048 # setenv fbm 'lcd:640x480@60'
PCM048 # save
```

Use LCD-011

```
PCM048 # setenv disp '"lcd"'
PCM048 # setenv fbm 'lcd:240x320'
PCM048 # save
```

It is also possible to change the display on-the-fly once Linux has been booted.

LCD -> DVI Linux Kernel Settings

```
root@phyCORE:~ echo 0 > /sys/devices/platform/omapdss/display0/enabled
root@phyCORE:~ echo "dvi" > /sys/devices/platform/omapdss/manager0/display
root@phyCORE:~ echo 1 > /sys/devices/platform/omapdss/display2/enabled
```

DVI -> LCD Linux Kernel Settings

```
root@phyCORE:~ echo 0 > /sys/devices/platform/omapdss/display1/enabled
root@phyCORE:~ echo "lcd" > /sys/devices/platform/omapdss/manager0/display
root@phyCORE:~ echo 1 > /sys/devices/platform/omapdss/display0/enabled
```

4 Building a BSP

4.1 Managing Configurations

PTXdist uses the KConfig files present throughout the BSP to allow users to easily configure individual settings and drivers.



NOTE: All target source code is located in the ***OSELAS.BSP-Phytec-phyCORE-AM35xx/platform-phyCORE-AM35xx/build-target/*** directory.

For driver development and general settings or carrier board design it is necessary to know about the following three files to help integrate and modify features on the system:

1. linux kernel source is ***build-target/linux-2.6.32***
linux-2.6.32/arch/arm/mach-omap2/board-pcm048.c is the board config file
2. u-boot source is ***build-target/u-boot-2009.11***
u-boot-2009.11/include/configs/pcm048.h is the u-boot config file
3. x-loader source is ***build-target/x-load-f243938***
x-load-f243938/include/configs/pcm048.h is the x-loader config file

4.1.1 Platform

The platform config contains the default settings for each platform including what bootloaders, kernels, and filesystem images are to be built. The settings for the platform are modified using the following command:

```
ptxdist platformconfig
```

As a user, it is rare to ever modify these settings, but it may be useful to view them.

4.1.1.1 Enabling Graphics SDK

One of those rare modifications to the platformconfig is to enable the omap35_graphics_sdk. The image that is pre-installed on the NAND flash for the kit already has the Graphics SDK enabled, but each user must register on TI's website and download the Graphics SDK http://software-dl.ti.com/dsps/dsps_public_sw/sdo_sb/targetcontent/gfxsdk/latest/index_FDS.html.

After the download is complete, copy the **Graphics_SDK_setuplinux_#_##_##_##_#.bin** file to the **OSELAS.BSP-Phytec-phyCORE-AM35xx-PD11.1.0/local_src** directory.

To enable it to be built into the next image, execute the following:

```
ptxdist platformconfig
```

Following that, select the *menu option Graphics driver Menu --->* and then select the *omap35x_graphics_sdk* option. Beneath that, if the current listed SDK version does not match the *#_##_##_##_* pattern of your download, select and change it accordingly. Exit the configuration completely and select **Yes** when it asks if you would like to save your configuration.

Now, after a successful build, the graphics services will be enabled and automatically load when this image is booted.

4.2 Kernel

Use the kernel config to modify what drivers and support are included in a linux kernel build. It is essential to modify this configuration file during development especially when hardware devices are changed from the development kit, affecting the overall driver and settings structure.

```
ptxdist kernelconfig
```

4.2.1 Root Filesystem

In the overall menuconfig it is easy to modify the applications and content that is built into the filesystem by toggling options in the base configuration. This allows both minimal and more complete filesystem builds to be created easily.

```
ptxdist menuconfig
```

4.3 Building Images with PTXdist

Building images with PTXdist is very simple. The command `ptxdist go` does all of the required steps to compile all of the sources and make changes only where necessary. `ptxdist images` builds the filesystem images.

```
ptxdist go
ptxdist images
```

All images are then stored in

OSELAS.BSP-Phytec-phyCORE-AM35xx/platform-phyCORE-AM35xx/images.

5 Flashing Images

Flashing images can be relatively complex and there are two main ways to accomplish this: TFTP and MMC. If you wish to flash an image with a different name than what is designated in orange below, set the respective U-Boot environment variable to reflect that change in name:

- xload=x-load.bin.ift
- uboot=u-boot.bin
- bootfile=linuximage
- fileys=root.jffs2



TIP

If for any reason it is necessary to re-flash the example images, refer to [Chapter 2.1](#) Installing the phyCORE-AM3517 Kit CD to locate the images.

5.1 X-Loader

Method: TFTP. A valid **x-load.bin.ift** image must be on the TFTP server

```
PCM048 # run update_xload
```

Method: MMC. A valid **MLO** image must be on the SD/MMC card

```
PCM048 # mw.b 0x81600000 0xff 0x20000
PCM048 # mmc init
PCM048 # fatload mmc 0 ${loadaddr} MLO
PCM048 # nand erase 0 20000
PCM048 # nandeccl hw
PCM048 # nand write.i ${loadaddr} 0 20000
```

5.2 U-Boot

Method: TFTP. A valid **u-boot.bin** image must be on the TFTP server

```
PCM048 # run update_uboot
```

Method: MMC. A valid **u-boot.bin** image must be on the SD/MMC card

```
PCM048 # mw.b 0x81600000 0xff 0x160000
PCM048 # mmc init
PCM048 # fatload mmc 0 ${loadaddr} ${uboot}
PCM048 # nand erase 80000 160000
PCM048 # nandeccl sw
PCM048 # nand write.i ${loadaddr} 80000 160000
```

5.2.1 Restore default U-Boot settings

If the U-Boot environment variables need to be restored for any reason, simply delete the parameter save location in NAND, and the defaults will be restored with the next boot.

```
PCM048 # nand erase 240000 40000
```

5.3 Kernel

Method: TFTP. A valid `linuximage` must be on the TFTP server

```
PCM048 # run update_kernel
```

Method: MMC. A valid `linuximage` must be on the SD/MMC card

```
PCM048 # mw.b 0x81600000 0xff 0x300000
PCM048 # mmc init
PCM048 # fatload mmc 0 ${loadaddr} ${bootfile}
PCM048 # nand erase 280000 300000
PCM048 # nandeccl sw
PCM048 # nand write.i ${loadaddr} 280000 300000
```

5.4 Root Filesystem

Method: TFTP. A valid `root.jffs2` image must be on the TFTP server

```
PCM048 # run update_rootfs
```

Method: MMC. A valid `root.jffs2` must be on the SD/MMC card

```
PCM048 # mw.b 0x81600000 0xff 0x6000000
PCM048 # mmc init
PCM048 # fatload mmc 0 ${loadaddr} ${filesys}
PCM048 # nand erase 780000 6000000
PCM048 # nandeccl sw
PCM048 # nand write.i ${loadaddr} 780000 6000000
```


Appendix A: Helpful Hints and Tips

A.1 Where to Find More Information

If any additional help is required, below is a list of links that will supply additional information about a variety of topics:

- PHYTEC phyCORE-AM3517 Developer Wiki:
<https://www.phytec.com/wiki/index.php?title=Category:PhyCORE-AM3517>
- Texas Instruments Sitara Linux Wiki:
http://processors.wiki.ti.com/index.php/Category:Sitara_Linux
- For technical support on phyCORE-AM3517, contact support@phytec.com
- For commercial support, contact sales@phytec.com