

**Android 8.1 BSP Source Release Notes**

**v1.00**

**ANDROID**



***TechNexion***  
INNOVATORS OF TECHNOLOGY

## TABLE OF CONTENTS

Supported Products.....	3
Obtaining the Android 8.1 BSP source code .....	3
Preparing a build system.....	3
Setting up a linux host.....	3
Building Android.....	5
Building commands overview.....	5
An example build .....	5
Changing build modes.....	6
Troubleshooting .....	6
General.....	6
WiFi Problems.....	6
For more information, please visit as below website. ....	6

## Android 8.1 BSP Source Release Notes v1.00 2018-03-26

### Supported Products

This release supports the following products:

- PICO-IMX8 + PICO-PI

### Obtaining the Android 8.1 BSP source code

The Android 8.1 source code for TechNexion products is freely downloadable at below link.

[ftp://ftp.technexion.net/development\\_resources/development\\_tools/NXP/android/8.1/](ftp://ftp.technexion.net/development_resources/development_tools/NXP/android/8.1/)

### Preparing a build system

The minimum system requirements to build from source are as below:

- A 64-bit environment is required.
- Processor is expected to have 4 cores with a clock speed of above 2Ghz as a min.
- 16 GB RAM working memory
- 300 GB free hard disk space

### Setting up a Linux host

An android build can be very "picky" about software versions on the host machine.

Google maintains setup instructions for official Android releases at

<http://source.android.com/source/initializing.html>

TechNexion (and NXP/Freescale) Android source code requires some additional packages on top of the Google requirements.

On an Ubuntu 18.04 LTS 64-bit based system, these additional packages can be installed by:

```
$sudo apt-get update

$sudo apt-get install bc make build-essential python

$sudo apt-get install zip unzip libxml2 flex bison m4 ccache

$sudo apt-get install libc6:i386 libncurses5:i386 libstdc++6:i386
```

## Android 8.1 BSP Source Release Notes v1.00 2018-03-26

```
$sudo apt-get install uuid uuid-dev  
  
$sudo apt-get install zlib1g-dev liblz-dev  
  
$sudo apt-get install liblzo2-2 liblzo2-dev  
  
$sudo apt-get install lzop  
  
$sudo apt-get install git-core curl  
  
$sudo apt-get install u-boot-tools  
  
$sudo apt-get install mtd-utils  
  
$sudo apt-get install android-tools-fsutils  
  
$sudo apt-get install openjdk-8-jdk  
  
$sudo apt-get install g++-multilib  
  
$sudo apt-get install gcc-multilib  
  
$sudo apt-get install lib32ncurses5-dev  
  
$sudo apt-get install lib32readline6-dev  
  
$sudo apt-get install lib32z1-dev  
  
$sudo apt-get install libxml2-utils  
  
$sudo apt-get install gcc-aarch64-linux-gnu  
  
$sudo apt-get install gcc-arm-linux-gnueabi
```

## Building Android

Building Android is done in two steps. The first step configures the target product, and the second step does the build itself.

### Building commands overview

The general commands for Android builds are

```
source {file}
```

to set the environment for a specific target (specified by *{file}*),

```
cook
```

for a full clean build,

```
heat
```

for an incremental build, and

```
throw
```

for cleaning the build files. The 'cook' and 'heat' commands can accept a parameter '-j *N*' to build on *N* processors. It is recommended to use a value of *N* matching the number of CPUs in the computer. This can greatly decrease the build time.

### An example build

To configure the build target, one 'sources' the necessary environment variables from the `cookers/` folder. For example, to build Android for PICO-IMX8M module on a PICO-PI-8M with HDMI display, one can issue the following commands:

```
source cooks/env.bash.imx8.pico-8m.pi.hdmi
```

```
cook -j4
```

See the `cookers/` folder for additional available system configurations.

## Changing build modes

The default Android compile setting is to build in "engineering mode". This is for creating a development image with debug tools and looser permissions.

To change to "user mode" (sometimes the preferred production build) can be done by modifying some of the build scripts.

Edit `cookers/env.bash`, and within the `heat()` and `cook()` functions change the lunch command lines (one per function) from

```
lunch "$TARGET_DEVICE"-userdebug
```

to

```
lunch "$TARGET_DEVICE"-user
```

Examples of Android runtime images can be downloaded from [ftp://ftp.technexion.net/demo\\_software/](ftp://ftp.technexion.net/demo_software/)

## Troubleshooting

This section describes answers to common questions.

### General

Before starting please read the board documentation.

Additionally, TechNexion kernel releases have their own software troubleshooting guide which can be found in TechNexion knowledge base.

### WiFi

The firmware for WiFi chips is not distributed with this source code package.

To obtain the firmware contact [support@technexion.com](mailto:support@technexion.com).

After obtaining the firmware package extract the compressed file to the following path;

```
technexion-imx8_android-8.1.0_fullsource_20180919/device/fsl/pico_8m/wifi-  
firmware/
```

### For more information, please visit below websites.

<https://github.com/TechNexion/u-boot-edm/wiki>

<https://www.technexion.com/support/knowledge-base/>