**Introducing Android**

****    ****  ****

Android is an open-source platform developed under the Open Handset Alliance to enable faster development of mobile applications and provision of services to the user. Google is the leading company to develop and promote Android, however there are other companies as well who are involved in the development of Android. Some of these include T-Mobile, HTC, Qualcomm and others. Android literally means a robot with human qualities and it seems it is just what the developers of this amazing mobile platform that combines an operating system, middleware and powerful application development, intended.

Android made its debut in 2007 on Google's G1 mobile phone developed in collaboration with T-Mobile. Although Google has claimed that Android would be open source, which means developers can twist and tweak Android's code to come out with their own Android versions, the reality is a little different. It turns out that Google has allowed access to only selective parts of Android's code while some other parts are still off bound.

**Android Features:**

Android has many amazing and unique features that are of significance to developers and users alike, some of which are:

* Application Framework that enables reuse and replacement of components
* Optimized Graphics that is powered by customized 2D graphics library and 3D graphics based on the OpenGL ES 1.0 specification
* Media Support for common video, audio, and still image formats (MPEG4, H.264, MP3, AAC, AMR, JPG, PNG, GIF)
* Provision of Bluetooth, EDGE, 3G and Wi-Fi
* SQLite for structured data storage
* Open source WebKit engine based integrated web browser
* Camera, GPS, Compass, and Accelerometer (dependent on hardware support)
* GSM Telephony
* Dalvik virtual machine optimised for mobile devices
* Rich Development environment that includes a device emulator, debugging tools, performance and memory profiling and a Eclipse IDE plugin.

Google Android will be available with a host of features that includes a web browser, email client, calendar, contacts, SMS feature, maps and others. All the Google Android applications are written in Java and run on Dalvik virtual machine, which itself functions on top of a Linux kernel.

Android's outstanding features are that developers can get complete access to the identical framework APIs as used by the core applications. Besides this the app architecture is so designed so as to make the reuse of components simpler and the capabilities of any application can be published and used by any other app subject to the relevant security constraints. The users can also replace the components using the same mechanism.

The fully integrated Android package comprising an OS, middleware, applications and user friendly interface is expected to considerably speed-up product development while lowering the cost of mobile services development

# Android Basics

Still trying to find a good link for basic Android stuff.

## Adding apps/shortcuts to the home page

Follow these steps to add apps or shortcuts to your home pages

1. Click on the icon of 16 little boxes on the right side of the screen
2. Find the app you want to add to your home page
3. Click on the app and and hold down
4. The screen should change to your home page
5. While still holding down, move the app to where you want it on the home page
6. Release the app and it will be placed on your home page

## Removing apps/shortcuts from home page

Follow these steps to delete an app or shortcut from your home pages

1. Click on the app or shortcut and hold down until a trash can pops up on the right side of the screen where the 16 little boxes usually are
2. Drag the app or shortcut to the trash can
3. The app/shortcut should now be removed from your home page
* This does not delete the app from your device, only from the home page

## Installing Apps to your ZT-180

There are several ways to install apps

### Market

Get Market working by following one of the methods in the main wiki

### From a micro SD card

Follow these steps to install an app from a micro SD card

1. Download the app to a location on your PC where you can find it again
2. If the app has a .zip extension, use your favorite extracting program to extract it
3. Copy the app .apk file to a micro SD card
4. Remove the micro SD card from your PC
5. Insert the micro SD card into your ZT-180
6. Using your favorite file browser (or the default OI browser) find your app on the micro SD card
7. Click on the app to install it
8. Click on "Open App Manager"
9. Click on "Install"
10. Click on "Install" again
11. If you want to use the app now, click "Open"
12. If you don't want to use it now, click "Done"

### Browsing to a file with ZT-180 web browser

It hasn't worked for me but supposedly you can hold down on a downloadable file (non Dropbox) and install is straight from the internet

### Dropbox

Files that are Dropbox can be downloaded straight to the ZT-180 by clicking on them

## Removing apps from your ZT-180

Follow these steps to remove an application from the ZT-180

1. Click on the "Settings" icon
2. Click on "Applications"
3. Click on "Manage Applications"
4. Click on the application you want to remove
5. Click on "Unintstall"
6. Click on "OK"
7. Wait for the application to uninstall
8. Click on "OK"
9. The application should now be uninstalled

Return to ZT-180 FAQ - [Back](http://www.slatedroid.com/wiki/index.php?title=Wiki:ZT-180_FAQ)

## Finding Firmware Build Date

To find your firmware version -

1. Click on the Application Drawer (the icon on the right side of the screen with 16 little boxes)
2. Click on the "Settings" icon
3. Scroll down and click on "About Device" or "About MID"
4. There should be a date in "Firmware Version"

### Firmware Build Date Chart

#### October 2010

* 20101030 reads as - android2.1-update1-20101030
* 20101020 reads as - android2.1-update1-20101020

20101012 could be any of 3 versions of firmware. Use below to determine which one you most likely have -

* 20101020 - There is an orange lock icon in the status bar
* 20101014 - There is a "ts calibrate" option in "Dev Tools" icon in the App Drawer
* 20101013 - If neither of the two above options are present then this is your firmware version

#### September 2010

* 20100929 reads as - android2.1-update1-0929
* 20100926 reads as - 2.1-update1

# ADB Basics

## Preparation

Make sure you have downloaded, installed, and updated Android SDK. Also make sure the wifi on your tablet is on and there are no errors popping up before you try to connect (make note of your tablets IP Address). Before executing the following commands to connect, make sure the tablet is the only USB device currently on and attached to your computer (that's what the -d is for in the first command and bypasses the need for USB drivers).

## Commands to Connect

On your computer, from the command prompt, go to the tools subfolder, or if you updated Android SDK the platform-tools subfolder, of your android sdk folder and type:

 adb -d tcpip 5555

(with OTG USB cable attached) and hit Enter. You should then see:

 \* daemon not running. starting it now on port #### \*

 \* daemon started successfully \*

 restarting in TCP mode port: 5555

Then type:

 adb connect {Your tablets IP}:5555

(with cable out) and hit Enter. You should then see:

 connected to {Your tablets IP}:5555

Then you know your tablet is connected to your computer over wifi.

## Some Basic ADB Commands

you can type in **adb ?** to find out all the commands from your command prompt:

-d - directs command to the only connected USB device

 returns an error if more than one USB device is present.

-e - directs command to the only running emulator.

 returns an error if more than one emulator is running.

-s <serial number> - directs command to the USB device or emulator with the given serial number.

 Overrides ANDROID\_SERIAL environment variable.

-p <product name or path> - simple product name like 'sooner', or a relative/absolute path to a product out directory like 'out/target/product/sooner'.

 If -p is not specified, the ANDROID\_PRODUCT\_OUT environment variable is used, which must be an absolute path.

devices - list all connected devices

connect <host>[:<port>] - connect to a device via TCP/IP

 Port 5555 is used by default if no port number is specified.

disconnect [<host>[:<port>]] - disconnect from a TCP/IP device.

 Port 5555 is used by default if no port number is specified.

 Using this command with no additional arguments will disconnect from all connected TCP/IP devices.

device commands:

 adb push <local> <remote> - copy file/dir to device

 adb pull <remote> [<local>] - copy file/dir from device

 adb sync [ <directory> ] - copy host->device only if changed

 (-l means list but don't copy)

 (see 'adb help all')

 adb shell - run remote shell interactively

 adb shell <command> - run remote shell command

 adb emu <command> - run emulator console command

 adb logcat [ <filter-spec> ] - View device log

 adb forward <local> <remote> - forward socket connections

 forward specs are one of:

 tcp:<port>

 localabstract:<unix domain socket name>

 localreserved:<unix domain socket name>

 localfilesystem:<unix domain socket name>

 dev:<character device name>

 jdwp:<process pid> (remote only)

 adb jdwp - list PIDs of processes hosting a JDWP transport

 adb install [-l] [-r] [-s] <file> - push this package file to the device and install it

 ('-l' means forward-lock the app)

 ('-r' means reinstall the app, keeping its data)

 ('-s' means install on SD card instead of internal storage)

 adb uninstall [-k] <package> - remove this app package from the device

 ('-k' means keep the data and cache directories)

 adb bugreport - return all information from the device

 that should be included in a bug report.

 adb help - show this help message

 adb version - show version num

DATAOPTS:

(no option) - don't touch the data partition

 -w - wipe the data partition

 -d - flash the data partition

scripting:

 adb wait-for-device - block until device is online

 adb start-server - ensure that there is a server running

 adb kill-server - kill the server if it is running

 adb get-state - prints: offline | bootloader | device

 adb get-serialno - prints: <serial-number>

 adb status-window - continuously print device status for a specified device

 adb remount - remounts the /system partition on the device read-write

 adb reboot [bootloader|recovery] - reboots the device, optionally into the bootloader or recovery program

 adb reboot-bootloader - reboots the device into the bootloader

 adb root - restarts the adbd daemon with root permissions

 adb usb - restarts the adbd daemon listening on USB

 adb tcpip <port> - restarts the adbd daemon listening on TCP on the specified port networking:

 adb ppp <tty> [parameters] - Run PPP over USB.

Note: you should not automatically start a PPP connection.

<tty> refers to the tty for PPP stream. Eg. dev:/dev/omap\_csmi\_tty1

[parameters] - Eg. defaultroute debug dump local notty usepeerdns

adb sync notes: adb sync [ <directory> ]

 <localdir> can be interpreted in several ways:

 - If <directory> is not specified, both /system and /data partitions will be updated.

 - If it is "system" or "data", only the corresponding partition is updated.

environmental variables:

 ADB\_TRACE - Print debug information. A comma separated list of the following values

 1 or all, adb, sockets, packets, rwx, usb, sync, sysdeps, transport, jdwp

 ANDROID\_SERIAL - The serial number to connect to. -s takes priority over this if given.

 ANDROID\_LOG\_TAGS - When used with the logcat option, only these debug tags are printed.

**Google's Android operating system is the new kid on the block, which is seeing an incredible surge in popularity thanks to the support it is receiving from manufacturers. In Australia the range of Android phones is growing faster than any other, and include models across all pricing tiers, from top-shelf handsets to prepaid models as well.**

Learning any new system can be tough, especially when you have been using another system for many years. Google has done a good job of re-imaging a smartphone system, taking some of the best features from the systems we know and love, and adding a few new tricks of its own.

Most of the language used throughout the system include common-sense descriptions of each button and menu, but there are still a few tricky spots. This guide is intended for Android beginners and will complement the user manual that comes with your phone, not replace it. We'll be focusing on elements that are common to all Android phones, so keep your manual close for tips and tricks specific to your phone model.

## The Menu key



*There's a menu key on every Android.*(Credit: CBSi)

There is no better place to start this guide than to look at the physical Menu key on every Android phone. The menu key is like clicking the right mouse button on a PC; it brings up a context-sensitive list of options depending on which screen you are currently viewing.

Context sensitive means you'll get a different menu depending on the app you are using. In Google Maps the Menu key brings up options and search functionality, and in the music player the Menu key will let you switch albums and artists. The Menu key is your fix-anything-button — if you are looking for an option that isn't available on the touchscreen, press Menu and see if what you're looking for is there.

## Long presses

While the Menu key displays options and settings that will affect an entire application, a long press on the touchscreen of your Android phone will bring up item-specific options. This gesture should work on any of the core applications with lists of items like contacts, messaging, Gmail, the music player, calendar and image gallery.

In your email, for example, a long press on any email message in the Inbox will bring up options to delete, forward, reply, etc without having to open the message. A long press on a contact entry in your address book will bring up communications options for that contact, depending on what details you have entered about them.



*Three examples of the long press gesture: on the home screen, in a message and on a web page.*(Credit: CBSi)

**If you long press on...**

* **An empty area of the home screen:** it will open options to create a shortcut or widget in this space.
* **A shortcut on the home screen:** it will enable the ability to move the shortcut, drag to the bottom of the screen to remove it all together.
* **An app in the app drawer:** it will create a shortcut on the home screen for that app (keep holding to move the shortcut to where you want it).
* **A message thread in your inbox:** it will open options to delete a message or view the contact details.
* **A specific message in an SMS conversation:** it will open options to delete a message, copy message text and send it to your SIM.
* **A contact in your address book:** it will open options to edit contact details, link the contact to other contact details or delete the details altogether.
* **A date in the calendar:** it will quickly add an appointment option.
* **A hyperlink in a web page:** it will show you the options to open the link in a new window, copy the web address, bookmark the link, or share the link via email or SMS.

This is just a sample of the places where a long press gesture will open up options for you. Basically anything you can select with a short press also contains variable options under a long press gesture as well.

## Important settings

The global settings for all Android phones are pretty comprehensive and often self-explanatory. There are some important features buried in Settings, however, so read on for a guide as to where to find them.



*How to turn 3G data on and off.*(Credit: CBSi)

**Turning 3G on and off** — Android phones can chew through data, so knowing how to turn 3G data on and off is essential. From the home screen press Menu > Settings > Wireless and network > Mobile networks. Unfortunately, the specific option differs from phone to phone, on the HTC Desire you can uncheck the "Mobile Networks" box, while the Galaxy S only has a "Use only 2G networks" option, which leaves data on but reduces the capabilities of data transmission to a minimum.

**Turning Wi-Fi on and off** — in the same menu as the 3G switcher you'll find the option to turn on Wi-Fi and settings for connecting to networks in your vicinity. There's also an option to have Android notify you when you're in range of an open network, which is handy for grabbing free internet when you're out and about. If you need to add advanced Wi-Fi settings, like DNS server details, connect to a network then press the Menu button on your phone while you're in the Wi-Fi settings screen and select Advanced Settings.

**Accounts and Sync** — if you need to add, amend or delete any of your personal account details from the phone, this is the menu you need to find. Within this menu you can connect to your social networks and define how often you want your phone to synchronise with these networks. You can turn on and off the phone's automatic syncing feature in this menu as well, and unless you have a huge data cap, we suggest you turn this feature off and manually sync programs like email and Facebook updates.

**Factory data reset** — there are a number of reasons to want to wipe your personal information off of a mobile phone, and Android makes this task extremely easy to do — if you know where to find the option. It used to live under "SD card and phone storage" so if you are using Android version 1.5 or 1.6, this is where you should look. For newer versions of the platform, take a peek under the "Privacy" setting menu.

## The Launcher

The Launcher is the official name given to the home screen on an Android phone, and it refers to any program that controls the look and feel of these screens. For example, if you buy an HTC Desire the launcher is called HTC Sense and it looks and works differently to the launcher you would use if you bought a Samsung Galaxy S. The Launcher is the one place that manufacturers can differentiate their Android product from their competitors, and it is often the difference between a good Android phone and a lousy one.

Regardless of which Launcher software you use, there are a few key features that apply to them all.

**Adding shortcuts and widgets** — use a long press gesture on an empty space on any of your home screens to bring up a list of items you can add to that space, including widgets and shortcuts. If you download a widget from the Android Market it will show up in this list. Alternatively, if you long press on an app in the App Drawer it will transport a shortcut for that app to the home screen. Keep holding down on the icon and move it to a free space. If you want it on a different screen, move the icon to the left or right of the screen to change home screen panels.



*You can change the wallpaper in three steps.*(Credit: CBSi)

**Changing the wallpaper** — press the physical Menu key on your phone and select wallpaper. This will display all the locations of images on your phone. Find the image you want, crop it as necessary and apply.

**Notifications panel** — missed a call or received an SMS? Details of these events are stored in the notifications panel. Simply slide your finger downwards from the top of the screen to reveal this feature.

## Apps

Like Apple's iPhone, third-party applications are one of the strongest selling points for any new Android phone. If there is a feature missing from your phone chances are there's an app to fill that gap available to download from the Market. Using the Market is easy, and setting up an account for purchases is simple if you own a credit card.



*You'll need to add a credit card to your Google Account before purchasing apps on the Market.*(Credit: CBSi)

**How to purchase apps** — everything on your Android phone is linked back to a Google account, typically a Gmail account. The Android Market is no different, and all app purchases are credited to this same account. The first time you attempt to purchase an app you'll be asked for your credit card details. If you punch these details in they will be linked to your Google account and you won't have to enter them in again when making purchases.

**Uninstalling apps** — mobile phone applications stores are a real mixed bag, and chances are you will download plenty of apps you'll want to ditch after testing. The most efficient way to delete an app is to return to the Market, press the Menu key, select Downloads, choose the app from the list you downloaded and select uninstall. If you paid for the app you will also have the option to refund your money if you uninstall within 24 hours of purchasing. Alternatively, there are dozens of apps online that do the uninstalling for you. We use Uninstaller Pro, which has the option to batch uninstall a collection of apps in one hit.