
Openmoko™ is a project dedicated to delivering mobile phones with an open software stack. Openmoko is a project driven by a community of passionate and intelligent volunteers. Openmoko™, born as an Open Source project under GPL and LGPL license, is dedicated to deliver an open software stack on mobile platforms. Openmoko shipped its first product, the Neo 1973, on July 9 2007; and then turned into a start-up company with one aim: create great mobile products using the Openmoko stack.

## Main Projects Under Open Moko

The first sub-project is Openmoko Linux, a Linux-based operating system designed for mobile phones, built using free software.

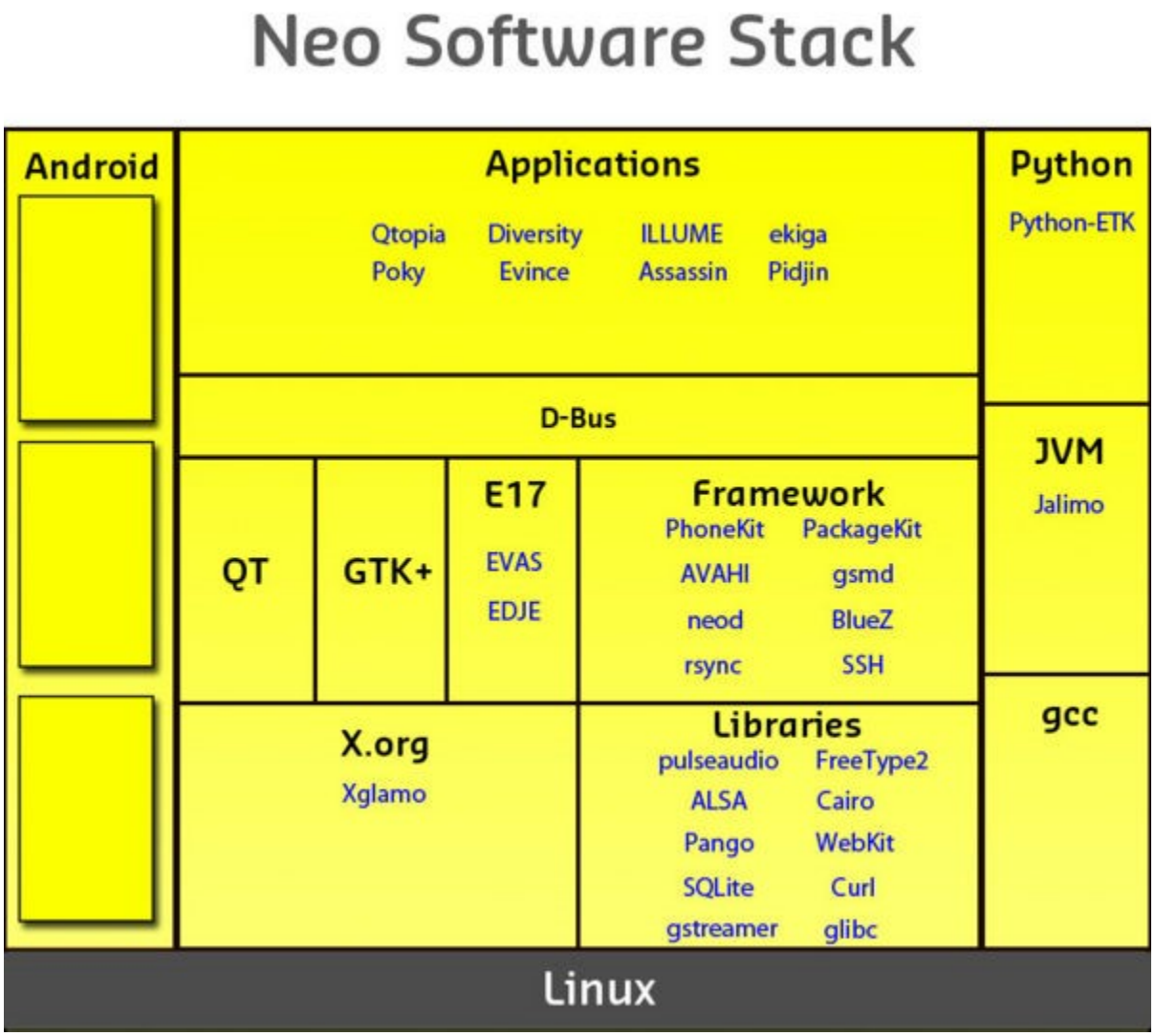
The second sub-project is the development of hardware devices on which Openmoko Linux runs. The first device released was the Neo 1973,which was followed up by the Neo FreeRunner on 25 June 2008. Unlike most other mobile phone platforms, these phones are designed to provide end users with the ability to modify the operating system and software stack.

## OpenMoko Linux

Openmoko Linux is the software subproject of Openmoko.
Initial versions of the software are developed for the hardware developed by the Openmoko project (Neo 1973 and Neo FreeRunner devices from FIC). There has been moderate effort as yet in using the OS on other hardware platforms.
Unlike most other mobile phone platforms, the phones on which Openmoko Linux runs are designed to provide end users with the ability to modify the operating system and software stack. The platform is also supported by other mobile phones.
Openmoko Linux uses the Linux kernel, GNU libc, the X. Org server plus their own graphical user environment built using the EFL toolkit, GTK+ toolkit, Qt toolkit and the illume window manager (previously Matchbox window manager) . The Open Embedded build framework and a modified version of ipkg package system, called Opkg, are used to create and maintain software packages.

This is a very different approach than that of Android (in which everything except

Linux, Webkit, and the Java language inside of Android seems non-standard). Applications targeted for Android must be substantially rewritten and are largely unportable. Many existing Linux desktop apps can be easily ported to Openmoko. (However the limited computational power and screen resolution require substantial reworking of existing applications, in order to render them usable in a finger-oriented, small-screen environment.)




The Kernel on the GTA01 is based on a vanilla 2.6.21.3 Linux kernel from kernel.org.

Some additional patches are required for

* S3C2410 Usb Device Controller
    o Driver from iPaq H1940 linux project
* S3C2410 SD Card Controller

    o Driver from the TomTom GO kernel
  * QT2410 machine support
    o This is just some glue that puts all pieces together
  * S3C2410 touch screen driver
    o Again from iPaq H1940 linux project
  * GTA01 machine support
    o Some glue/configuration to pull all pieces together
  * GSM Communication Infrastructure
    o The kernel-level part (TS07.10 line discipline, GPRS line discipline)

**Gnu Libc**

The GNU C Library**,** commonly known as glibc, is the C standard library released by the GNU Project**.** Originally written by the Free Software Foundation **(**FSF) for the GNU operating system.

Glibc is used in systems which run many different kernels and different hardware architectures. Its most common use is in systems using the Linux kernel on x86 hardware, but officially supported hardware includes: x86, Motorola 680x0, DEC Alpha, PowerPC, ETRAX CRIS, s390, and SPARC. It officially supports the Hurd and Linux kernels. Additionally, there are heavily patched versions that run on the kernels of FreeBSD and NetBSD (from which Debian GNU/kFreeBSD and Debian GNU/NetBSD systems are built, respectively), as well as the kernel of OpenSolaris[8]. It is also used (in an edited form) as the libroot of BeOS and hence Haiku.

glibc provides the functionality required by the Single UNIX Specification, POSIX (1c, 1d, and 1j) and some of the functionality required by ISO C99, Berkeley Unix (BSD) interfaces, the System V Interface Definition (SVID) and the X/Open Portability Guide (XPG), Issue 4.2, with all extensions common to XSI (X/Open System Interface) compliant systems along with all X/Open UNIX extensions.

In addition, glibc also provides extensions which have been deemed useful or necessary while developing GNU

**X.Org Server**

The X.Org Server (officially the X.Org Foundation Open Source Public Implementation of X11) is the X server in the official reference implementation of the X Window System. The current stable release is 1.6.1, which is part of X11R7.5, and was released on 25 February 2009.It is both open source and free software.

The project is supported and overseen by the X.Org Foundation and is hosted by freedesktop.org.

The X.Org Server is increasingly popular with the free software Unix-like operating systems, being adopted in most Linux distributions and BSD variants, with the exception of NetBSD (X.Org is part of the base system in NetBSD 5.0, a few less common

platforms still use Xfree86). It is also included in Sun Microsystems' Solaris, and is the server of choice for x86 systems; SPARC-based systems almost exclusively use Sun Microsystems's proprietary Xsun server, as Sun graphics driver support for X.Org is very limited. It is also used in Cygwin/X, Cygwin's implementation of the X server for Microsoft Windows, and in Xming. Mac OS X versions prior to 10.5 ("Leopard") ship with an XFree86-based server, but 10.5's X server is based on the X.Org codebase.

**GTK+**

GTK+, or The GIMP Toolkit, is a cross-platform widget toolkit for creating graphical user interfaces. It is one of the most popular toolkits for the X Window System, along with Qt.

GTK+ was initially created for the GNU Image Manipulation Program (GIMP), a raster graphics editor, in 1997 by Spencer Kimball and Peter Mattis, members of eXperimental Computing Facility (XCF) at UC Berkeley.

Licensed under the LGPL, GTK+ is free software and is part of the GNU Project.

GTK+ is written in the C programming language, and its design uses the GObject object system. The GNOME platform provides language bindings for:

* C
* C++ (gtkmm)
* Perl (Gtk2-perl)
* Ruby (ruby-gtk2)
* Python (PyGTK)
* Java (java-gnome) (not available for Microsoft Windows)
* C# (Gtk#)

GTK+ was originally targeted at the X Window System, and this remains its primary target platform. Other targeted platforms are Microsoft Windows (Windows 2000 and upwards, near complete support), DirectFB, and Quartz (Mac OS X v10.4 and upwards, still under development).

Environments that use GTK+

* GNOME is based on GTK+, meaning that GNOME programs use GTK+
* Xfce is based on GTK+, though its applications typically do not depend on as many libraries (this is the difference between something being branded as a “GNOME program” or as a “GTK+ program”).
* LXDE is based on GTK+, stands for "Lightweight X11 Desktop Environment"
* ROX Desktop a lightweight desktop, with features from the GUI of RISC OS
* GPE Palmtop Environment
* Maemo (Nokia's Internet-tablet framework)
* Access Linux Platform (successor of the Palm OS PDA platform)

* One Laptop Per Child project uses GTK+ and PyGTK
* Broncho project (A mobile linux project)

**EFL Library**

The Enlightenment Foundation Libraries, or EFL, is a set of open source graphical software libraries that grew out of the Enlightenment window manager project and is developed by Enlightenment.org with some sponsorship from Terra Soft Solutions.[1] The project's focus is to make the EFL a flexible yet powerful and easy to use set of tools to extend the capabilities of both the Enlightenment window manager and other software projects based on the EFL. The libraries were created for version 0.17 of the window manager. The libraries are meant to be portable and optimized to be functional even on devices such as PDAs.

Core components

Evas

Evas is the EFL canvas library, for creating areas, or windows, that applications can draw on in an X Window System. The EFL uses hardware-acceleration where possible to allow it to work faster, but is also designed to work on lower-end hardware, falling back to lower color and quality for graphics if necessary. Unlike most canvas libraries, it is primarily image-based (as opposed to vector-based) and fully state-aware (the vast majority of canvases are stateless, requiring the programmer to keep track of state).

Edje

Edje is a library that attempts to separate the user interface from the application. It allows applications to be skinnable, so that it is possible to change the GUI of an application without changing the application itself. Edje-based applications use files which contain the specifications for the GUI layout that is to be used. Edje themes are contained using EET generated files.

Ecore

Ecore is an event abstraction, and modular convenience library, intended to simplify a number of common tasks. It is modular, so applications need only call the minimal required libraries for a job. Ecore simplifies working with X, Evas, and also a few other things, such as network communications.

Embryo

Embryo implements a scripting language used by other parts of the EFL, such as Edje. The language has a C-like syntax, and was based on the SMALL language.

### EET

EET is a library that allows the creation of compressed archive-like files, similar to .zip files, but designed to be light-weight, efficient and quick. EET forms the basis of theme files in the EFL, i.e. if you want to install a theme for Enlightenment or another themable EFL app, you would be installing an EET-format file, which contains all of the theme graphics and configuration and do not need to be extracted onto the filesystem in order to be used.

## Helper Components

### Imlib2

While not strictly part of the EFL, Imlib2 is the library used by the EFL for low-level graphics manipulation and display. It is capable of loading and manipulating graphics files of many formats, and is also capable of displaying them in an X Window System. The EFL developers boast that Imlib2 is very optimized and that it does what it does "faster than anything else".

### EDB

EDB is a library wrapped around the Berkeley DB 2.7.7 library, intended to provide a database API that is quick and easy to use.

### Exml

This library provides an abstract interface to an XML parser/writer, using ecore data structures and making things generally easy to get around in.

### Epeg

Epeg is a minor library that creates thumbnails of large numbers of JPEG images very quickly.

### Epsilon

Epsilon is a library for creating thumbnails of many types of images, designed to be compliant with freedesktop.org's Thumbnail Managing Standard. Epsilon supports all of the file formats that Imlib2 supports, including PNG, JPEG (will use Epeg for this if it is present), TIFF, XCF, GIF, etc.

Engrave

This library provides an API for editing EDJE's .edj files.

Esmart

Esmart is a library providing a collection of smart objects, objects that provide some functionality or other in an easy-to-use container.

Emotion

Emotion is a library providing video-playing capabilities through the use of smart-objects. Emotion is based on libxine, a well established video playing library, and so supports all of the video formats that libxine supports, including Ogg Theora, DivX, MPEG2, etc.

Etk

Etk, or the Enlightened ToolKit is a widget set based on the EFL that makes heavy use of Evas and Edje to provide a fast, stable, and scalable library that can be used to create both rich and fast applications that can be used on anything from every day desktop computers to small PDA's and set-top boxes.

EWL

EWL is the Enlightenment Widget Library and it provides themable widgets for applications and serves as yet another way to easily start creating applications with the EFL. The EWL provides typical widgets such as windows, buttons, lists, text views, etc. and the theme engine allows for animated effects and such.

Defunct/Deprecated EFL Components

* EWD (features now added to ecore)
* Estyle (features have been absorbed by etox)
* Etox (obsolete)
* Ebits (replaced by Edje)
* Evoak (dead)

**Qt toolkit**

Qt (pronounced as the English word "cute") is a cross-platform application development framework, widely used for the development of GUI programs (in which case it is known as a widget toolkit), and also used for developing non-GUI programs such as console tools and servers. Qt is most notably used in KDE, Opera, Google Earth, Skype,

Qt Extended, Adobe Photoshop Album, VirtualBox and OPIE. It is produced by the Norwegian company Qt Software, formerly known as Trolltech, a wholly owned subsidiary of Nokia since June 17, 2008.

Qt uses C++ with several non-standard extensions implemented by an additional pre-processor that generates standard C++ code before compilation. Qt can also be used in several other programming languages; via language bindings. It runs on all major platforms, and has extensive internationalization support. Non-GUI features include SQL database access, XML parsing, thread management, network support and a unified cross-platform API for file handling.

Distributed under the terms of the GNU Lesser General Public License (among others), Qt is free and open source software.

Qt is released by Trolltech on the following platforms:

* Qt/X11 – Qt for X Window System (Unix / Linux)
* Qt/Mac – Qt for Apple Mac OS X
* Qt/Windows – Qt for Microsoft Windows
* Qt/Embedded – Qt for embedded platforms (PDA, Smartphone, etc.)
* Qt/WinCE – Qt for Windows CE
* Qt Jambi – Qt for Java
* Qt Extended – Application platform for Embedded Linux-based mobile computing devices (Discontinued)

Qt software on October 20, 2008 announced a version of Qt on S60 platform.

There are four editions of Qt available on each of these platforms, namely:

* Qt Console – edition for non-GUI development
* Qt Desktop Light – entry level GUI edition, stripped of network and database support
* Qt Desktop – complete edition
* Qt Open Source Edition – "complete" edition, with some exceptions,[12] for free software/open source developers

## Hardware

The Phones released under the project are -

**Neo 1973** - The Neo 1973 is the first phone designed to run Openmoko. It is a phone that can be used with any GSM operator, and it is manufactured by FIC who instigated the Openmoko project. The Neo1973 phone is the first hardware platform

to take advantage of Openmoko.


**Physical Dimensions**

* 120.7 x 62 x 18.5 mm (4.75 x 2.44 x 0.728 inch)
* 184 +/- 5 g (6.5 ounces)
* For the purposes of acquiring/cutting a properly sized screen protector, the display hole of the case is about 45 x 59 mm, while the top cover internal frame can house up to a 53 x 74 mm protector; sizes much larger than the display hole would obviously necessitate removing the front cover for installation.
* A Useful size comparison between the Neo1973, iPhone, Motorola A1200 and the SEM600i can be seen at sizeasy

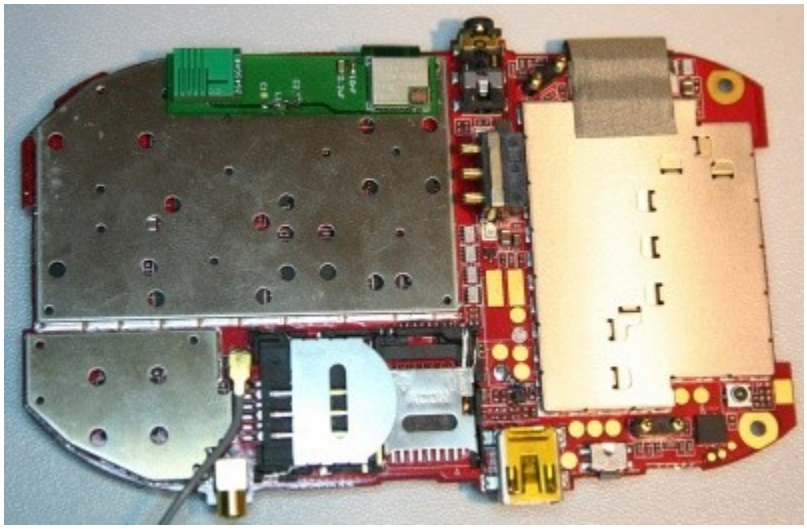
**Main components**

Display(Top) Side

## Processor

The main Processor (CPU) of the Neo1973 is a Samsung S3C2410AL-26 (Capable of running up to 266 MHz)

* Product Homepage: S3C2410.htm
* User Manual: 2410UserManual.pdf
* Core: ARM920T
* Instruction Set: ARMv4
* BSDL File: S3C2410_BGA_BSDLJTAGFILE.bsd
* GPIO Assignments: https://svn.openmoko.org/trunk/doc/hardware/GTA01Bv4/gpio.txt



Component(Back) side

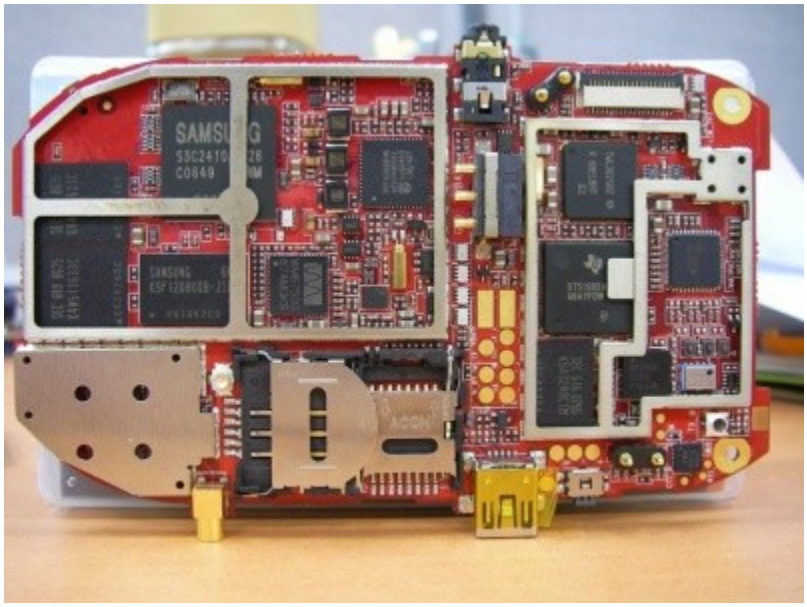
## Flash

64MB Samsung NAND flash (K9F1208U0B) attached to S3C2410 NAND controller.

* Product Homepage: [1]
* Data Sheet: [2]
* Connected to: S3C2410 NAND controller

This is the only flash memory in the device. The S3C2410 boots directly from nand, using the S3C2410 Steppingstone.

Only Free software is used .



Component Side

**RAM**

128MB SDRAM (2x Samsung K4M511633C) attached to S3C2410 SDRAM controller

* Product Homepage: [3]
* Data Sheet: [4]
* Connected to: S3C2410

**GSM/GPRS**

The GSM (including GPRS) modem is Texas Instruments Calypso based.

* Connected to: S3C2410 UART1 (full-uart, RxD, TxD, CTS, RTS), /dev/ttySAC0 in userspace
* PM Driver: https://svn.openmoko.org/trunk/src/target/kernel/patches/gta01-power_control.patch
* Accessible GSM/GPRS antenna jack (if battery cover is removed)

CALYPSO digital baseband

Unfortunately we cannot provide many details on the GSM chipset due to very tight NDAs. However, this is not neccessarily required, since it interfaces using a standard UART serial line with the S3C2410. On that interface, GSM 07.05, GSM 07.10 and other standardized protocols are used.
TWL3014 analog baseband

Product Homepage: TWL3014
TRF6151 RF Transceiver

Product Homepage: TRF6151
GPRS Class12/CS4
AGPS

Hammerhead PMB 2520 AGPS from Global Locate.

* Product Homepage: Hammerhead
* Connected to: S3C2410 UART2 (full-uart, RxD, TxD, CTS, RTS) /dev/ttySAC1 in userspace
* Driver: Implemented as a binary with NMEA output suitable for the gpsd daemon (gpsd)
* PM Driver: https://svn.openmoko.org/trunk/src/target/kernel/patches/gta01-power_control.patch
* Externally-accessible GPS antenna connector of type MMCX

The binary driver is available - see GLLIN. Efforts to reverse engineer the protocol are partially detailed in Hammerhead/Protocol, these have stalled since the announcement that FreeRunner will use a different GPS.

**MicroSD-Card**

The Neo1973 has one microSD aka Transflash slot. It supports SDHC. MicroSD slot is under battery.

* Connected to: S3C2410 MMC/SD controller
* Mounted to: /media/card
* Driver: https://svn.openmoko.org/trunk/src/target/kernel/patches/s3c_mci.patch
* Supported microSD cards
* Specifications: SD Simplified Specification, MMC (partial), MMC (product

manual)

**LCD Module (LCM)**

This is a 2.8" diagonal (1.7" x 2.27" - 43mm x 58mm) 480x640 toppoly (tpo) TD028TTEC1 module (283 DPI), using a Toshiba JBT6K74 TFT LCD Driver Chipset.

* Homepage: Activer-Matrix-VGA.htm
* Driver: https://svn.openmoko.org/trunk/src/target/kernel/patches/gta01-jbt6k74.patch
* Backlight Driver: https://svn.openmoko.org/trunk/src/target/kernel/patches/gta01-backlight.patch
* Connected to: S3C2410 Display Controller and S3C2410 SPI Interface channel 1
* Backlight controllable via /sys/class/backlight/gta01-bl

**Touch Screen**

* Connected to: S3C2410 TS controller
* Driver: https://svn.openmoko.org/trunk/src/target/kernel/patches/s3c2410_touchscreen.patch

**Stylus**

Seemingly identical to this one on ebay
Bluetooth

Delta DFBM-CS320 Class2 Module, using CSR BlueCore4 (V2.0+EDR).

* Data Sheet: 2.DFBM-CS320.pdf
* CSR Data Sheet: CS-101564-DSP10 BlueCore4-ROM Product Data Sheet.pdf
* Driver: Stock Linux Kernel BlueZ
* Connected to: S3C2410 USB Host controller (OHCI)
* PM Driver: https://svn.openmoko.org/trunk/src/target/kernel/patches/gta01-power_control.patch

**Vibrator**

* Driver: https://svn.openmoko.org/trunk/src/target/kernel/patches/gta01-vibrator.patch
* Connected to: S3C2410 GPIO
* Controllable via /sys/class/leds/gta01:vibrator

**USB Host**

The USB Host controller is inside the S3C2410

* Driver: Stock Linux kernel ohci_hcd

**USB Device**

The USB Device controller is inside the S3C2410

* Driver: https://svn.openmoko.org/trunk/src/target/kernel/patches/s3c2410_udc.patch
* Please see USB Product IDs on information about which Vendor/Product IDs we use
* 1200mAh lithium battery charges when connected to powered host.
* Mini-B connector this one.
* This can be used as a USB host: Neo1973_USB_host

**I2C Devices**

The I2C is a simple communication standard intended to move small amounts of data a few inches between chips. Please see Neo I2C Devices for more information & a list of devices & the addresses currently in use & documented for the Neo1973.
Audio

See also: Neo1973 Audio Subsystem
Wolfson Codec

Main article: WM8753

There's a WM8753 Wolfson Microelectronics codec (This is not a "smart" codec that can interpret MP3/... it is a simple dumb "sound card".
Stereo Amplifier

There's a National Semiconductor LM4857 Stereo Amplifier at the analog audio output of the WM8753

* Product Homepage: LM4857.html
* Data Sheet: LM4857.pdf
* Connects to: S3C2410 I2C (Control)

**Analog wired Headset**

There's a four-ring 2.5mm stereo jack which provides connectivity to old-fashioned wired headsets.

The headsets used by Motorola smartphones (A780,A1200, ...) and the V-360 have a compatible configuration.

Pinout: [5]

base = ground
speaker left  (internal impedance 33R) to ground. (+jackinsert detection)
speaker right (internal impedance 33R) to ground.
tip = mic electret condenser type, to ground.
bias (power for mic) 2K2 from +3.3v(wolfson codec)
(+HoldButton shortcircuit to ground)

**Bluetooth Headset**

This one is wired via PCM bus from the CSR Bluetooth chip to the Wolfson codec.
Power Management

A Philips PCF50606 is used for power management.

* Data Sheet: PCF50606/605
* User Manual: pcf50606.pdf
* Connected to: S3C2410 via I2C, client address is 0x08.
* Driver Source: https://svn.openmoko.org/trunk/src/target/kernel/patches/gta01-pcf50606.patch

**Battery**

The Neo1973 Battery is compatible with a Nokia BL5C battery. According to this post on the mailinglist. Photo of the battery inside the Neo1973.
**Buttons**

The Neo1973 features two buttons:

1. The Power Button
2. The "Aux" button

**Neo Freerunner -** The **Neo FreeRunner** (internal codename GTA02) is the second phone designed to run Openmoko software and is the direct descendant of the earlier Neo 1973.  Hardware specs ecs are at Neo Free Runner GTA 02 Hardware


**Features**

* Display- Topply o2.8, 480 x 640 pixels, VGA, 200 NIT minimum, resistance type touch

* User Interface Navigation- Touch screen on LCD, 2 control “buttons”, 1 Power button, 1 Aux for 911 emergency call

* Built-in 802.11b/g Radio (Atheros chipset AR6001 Flash version)

* Built-in Bluetooth 2.0 + EDR (CSR and support PCM audio , BC4 firmware version)

* Built-in 2D/3D graphics acceleration chip (S-Media 3362)

* 2 built-in Tri-Axis sensors (ST accelerometer LIS302DL)

* Built in GPS Radio – -130 dBm with internal antenna, -157 dBm tracking on chipset specification, TTFF under 40 seconds with -130 dBm signal strength, and tracking (u-Blox)

* Antenna – Specialized antenna for best in hand hold GPS, GPRS and Wi-Fi/Bluetooth performance are required, -105dBm on receiving, Tx 30dbm+2 on GSM

* External Antennae – MMCX GPS connector

* GPRS Radio –GSM/GPRS radio. A Pre-PTCRB certified module will be preferred

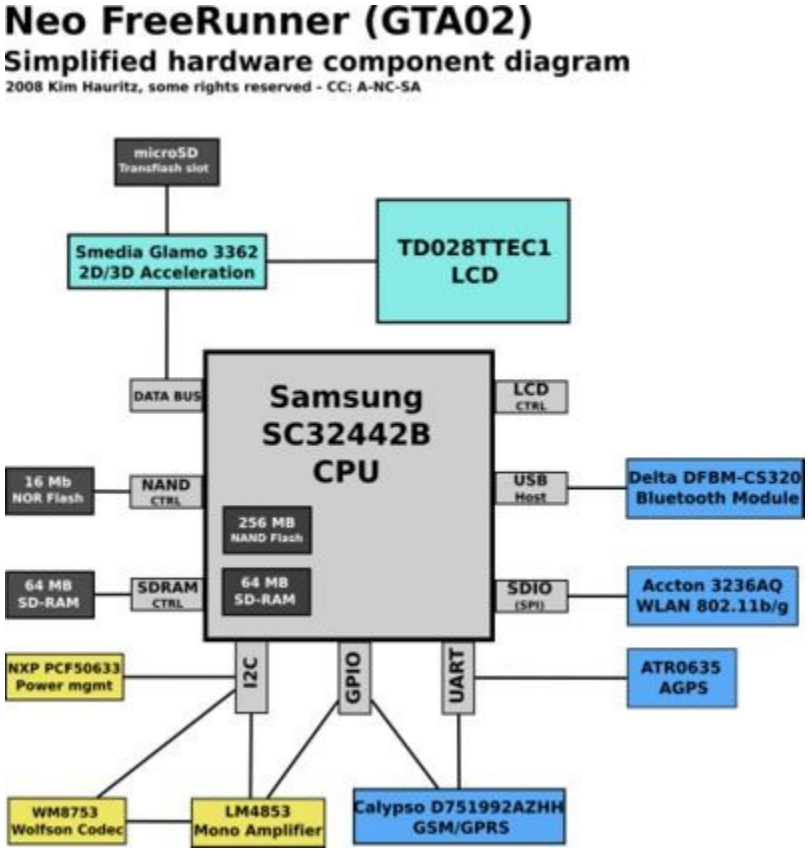
* Linux – Linux kernel 2.6.24 or later Openmoko kernel

* USB - Client and Host mode switch-able (to be used for software downloading), provide host 5V power

* Power- Normal mode power will be via 1200 mAh battery with built-in coulomb counter, could charge via specialized charger. Internal Lithium Ion or Lithium Polymer battery will keep device in standby mode. Battery life (Approximation/Ideal Target) Standby time 150-200 Hrs (GSM) Talk time (Backlight off) Up to 3-4 hrs(GSM)

* LED- LED indicator under Aux/Power button key

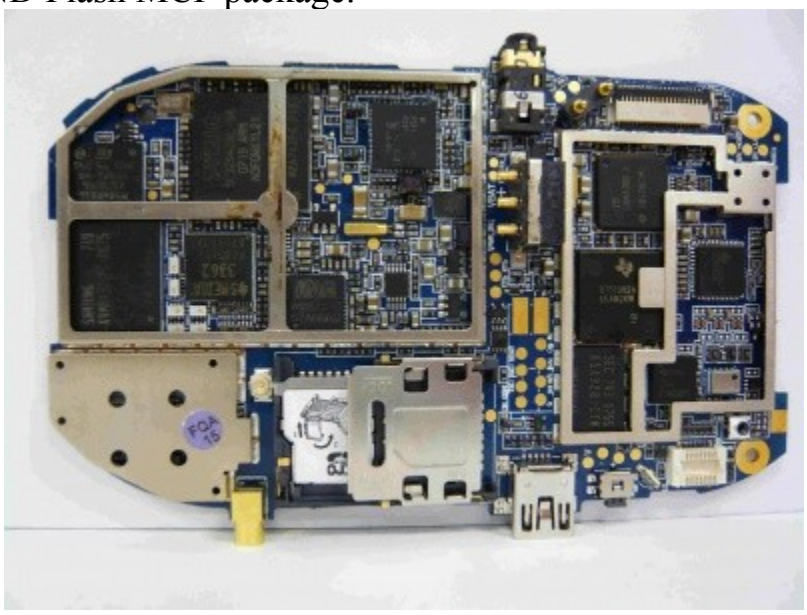
## Hardware Specification




Hardware Electrical

Simplified hardware spec diagram

* 400/500 MHz Samsung 2442B Processor/SOC (400 minimum, ARM920T core, ARMv4T)
* Boot code in NAND FLASH or 2MB NOR FLASH (optional design)
* 128 MB SDRAM total, 64 MB CPU internal, 64 MB external
* 256MB NAND Flash MCP package.



display (top) side NOTE: GTA02 A5 PCBA Component Side photo

**Display**

* Topploy VGA ; 2.8” diagonal, 480 x 640 pixels, 16 bit color depth
* Transmissive display: good readability in high ambient light is essential
* White LED backlight. Required brightness is 200 NIT minimum.
* Resistance type touch panel.

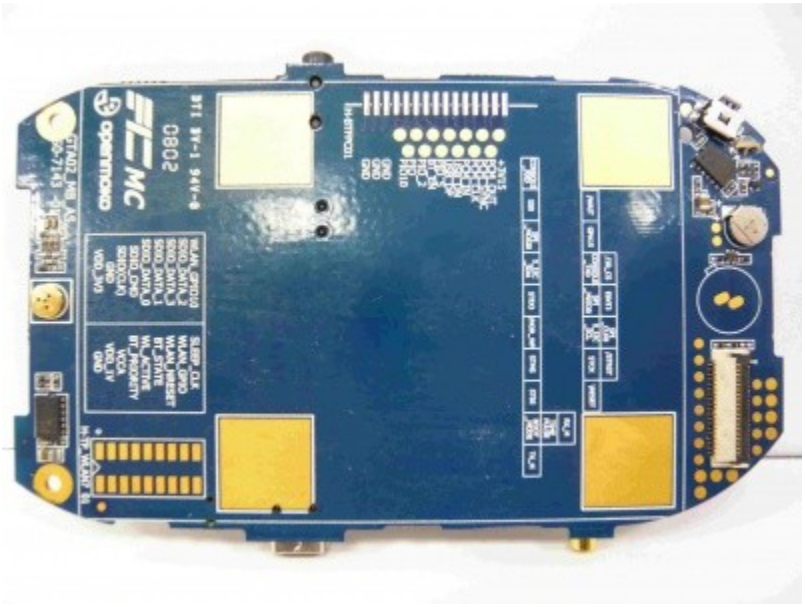
**WiFi 802.11 b/g transceiver**

* Must have GPL support source or GPL compatible policy
* TX power at 11 Mbps: 13 dBm minimum
* RX sensitivity at 11 Mbps: -89 dBm desired, -83 dBm minimum
* AP mode desirable, not required
* WEP and WPA supported
* Atheros preferred because it's GPL policy

**Serial interfaces (UART)**

* Three serial interfaces are required
* Console

* A-GPS or GPS
* GSM/GPRS



component (back) side NOTE: GTA02 A5 PCBA Print Side photo

**Accelerometer**

* 2x accelerometer required
* Could support interrupt while suspend or power save mode
* 3 axis sensing

**A-GPS**

* GPS chipset receiver and antenna
* Sensitivity at Antenna port: -157 dBm tracking on chipset specification
* LNA and SAW filter for maximum interference protection
* Cold start time to first fix: 40 sec typical at -130 dBm, 60 sec max
* Must support GPL for Assist-GPS function with open API
* Industry quality GPS
* Could fit in GTA01 GPS area on the PCB

**GPS Antenna Performance**

* Antenna is passive and internal; 15 mm x 15 mm ceramic patch is nominal design
* Antenna LNA and SAW filter are required to meet GPS performance
* 15 mm square ground plane (minimum 1 mm ground border around patch) (TBA)
* There will be one external GPS antenna connector (MMCX)
* C/N ratio should higher than 35 on production testing

**Buttons**

* Touch screen over LCD is primary data entry mechanism
* Two “hard” buttons: Power button (on side of Neo1973) is a mechanical switch actuated by a plastic pushbutton in a hole in the housing. Aux (911) button on the top of the device, All two of these buttons, when pushed by the operator, are binary inputs (on/off or pressed/not pressed) to the software. The effect of each button is determined by the application software in the device
* Buttons may need to be backlit
* 50000 cycles on hardware specification

**Sound outputs**

* Speaker in box (need good volume and acoustic behavior in noisy environments)
* Audio is monophonic*Max volume: 100 dB at 5 cm to assure good performance in environment.
* Support earphone with mic by jack

**Power Design Requirements**

* Software based power management unit preferred
* NXP PCF series preferred
* Need support charge from USB function
* Need support powered by USB function
* Power switch: Neo1973 will have a power switch, for power on/off and suspend
* Power/Aux switch must be backlit
* Switch controls whether device is running or suspended by presses of the switch
* Switch does not shut off the power; it only suspends/resumes the device
* Internal Li-Ion or Li-Polymer battery is included. This battery supplies standby power to the device eliminates the rebooting of the device when local power is again reapplied. Battery is 1200 ma-hr.
* Battery life (Approximation) Ideal/Target Standby time 150-200 Hrs (GSM) Talk time (Backlight off) Up to 4 hrs(GSM)
* Estimated current draw for the entire device when in suspend mode (and ALL peripherals are turned off or set for deep sleep) is <5 mA at 3.6 volts (Li-Ion terminal voltage).
* GSM module deep sleep(alive and keep contact with base station) stage should take less than 8mA
* Battery will reach half capacity (~600 mAh) with 500 charge-discharge cycles. This will occur in less than 2 years of daily service.
* When powered continuously, Neo1973 must suspend (to low power mode) based either on observed low battery voltage condition or a configurable time delay.
* Neo1973 must monitor battery status while suspended and resume automatically if

the charger is inserted.
* Primary power connection: 1200mAh battery
* USB charger have ID pin 47.5k pull down for Openmoko identification
* Indicators: an LED indicator visible from the side of the unit will illuminate when charging or have missing incoming call

**GSM/GPRS**

* 850/1800/1900 and 900/1800/1900 MHz bands must be supported
* Design should allow for multi-band version (850/900 MHz)
* Module based GPRS transceiver could meeting PTCRB and appropriate FCC certifications. It preferred that the module be pre-certified with PTCRB or OTA test
* FCC/CE certification required for GSM/GPRS part

**GSM-GPRS Antenna Performance**

* -105 dBm receiving on each channel (GSM/PCS)
* 30+2 dBm transmission on GSM channel

**Wi-Fi Modules**

* Must support GPL driver
* Atheros AR6k preferred
* Flash version required

**Wi-Fi Antenna Performance**

* The Wi-Fi antenna with TX 13 to 15 dBm
* RX -89 to -83 dBm @802.11b 11Mbps or an equivalent performance antenna

**Bluetooth**

* CSR BC4 or later solutions

**USB**

* Neo FreeRunner GTA02 will have USB, client/host. Using USB 1.1
* Provides USB host 5v power
* Could be powered by USB

**Microphone**

1 microphone is in the device
Firmware Image

* Using Linux 2.6.24 or later
* Could support boot from NAND or Boot from NOR
* Shipping image should come with basic phone function
* Could do full firmware upgrade by USB cable

**PSN**

* Device will have a PSN (product serial number) printed on the product label and machine readable in system NAND memory

**IMEI**

* Production phase should have IMEI code written

The device is being sold through multiple channels (including Openmoko.com), the official price has been set to 399$ but it could slightly vary in some countries (due to taxes).

## Why Openmoko is best of all ?

This article discusses the pros and cons of various operating systems for the mobile platform, and tells us why Openmoko is the true 'open' platform. There are many mobile platforms that exist on this planet today. The situation is very similar to the desktop segment, where we have operating systems like Linux, Windows and a few others. To a user, the fact that Linux is an open source model makes it a much better OS. We also have great mobile operating systems like Symbian, Windows Mobile Platform and Linux. But none of these are really 'open'. Surprised to see Linux in this list? Linux as a mobile OS comes in different shapes and sizes that are not very compatible with each other, as there are many different implementations of the platform. There are no common APIs available for developers to build on top of. Examples include implementations from Motorola and MontaVista. At least, this is not the case with proprietary OSs such as Symbian and Windows Mobile. They provide abundant APIs to aid third-party programmers to build on their platform. Let us now set aside all this for a while and concentrate on a Linux-based mobile platform project called Openmoko—a project that combines the true essence of open source with a rich application-programming interface. Openmoko is a true conversion of the desktop Linux into mobile Linux, which doesn't bring in any non-standard or hidden aspects. It was built using standard X-server, GTK, ALSA, Dbus and so on. The beauty is that even the hardware

platform is as open as the software platform. Openmoko makes use of a standard platform to make the mobile environment complete. Even the circuit-board hackers will find Openmoko a dream come true!