# Android Open Source Project - Building

## Where to find out about Android Open Source Project

https://source.android.com/

## Download the Source

https://source.android.com/setup/downloading

```
sudo apt-get install curl
curl https://storage.googleapis.com/git-repo-downloads/repo > ~/bin/repo
chmod a+x ~/bin/repo
mkdir repo
cd repo
sudo apt-get install git
git config --global user.name "John Mehan"
git config --global user.email "john.mehan@irdeto.com"
repo init -u https://android.googlesource.com/platform/manifest
repo sync
```

## Getting a Branch

If you want to use a particular branch, checkout the list of branches at: https://source.android.com/setup/build-numbers

```
repo init -u https://android.googlesource.com/platform/manifest -b android-8.1.0_r1
repo sync
```

## Building

See https://source.android.com/setup/building

#### Clean up

```
make clobber
```

#### Setup Environment

```
source build/envsetup.sh
```

#### Using HiKey 960

See https://source.android.com/setup/devices

From the root folder:

```
wget https://dl.google.com/dl/android/aosp/arm-hikey960-OPR-cf4e0c80.tgz
tar xzf arm-hikey960-OPR-cf4e0c80.tgz
./extract-arm-hikey960.sh
```

**Add Car Packages**

```
cd device/linaro/hikey

vi hikey960.mk
```

Update the hikey960.mk:

```
ifndef TARGET_KERNEL_USE

TARGET_KERNEL_USE=4.9

endif

TARGET_PREBUILT_KERNEL := device/linaro/hikey-kernel/Image.gz-hikey960-$(TARGET_KERNEL_USE)

TARGET_PREBUILT_DTB := device/linaro/hikey-kernel/hi3660-hikey960.dtb-$(TARGET_KERNEL_USE)


ifeq ($(TARGET_KERNEL_USE), 4.4)

  HIKEY_USE_LEGACY_TI_BLUETOOTH := true

else

  HIKEY_USE_LEGACY_TI_BLUETOOTH := false

endif


#

# Inherit the full_base and device configurations

$(call inherit-product, device/linaro/hikey/hikey960/device-hikey960.mk)

$(call inherit-product, device/linaro/hikey/device-common.mk)

$(call inherit-product, $(SRC_TARGET_DIR)/product/full_base.mk)


#

# Overrides

PRODUCT_NAME := hikey960

PRODUCT_DEVICE := hikey960

PRODUCT_BRAND := Android

PRODUCT_MODEL := AOSP on hikey960


BOARD_IS_AUTOMOTIVE := 1
```

```
PRODUCT_PACKAGES += \
    vehicle.default \
    android.car \
    car-service-lib-for-test \
    libvehiclenetwork-native \
    libvehiclenetwork-java \
    libvehiclenetwork-audio-helper \
    libvehiclenetworkproto-native \
    vehicle_network_service \
    CarServiceTest \
    vns_policy.xml \
    vehicle_tests \
    candump \
    cansend \
    cansniffer \
    android.car.cluster.loggingrenderer

PRODUCT_PACKAGE_OVERLAYS := packages/services/Car/car_product/overlay

PRODUCT_COPY_FILES += \
    device/generic/car/common/android.hardware.dummy.xml:system/etc/permissions/handheld_core_hardware.xml \
    packages/services/Car/car_product/init/init.car.rc:root/init.car.rc \
    packages/services/Car/car_product/init/init.bootstat.rc:root/init.bootstat.rc

PRODUCT_COPY_FILES += \
    frameworks/native/data/etc/android.hardware.type.automotive.xml:system/etc/permissions/android.hardware.type.automotive.xml \
    frameworks/native/data/etc/android.hardware.screen.landscape.xml:system/etc/permissions/android.hardware.screen.landscape.xml

PRODUCT_PROPERTY_OVERRIDES += \
    android.car.drawer.unlimited=true \
    android.car.hvac.demo=true \
    com.android.car.radio.demo=true \
    com.android.car.radio.demo.dual=true
```

**Build**

lunch <BUILD-BUILDTYPE>

All build targets take the form `BUILD-BUILDTYPE`, where the `BUILD` is a codename referring to the particular feature combination.

| BuiltType | Use |
| --- | --- |
| user | limited access; suited for production |
| userdebug | like "user" but with root access and debuggability; preferred for debugging |
| eng | development configuration with additional debugging tools |

For more information about building for and running on actual hardware, see Running Builds.

```
> lunch

Lunch menu... pick a combo:
    1. aosp_arm-eng
    2. aosp_arm64-eng
    3. aosp_mips-eng
    4. aosp_mips64-eng
    5. aosp_x86-eng
    6. aosp_x86_64-eng
    7. full_fugu-userdebug
    8. aosp_fugu-userdebug
    9. car_emu_arm64-userdebug
    10. aosp_car_emu_arm64-userdebug
    11. car_emu_arm-userdebug
    12. aosp_car_emu_arm-userdebug
    13. car_emu_x86_64-userdebug
    14. aosp_car_emu_x86_64-userdebug
    15. aosp_car_emu_x86-userdebug
    16. car_emu_x86-userdebug
    17. mini_emulator_arm64-userdebug
    18. m_e_arm-userdebug
    19. m_e_mips64-eng
    20. m_e_mips-userdebug
    21. mini_emulator_x86_64-userdebug
    22. mini_emulator_x86-userdebug
    23. uml-userdebug
    24. aosp_dragon-userdebug
    25. aosp_dragon-eng
    26. aosp_marlin-userdebug
    27. aosp_marlin_svelte-userdebug
    28. aosp_sailfish-userdebug
```

> *29. aosp_muskie-userdebug*
>
> *30. aosp_walleye-userdebug*
>
> *31. aosp_walleye_test-userdebug*
>
> *32. aosp_taimen-userdebug*
>
> *33. aosp_angler-userdebug*
>
> *34. aosp_bullhead-userdebug*
>
> *35. aosp_bullhead_svelte-userdebug*
>
> *36. hikey-userdebug*
>
> *37. hikey960-userdebug*

**Make**

> *make -j4*

**Connect to the board**

Connect the USB-C to the HiKey-960 board and power it up.

> *adb kill-server*
>
> *adb start-server*
>
> *adb shell*

You should be able to get a shell prompt.

If you get the following error, you may need to install an additional package

***error: insufficient permissions for device: user in plugdev group; are your udev rules wrong?***

***See [http://developer.android.com/tools/device.html] for more information***

> *sudo apt-get install android-sdk-platform-tools-common*

This package includes the udev rules to make adb work properly.

**Flash device**

Select fastboot mode turning ON switch 1 and 3 (for details, refer to the HiKey960 user guide).

Power the board.

Flash initial images:

```
> cd device/linaro/hikey/installer/hikey960
> ./flash-all.sh
```

**Output**

```
android out dir:/home/test/projects/aosp/out/target/product/hikey960
target reported max download size of 471859200 bytes
sending 'xloader' (151 KB)...
OKAY [  0.026s]
writing 'xloader'...
FAILED (remote: flash write back vrl failure)
finished. total time: 0.276s
target reported max download size of 471859200 bytes
sending 'ptable' (196 KB)...
OKAY [  0.027s]
writing 'ptable'...
OKAY [  0.030s]
finished. total time: 0.056s
target reported max download size of 471859200 bytes
sending 'fastboot' (3346 KB)...
OKAY [  0.099s]
writing 'fastboot'...
OKAY [  0.043s]
finished. total time: 0.142s
rebooting into bootloader...
OKAY [  0.020s]
finished. total time: 0.070s
< waiting for any device >
target reported max download size of 471859200 bytes
sending 'nvme' (128 KB)...
OKAY [  0.030s]
writing 'nvme'...
OKAY [  0.071s]
finished. total time: 0.101s
target reported max download size of 471859200 bytes
sending 'fw_lpm3' (212 KB)...
OKAY [  0.013s]
writing 'fw_lpm3'...
OKAY [  0.021s]
finished. total time: 0.034s
target reported max download size of 471859200 bytes
sending 'trustfirmware' (145 KB)...
OKAY [  0.030s]
writing 'trustfirmware'...
OKAY [  0.021s]
finished. total time: 0.051s
target reported max download size of 471859200 bytes
sending 'boot' (9872 KB)...
OKAY [  0.247s]
writing 'boot'...
OKAY [  0.079s]
finished. total time: 0.326s
target reported max download size of 471859200 bytes
sending 'dts' (14 KB)...
OKAY [  0.022s]
writing 'dts'...
OKAY [  0.017s]
finished. total time: 0.039s
target reported max download size of 471859200 bytes
sending sparse 'system' 1/3 (460796 KB)...
OKAY [ 10.968s]
writing 'system' 1/3...
OKAY [  4.139s]
sending sparse 'system' 2/3 (421196 KB)...
OKAY [ 10.042s]
writing 'system' 2/3...
OKAY [  3.041s]
sending sparse 'system' 3/3 (134240 KB)...
OKAY [  3.210s]
writing 'system' 3/3...
OKAY [  0.983s]
finished. total time: 32.383s
target reported max download size of 471859200 bytes
```

```
sending 'cache' (56 KB)...
OKAY [  0.027s]
writing 'cache'...
OKAY [  0.074s]
finished. total time: 0.101s
target reported max download size of 471859200 bytes
sending 'userdata' (4880 KB)...
OKAY [  0.139s]
writing 'userdata'...
OKAY [  3.402s]
finished. total time: 3.541s
```

# References

| | |
|---|---|
| Andriod Open Source Project | https://source.android.com/ |
| Downloading AOSP | https://source.android.com/setup/downloading |
| Building AOSP | https://source.android.com/setup/building |
| Using Repo | https://source.android.com/setup/using-repo |
| AOSP Releases/Branches | https://source.android.com/setup/build-numbers |