

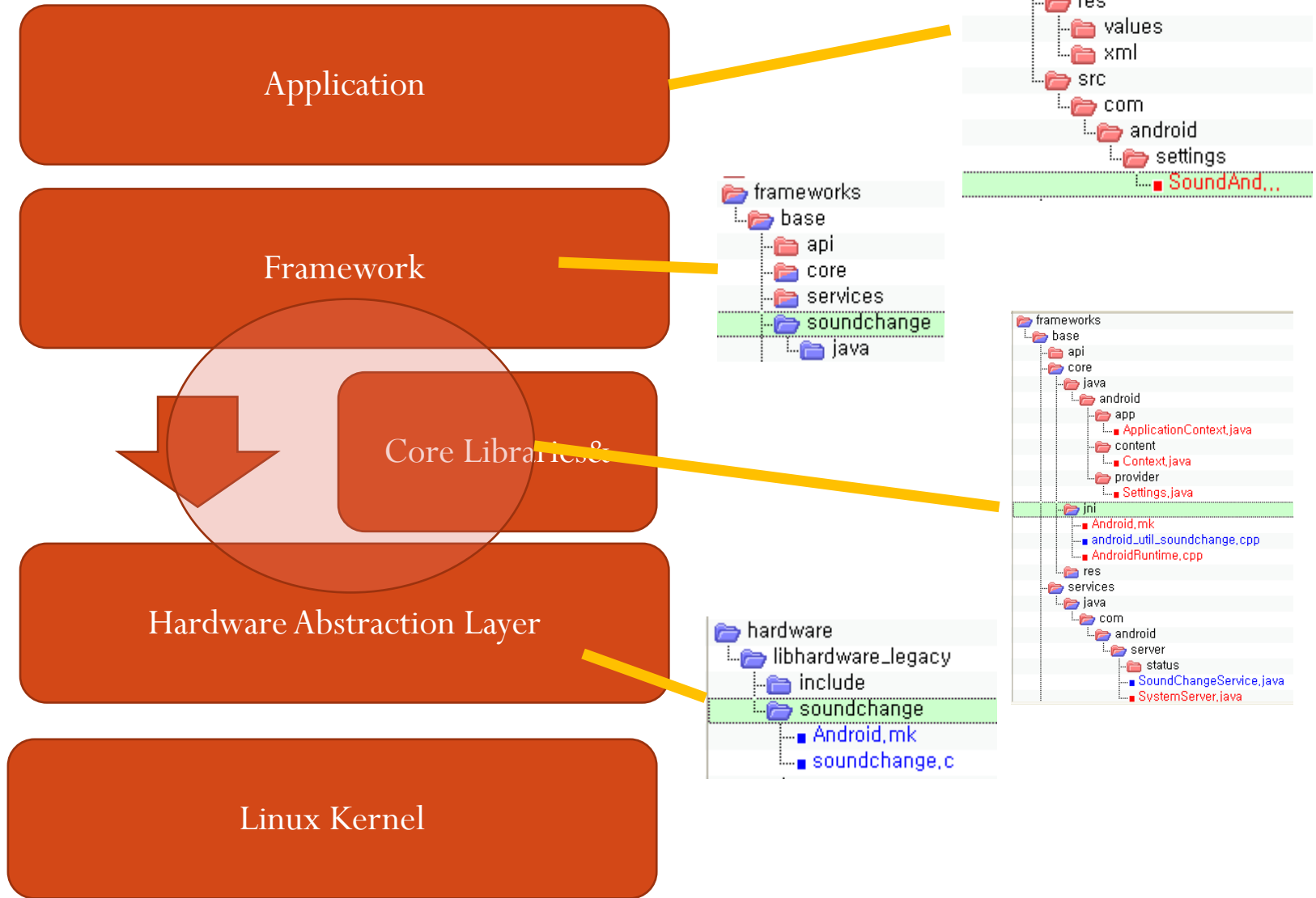
망고100 보드로 놀아보자-18

Android 메뉴 생성 및 아이콘 생성
(SPDIF,WM8960)

<http://cafe.naver.com/embeddedcrazyboys>

<http://www.mangoboard.com>

Android 구조



Android 메뉴 생성(커널 수정) (WM8960, SPDIF 드라이버 포함)

sound\soc\s3c24xx\Kconfig 파일에 아래와 같이 수정

```
config SND_MANGO100_WM8960
    bool "WM8960 Driver"
    depends on SND_S5P_MANGO100
    select SND_SOC_WM8960
    select SND_S5P_SOC_I2S
    select SND_S3C_I2SV2_SOC

config SND_MANGO100_HDMI_SPDIF
    bool "HDMI SPDIF Driver"
    depends on SND_S5P_MANGO100
    select SND_S5P_SPDIF
```

```
static struct snd_soc_dai_link mango_dai[] = {
{
    .name = "WM8960 I2S",
    .stream_name = "Tx/Rx",
    .cpu_dai = &s5p_i2s_dai[0],
    .codec_dai = &wm8960_dai,
    .init = mango_wm8960_init,
    .ops = &mango_i2s_ops,
},
};

static struct snd_soc_card mango = {
    .name = "mango",
    .platform = &s3c24xx_soc_platform,
    .dai_link = mango_dai,
    .num_links = ARRAY_SIZE(mango_dai),
};

static struct wm8960_setup_data mango_wm8960_setup = {
    .i2c_bus = 0,
    .i2c_address = 0x1a,
};
```

```
static struct snd_soc_dai_link mango100_dai[] = {
{
    .name = "HDMI-SPDIF",
    .stream_name = "HDMI-SPDIF Playback",
    .cpu_dai = &s5p_spdif_dai,
    .codec_dai = &s5p_hdmi_spdif_dai[0],
    .init = mango100_spdif_init,
    .ops = &mango100_spdif_ops,
},
};

static struct snd_soc_card mango100 = {
    .name = "mango100",
    .platform = &s3c24xx_soc_platform,
    .dai_link = mango100_dai,
    .num_links = ARRAY_SIZE(mango100_dai),
};
```

Android 메뉴 생성(커널 수정) (WM8960,SPDIF 드라이버 포함)

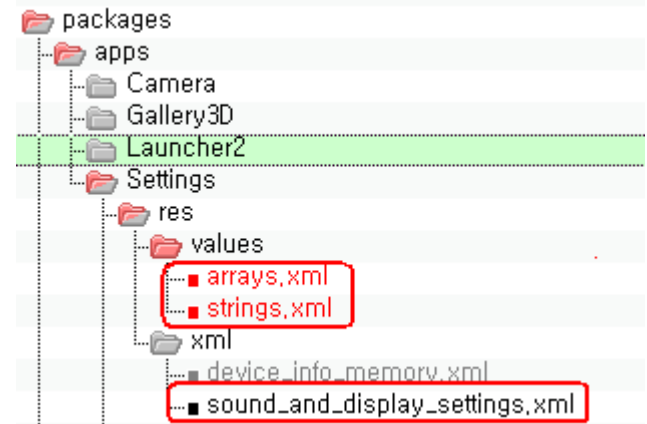
```
# ls -al /proc/asound/
dr-xr-xr-x  5 0      0      0 Apr 29 15:06 .
dr-xr-xr-x 64 0      0      0 Jan  1  1970 ..
dr-xr-xr-x  4 0      0      0 Apr 29 15:06 card0
dr-xr-xr-x  4 0      0      0 Apr 29 15:06 card1
-r--r--r--  1 0      0      0 Apr 29 15:06 cards
-r--r--r--  1 0      0      0 Apr 29 15:06 devices
lrwxrwxrwx  1 0      0      0 Apr 29 15:06 mango -> card0
lrwxrwxrwx  1 0      0      0 Apr 29 15:06 mango100 -> card1
dr-xr-xr-x  2 0      0      0 Apr 29 15:06 oss
-r--r--r--  1 0      0      0 Apr 29 15:06 pcm
-r--r--r--  1 0      0      0 Apr 29 15:06 timers
-r--r--r--  1 0      0      0 Apr 29 15:06 version

# ls /proc/asound/card0/
id      oss_mixer  pcm0c      pcm0p
# ls /proc/asound/card1
id      oss_mixer  pcm0c      pcm0p
# cat /proc/asound/card0/id
mango
# cat /proc/asound/devices
2:          : timer
3: [ 0- 0]: digital audio playback
4: [ 0- 0]: digital audio capture
5: [ 0]   : control
6: [ 1- 0]: digital audio playback
7: [ 1- 0]: digital audio capture
8: [ 1]   : control
# cat /proc/asound/cards
0 [mango          ]: WM8960 - mango
                        mango (WM8960)
1 [mango100       ]: HDMI-SPDIF - mango100
                        mango100 (HDMI-SPDIF)
```

Android 메뉴 생성

packages\apps\Settings\res\values\arrays.xml

```
<!-- Sound Change Setting. -->
<string-array name="sound_change_entries">
  <item>WM8960</item>
  <item>SPDIF</item>
</string-array>
<!-- Do not translate. -->
<string-array name="sound_change_values" translatable="false">
  <item>10</item>
  <item>20</item>
</string-array>
```



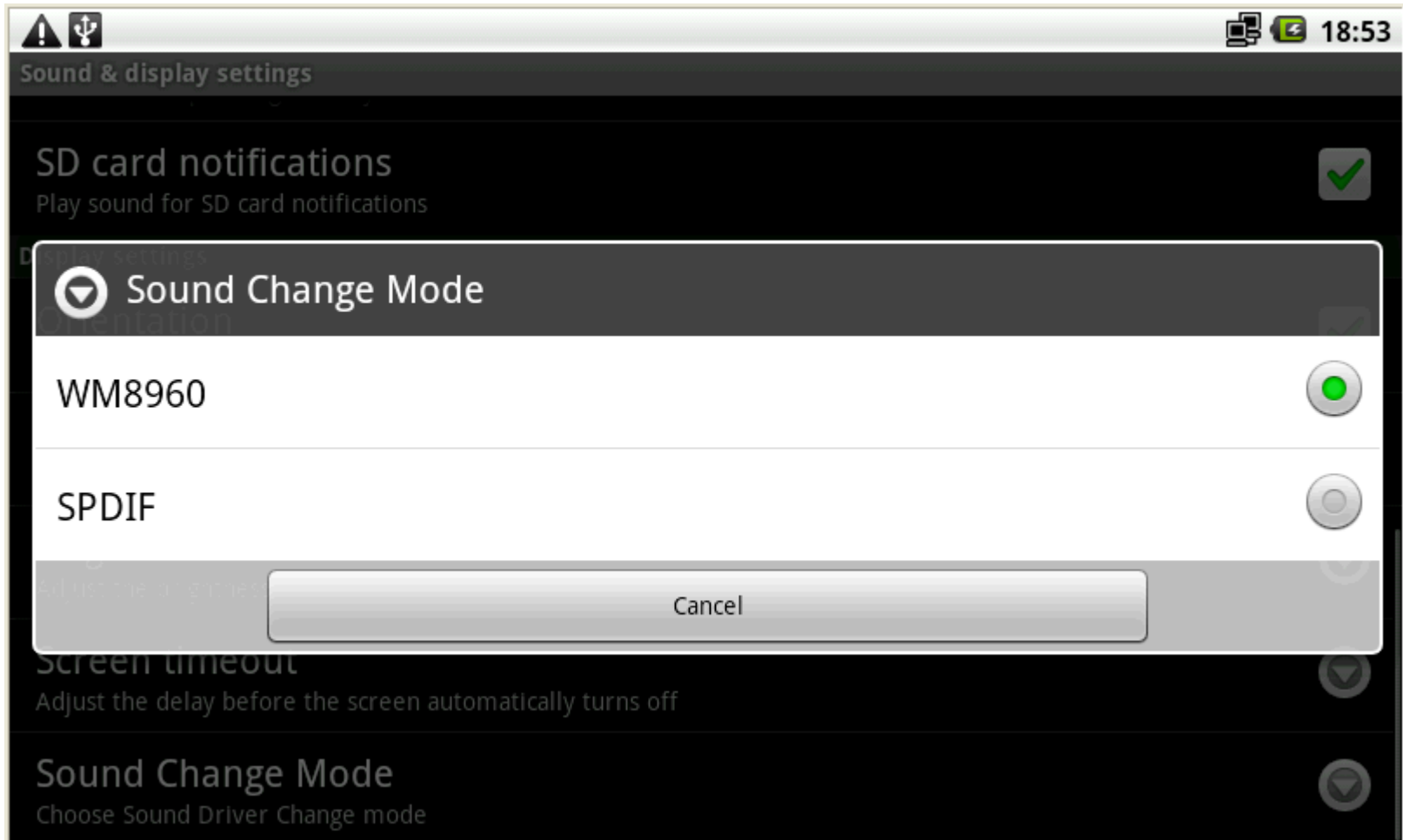
packages\apps\Settings\res\values\strings.xml

```
<string name="sound_change">Sound Change Mode</string>
<!-- Sound & display settings screen, setting option summary to change Sound Driver -->
<string name="sound_change_summary">Choose Sound Driver Change mode</string>
<!-- Sound & display settings screen, setting option name to change Sound Driver Mode -->
```

packages\apps\Settings\res\xml\sound_and_display_settings.xml

```
<ListPreference
  android:key="sound_change"
  android:title="@string/sound_change"
  android:summary="@string/sound_change_summary"
  android:persistent="false"
  android:entries="@array/sound_change_entries"
  android:entryValues="@array/sound_change_values" />
```

Android 메뉴 생성



Android 메뉴 생성

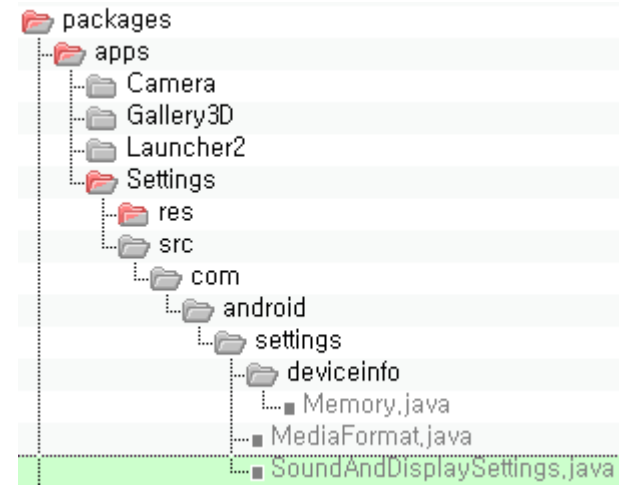
SoundAndDisplaySettings.java

```
import static android.provider.Settings.System.SOUND_CHANGE;
import android.util.soundchange.SoundChangeManager;
public class SoundAndDisplaySettings extends PreferenceActivity implements
    Preference.OnPreferenceChangeListener {
    private static final String TAG = "SoundAndDisplaySettings";

    /** If there is no setting in the provider, use this. */
    private static final int FALLBACK_SCREEN_TIMEOUT_VALUE = 30000;
    private static final int FALLBACK_SOUND_CHANGE_VALUE = 10;
    private static final int FALLBACK_EMERGENCY_TONE_VALUE = 0;

    private static final String KEY_SILENT = "silent";
    private static final String KEY_VIBRATE = "vibrate";
    private static final String KEY_SCREEN_TIMEOUT = "screen_timeout";
    private static final String KEY_SOUND_CHANGE = "sound_change";
```

```
private SoundChangeManager mSoundChangeManager;
ListPreference SoundChangePreference =
    (ListPreference) findPreference(KEY_SOUND_CHANGE);
    SoundChangePreference.setValue(String.valueOf(Settings.System.getInt(
        resolver, SOUND_CHANGE, FALLBACK_SOUND_CHANGE_VALUE)));
    SoundChangePreference.setOnPreferenceChangeListener(this);
        mSoundChangeManager = (SoundChangeManager) getSystemService(SOUNDCHANGE_SERVICE);
./packages/apps/Settings/src/com/android/settings/SoundAndDisplaySettings.java
```



```
public static final String SOUND_CHANGE = "sound_change";
./frameworks/base/core/java/android/provider/Settings.java
```

Android 메뉴 생성

SoundAndDisplaySettings.java

```
import android.util.soundchange.SoundChangeManager;
```

```
./packages/apps/Settings/src/com/android/settings/SoundAndDisplaySettings.java
```

SoundChangeManager.java, ISoundChangeManager.aidl, SoundChangeNative.java

소스 추가

```
package android.util.soundchange;
```

```
import java.util.List;
```

```
import android.annotation.SdkConstant;
```

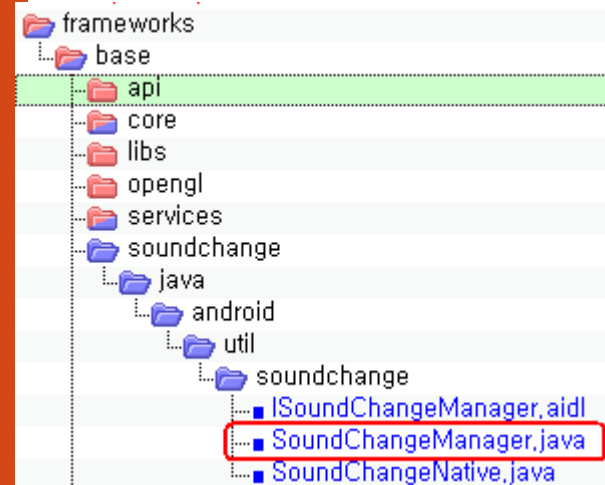
```
import android.annotation.SdkConstant.SdkConstantType;
```

```
import android.os.Handler;
```

```
import android.os.RemoteException;
```

```
import android.util.Log;
```

```
public class SoundChangeManager {
```



Android 메뉴 생성

SoundAndDisplaySettings.java

```
private SoundChangeManager mSoundChangeManager;
ListPreference SoundChangePreference =
    (ListPreference) findPreference(KEY_SOUND_CHANGE);
SoundChangePreference.setValue(String.valueOf(Settings.System.getInt(
    resolver, SOUND_CHANGE, FALLBACK_SOUND_CHANGE_VALUE)));
SoundChangePreference.setOnPreferenceChangeListener(this);
    mSoundChangeManager = (SoundChangeManager) getSystemService(SOUNDCHANGE_SERVICE);
./packages/apps/Settings/src/com/android/settings/SoundAndDisplaySettings.java
```

```
public Object getSystemService(String name) {
    ..
} else if (SOUNDCHANGE_SERVICE.equals(name)) {
    return getSoundChangeManager();
}
./frameworks/base/core/java/android/app/ApplicationContext.java
```

```
private SoundChangeManager getSoundChangeManager()
{
    synchronized (sSync) {
        if (sSoundChangeManager == null) {
            IBinder b = ServiceManager.getService(SOUNDCHANGE_SERVICE);
            ISoundChangeManager service = ISoundChangeManager.Stub.asInterface(b);
            sSoundChangeManager = new SoundChangeManager(service);
        }
    }
    return sSoundChangeManager;
}
./frameworks/base/core/java/android/app/ApplicationContext.java
```

Android 메뉴 생성

SoundAndDisplaySettings.java

APP

- **mSoundChangeManager = (SoundChangeManager) getSystemService(SOUNDCHANGE_SERVICE);**
- **./packages/apps/Settings/src/com/android/settings/SoundAndDisplaySettings.java**

FRAMEWORK

- **public Object getSystemService(String name) {..**
- **./frameworks/base/core/java/android/app/ApplicationContext.java**

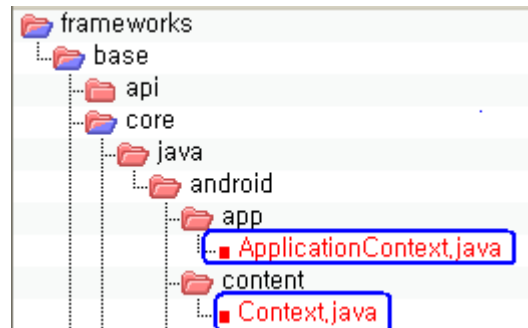
FRAMEWORK

- **private SoundChangeManager getSoundChangeManager(){**
- **}./frameworks/base/core/java/android/app/ApplicationContext.java**

Android Framework

```
public Object getSystemService(String name) {  
    ..  
} else if (SOUNDCHANGE_SERVICE.equals(name)) {  
    return getSoundChangeManager();  
}   
./frameworks/base/core/java/android/app/ApplicationContext.java
```

```
public static final String SOUNDCHANGE_SERVICE = "soundchange";  
./frameworks/base/core/java/android/content/Context.java
```



Android Framework

```
private SoundChangeManager getSoundChangeManager()
{
    synchronized (sSync) {
        if (sSoundChangeManager == null) {
            IBinder b = ServiceManager.getService(SOUNDCHANGE_SERVICE);
            ISoundChangeManager service = ISoundChangeManager.Stub.asInterface(b);
            sSoundChangeManager = new SoundChangeManager(service);
        }
    }
    return sSoundChangeManager;
}
./frameworks/base/core/java/android/app/ApplicationContext.java
```

```
soundchange = new SoundChangeService(context);
ServiceManager.addService(Context.SOUNDCHANGE_SERVICE, soundchange);
./frameworks/base/services/java/com/android/server/SystemServer.java
```

```
package com.android.server;
public class SoundChangeService<synchronized> extends ISoundChangeManager.Stub {
    private static final String TAG = "SoundChangeService";

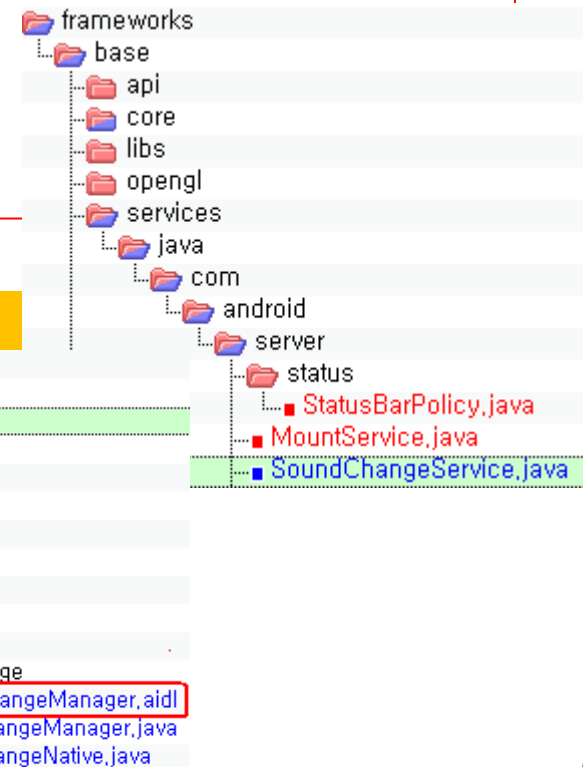
    public SoundChangeService(Context context) {
        mContext = context;
        mPreSoundChangeState = getPersistedState();
        setSoundChangeMode(mPreSoundChangeState);
    }
}
frameworks\base\services\java\com\android\server\SoundChangeService.java
```

Android Framework(Binder)

```
private SoundChangeManager getSoundChangeManager()
{
    synchronized (sSync) {
        if (sSoundChangeManager == null) {
            IBinder b = ServiceManager.getService(SOUNDCHANGE_SERVICE);
            ISoundChangeManager service = ISoundChangeManager.Stub.asInterface(b);
            sSoundChangeManager = new SoundChangeManager(service);
        }
    }
    return sSoundChangeManager;
}
./frameworks/base/core/java/android/app/ApplicationContext.java
```

```
public class SoundChangeService<synchronized> extends ISoundChangeManager.Stub {
```

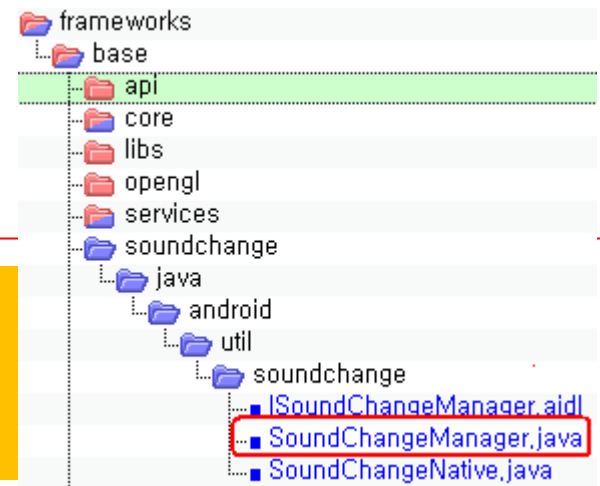
```
import android.util.soundchange.ISoundChangeManager;
./frameworks/base/core/java/android/app/ApplicationContext.java
```



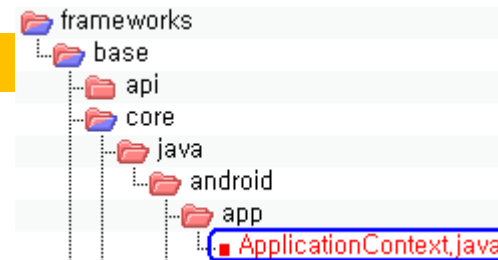
Android Framework(Binder)

```
private SoundChangeManager getSoundChangeManager()
{
    synchronized (sSync) {
        if (sSoundChangeManager == null) {
            IBinder b = ServiceManager.getService(SOUNDCHANGE_SERVICE);
            ISoundChangeManager service = ISoundChangeManager.Stub.asInterface(b);
            sSoundChangeManager = new SoundChangeManager(service);
        }
    }
    return sSoundChangeManager;
}
./frameworks/base/core/java/android/app/ApplicationContext.java
```

```
ISoundChangeManager mService;
public SoundChangeManager(ISoundChangeManager service) {
    Log.i(TAG, "Init SoundChange Manager");
    mService = service;
}
```



```
import android.util.soundchange.SoundChangeManager;
```



Android 메뉴 생성

SoundAndDisplaySettings.java

```
public boolean onPreferenceChange(Preference preference, Object objValue) {  
    ...  
} else if (KEY_SOUND_CHANGE.equals(key)) {  
    int value = Integer.parseInt((String) objValue);  
    try {  
        Settings.System.putInt(getContentResolver(),  
                                SOUND_CHANGE_MODE, value);  
    } catch (NumberFormatException e) {  
        Log.e(TAG, "could not set sound change mode setting", e);  
    }  
    // update the sound change mode  
    mSoundChangeManager.setSoundChangeMode(value);  
}
```

• packages\apps\Settings\src\com\android\settings\SoundAndDisplaySettings.java

```
private static final String KEY_SOUND_CHANGE = "sound_change";  
• packages\apps\Settings\src\com\android\settings\SoundAndDisplaySettings.java
```

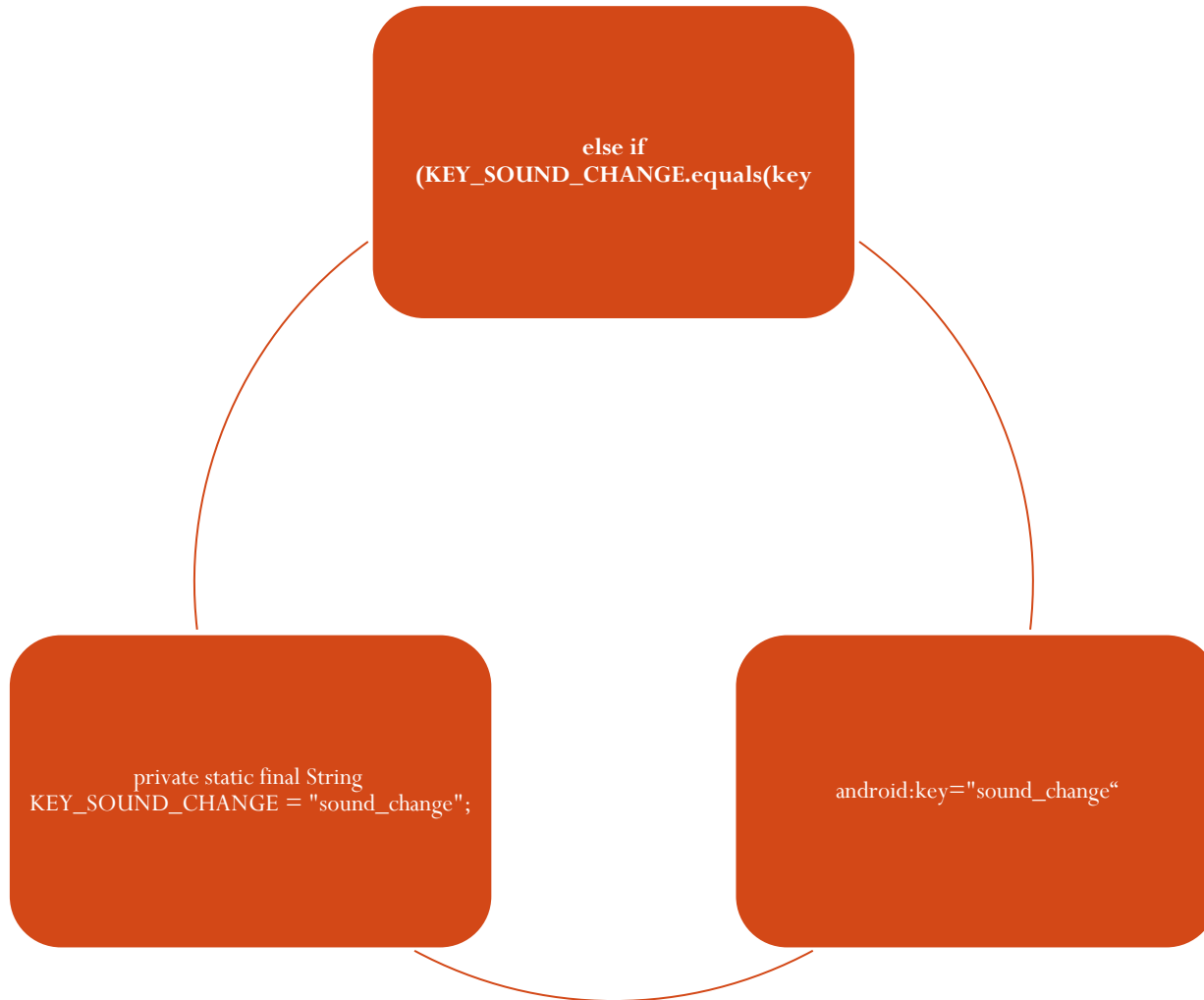
<ListPreference

android:key="sound_change"

packages\apps\Settings\res\xml\sound_and_display_settings.xml

Android 메뉴 생성

SoundAndDisplaySettings.java



Android 메뉴 생성 (APP~JNI)

```
./packages/apps/Settings/src/com/android/settings/SoundAndDisplaySettings.java
```

```
mSoundChangeManager.setSoundChangeMode(value);
```



```
./frameworks/base/services/java/com/android/server/SoundChangeService.java
```

```
setSoundChangeMode(mPreSoundChangeState);
```

```
SoundChangeNative.SoundDriverChange(mode);
```



```
./frameworks/base/soundchange/java/android/util/soundchange/SoundChangeNative.java
```

```
public native static void SoundDriverChange(int mode);}
```

Android Framework(JNI)

service

- `SoundChangeNative.SoundDriverChange(mode);`
- `./frameworks/base/services/java/com/android/server/SoundChangeService.java`

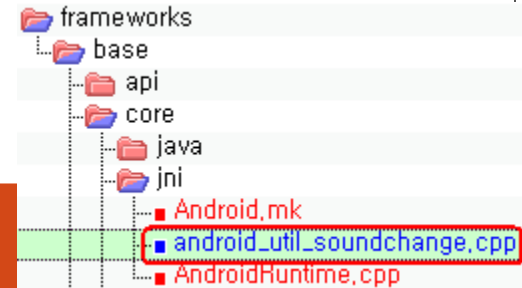
JNI

- `public native static void SoundDriverChange(int mode);`
- `./frameworks/base/soundchange/java/android/util/soundchange/SoundChangeNative.java`

JNI

- `{ "SoundDriverChange", "(I)V", (void*)android_util_soundchange_SoundChangeManager },`
- `./frameworks/base/core/jni/android_util_soundchange.cpp`

Android Framework(JNI)




```
{  
    return AndroidRuntime::registerNativeMethods(env,  
        "android/util/soundchange/SoundChangeNative", g_methods, NELEM(g_methods));  
}
```

```
public class SoundChangeNative {  
    public native static void SoundDriverChange(int mode);  
} ./frameworks/base/soundchange/java/android/util/soundchange/SoundChangeNative.java
```


```
static JNINativeMethod g_methods[] = {  
    { "SoundDriverChange", "(I)V", (void*)android_util_soundchange_SoundChangeManage  
        r },  
};
```

Android Framework(JNI~HAL)

```
android_util_soundchange_SoundChangeManager()  
{::sound_change_service(mode);}  
./frameworks/base/core/jni/android_util_soundchange.cpp
```



```
int sound_change_service(int mode)  
hardware/libhardware_legacy/soundchange/soundchange.c
```



```
file_copy("/system/etc/asound_wm8960.conf", "/system/etc/asound.conf");
```

Android Framework(LIB~HAL)

libandroid_runtime.so

./frameworks/base/core/jni/android_util_soundchange.cpp

```
::sound_change  
_service(mode)  
;
```

./hardware/libhardware_legacy/soundchange/soundchange.c

```
int sound_change_service(int  
mode)
```

Libruntime.so 등록하기 (core Libraries)

- `extern int register_android_util_soundchange_SoundChangeManager(JNIEnv* env);`
- `REG_JNI(register_android_util_soundchange_SoundChangeManager),`
- `frameworks\base\core\jni\AndroidRuntime.cpp` 에 추가



Android 메뉴 생성 (aidl등록)

```
LOCAL_SRC_FILES += \  
soundchange/java/android/util/soundchange/ISoundChangeManager.aidl \  
\  
.
```

Frameworks/base/Android.mk 파일에 추가

```
LOCAL_SRC_FILES := $(call find-other-java-files,$(FRAMEWORKS_BASE_SUBDIRS))
```



```
define find-other-java-files  
    $(call find-subdir-files,$(1) -name "*.java" -and -not -name ".*")  
Endef  
./build/core/definitions.mk
```

Android (current.xml)

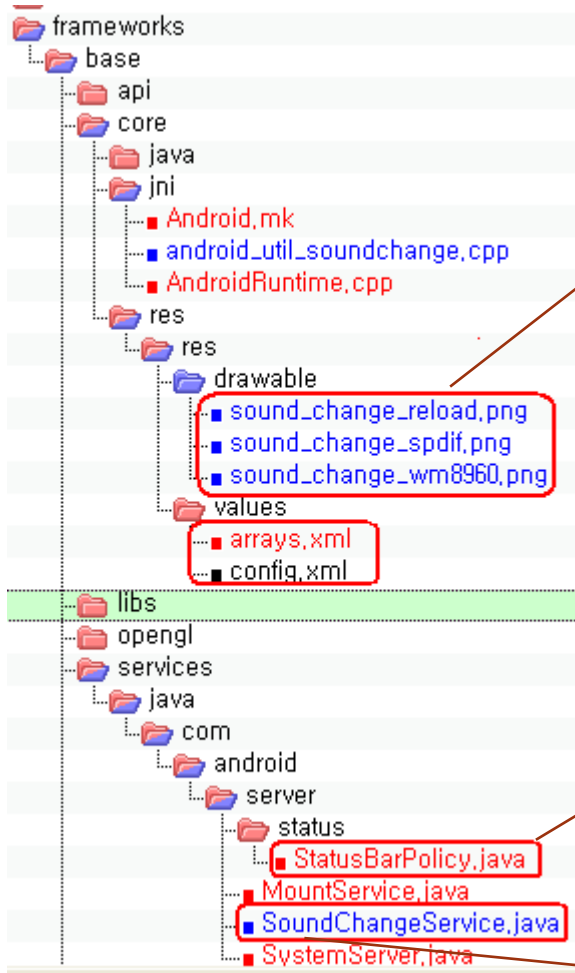
frameworks\base\api\current.xml

```
<field name="SOUNDCHANGE_SERVICE"  
type="java.lang.String"  
transient="false"  
volatile="false"  
value="&quot;soundchange&quot;"  
static="true"  
final="true"  
deprecated="not deprecated"  
visibility="public"  
>  
</field>
```

```
<field name="SOUND_CHANGE"  
type="java.lang.String"  
transient="false"  
volatile="false"  
value="&quot;sound_change&quot;"  
static="true"  
final="true"  
deprecated="not deprecated"  
visibility="public"  
>  
</field>
```

make update-api : 새로 추가된 java class 를 android platform 에 등록.
#make update-api를 실행하면 current.xml이 자동으로 등록 됨
#make
update-api: \$(INTERNAL_PLATFORM_API_FILE) | \$(ACP)
./build/core/tasks/apicheck.mk

Android 아이콘 생성



아이콘이미지

```
else if  
(action.equals(SoundChangeManager.SOUND_CHANGE_STATE_CHANGED_ACTION)){  
    updateSoundChange(intent);  
}
```

Control

```
(SoundChangeManager.SOUND_CHANGE_STATE_CHANGED_ACTION)
```

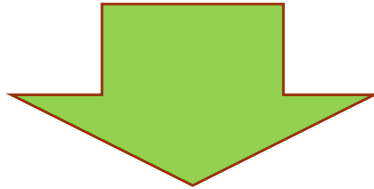
Android 아이콘 생성

```
final Intent intent = new Intent(SoundChangeManager.SOUND_CHANGE_STATE_CHANGED_ACTION);  
mContext.sendStickyBroadcast(intent);  
./frameworks/base/services/java/com/android/server/SoundChangeService.java
```



```
filter.addAction(SoundChangeManager.SOUND_CHANGE_STATE_CHANGED_ACTION);
```

```
private BroadcastReceiver mIntentReceiver = new BroadcastReceiver() {  
else if (action.equals(SoundChangeManager.SOUND_CHANGE_STATE_CHANGED_ACTION)){  
    updateSoundChange(intent);  
}  
./frameworks/base/services/java/com/android/server/status/StatusBarPolicy.java
```



```
updateSoundChange(intent);
```

Android 아이콘 생성



```
private final void updateSoundChange(Intent intent) {
    final int event = intent.getIntExtra(SoundChangeManager.EXTRA_SOUND_CHANGE_STATE, SoundChangeManager.SOUND_CHANGE_STATE_UNKNOWN);
    int iconId;
    switch (event) {
        case SoundChangeManager.SOUND_CHANGE_STATE_WM8960:
        default:
            iconId = sSoundChangeImages[0];
            break;
        case SoundChangeManager.SOUND_CHANGE_STATE_SPDIF:
            iconId = sSoundChangeImages[1];
            break;
        case SoundChangeManager.SOUND_CHANGE_STATE_RELOAD:
            iconId = sSoundChangeImages[2];
            break;
    }
    mSoundChangeData.iconId = iconId;
    mService.updateIcon(mSoundChangeIcon, mSoundChangeData, null);
}
```

```
intent.putExtra(SoundChangeManager.EXTRA_SOUND_CHANGE_STATE, mode);
```

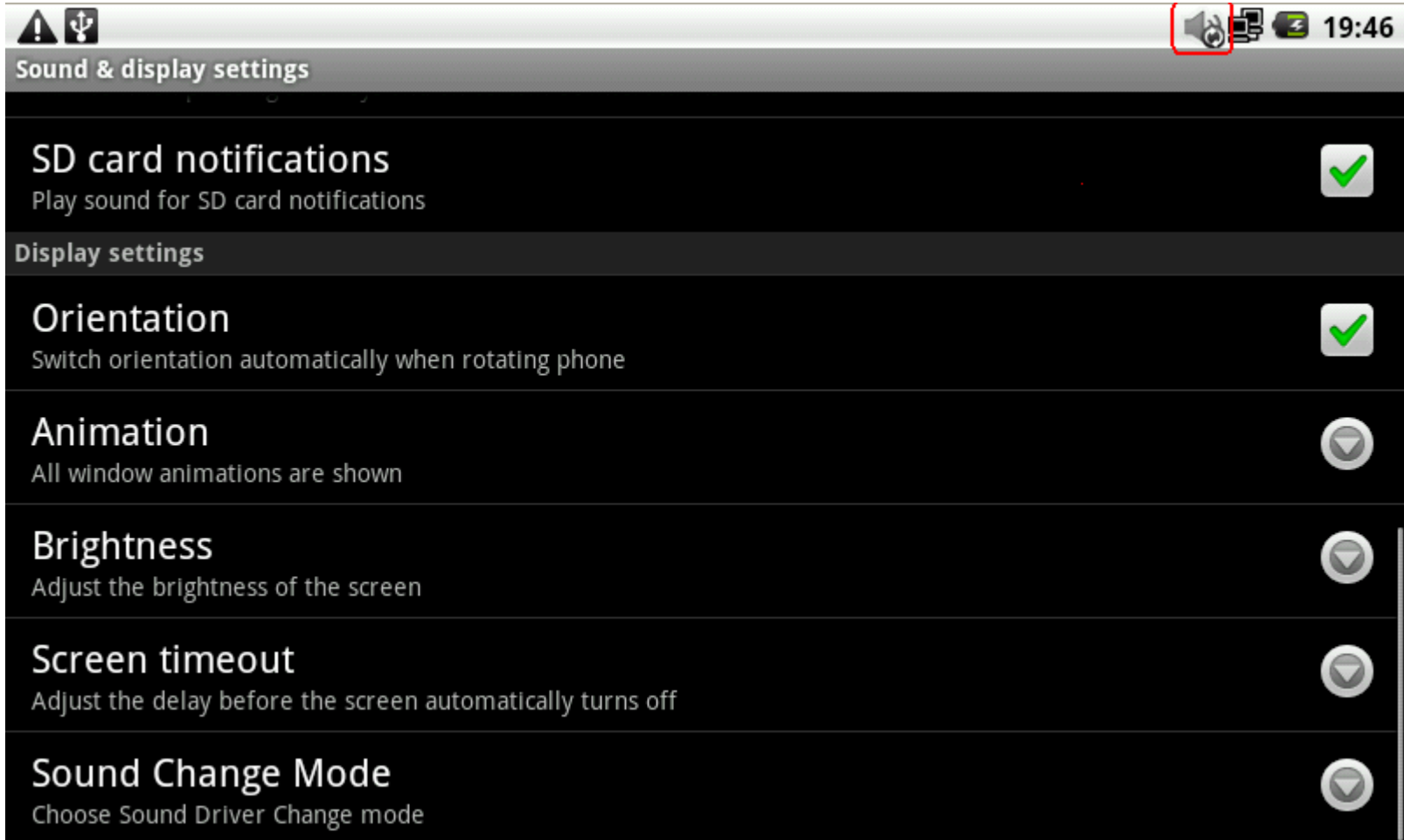
```
private static final int[] sSoundChangeImages = new int[] {
    com.android.internal.R.drawable.sound_change_wm8960,
    com.android.internal.R.drawable.sound_change_spdif,
    com.android.internal.R.drawable.sound_change_reload
};
private IBinder mSoundChangeIcon;
private IconData mSoundChangeData;
```

```
public void updateIcon(IBinder key,
    String slot, String iconPackage, int iconId, int iconLevel) {
    enforceStatusBar();
    updateIcon(key, IconData.makeIcon(slot, iconPackage, iconId, iconLevel, 0), null);
}
```

./frameworks/base/services/java/com/android/server/status/StatusBarService.java

<http://cafe.naver.com/embeddedcrazyboys>

안드로이드 아이콘 생성 결과

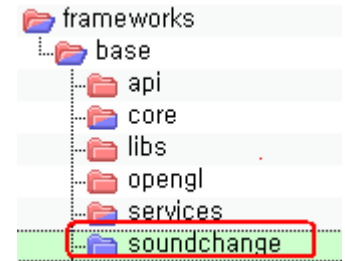


안드로이드 메뉴 생성(build)

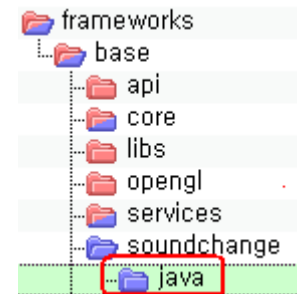
build\core\pathmap.mk 파일 수정

```
FRAMEWORKS_BASE_SUBDIRS := \  
    $(addsuffix /java, \  
        core \  
        graphics \  
        location \  
        media \  
        opengl \  
        sax \  
        telephony \  
        ethernet \  
        wifi \  
        vpn \  
        keystore \  
        soundchange \  
    )
```

```
FRAMEWORKS_BASE_JAVA_SRC_DIRS := \  
    $(addprefix frameworks/base/, $(FRAMEWORKS_BASE_SUBDIRS))
```



addsuffix /java ↓



안드로이드 메뉴 생성(aidl)

frameworks\base\Android.mk 파일 수정

```
# FRAMEWORKS_BASE_SUBDIRS comes from build/core/pathmap.mk
LOCAL_SRC_FILES := $(call find-other-java-files,$(FRAMEWORKS_BASE_SUBDIRS))

LOCAL_SRC_FILES := $(filter-out \
                    org/mobilecontrol/% \
                    ,$(LOCAL_SRC_FILES))
ifeq ($(TARGET_BUILD_TYPE),debug)
    LOCAL_SRC_FILES += $(call find-other-java-files,core/config/debug)
else
    LOCAL_SRC_FILES += $(call find-other-java-files,core/config/ndebug)
endif

LOCAL_SRC_FILES += \

    ethernet/java/android/net/ethernet/IEthernetManager.aidl \
    soundchange/java/android/util/soundchange/ISoundChangeManager.aidl \
```

<http://cafe.naver.com/embeddedcrazyboys>

AIDL은 Android Interface description language의 약자

Android build (find-other-java-files)

```
LOCAL_SRC_FILES := $(call find-other-java-files,$(FRAMEWORKS_BASE_SUBDIRS))
```



```
define find-other-java-files
    $(call find-subdir-files,$(1) -name "*.java" -and -not -name ".*")
Endef
./build/core/definitions.mk
```

안드로이드 메뉴 생성

frameworks\base\core\java\android\app\ApplicationContext.java에 추가

```
import android.util.soundchange.ISoundChangeManager;  
import android.util.soundchange.SoundChangeManager;
```

frameworks\base\soundchange\java\android\util\soundchange\SoundChangeManager.java 를 포함

```
package android.util.soundchange;
```

```
import java.util.List;
```

```
import android.annotation.SdkConstant;
```

frameworks\base\soundchange\java\android\util\soundchange\ISoundChangeManager.aidl

```
package android.util.soundchange;
```

```
interface ISoundChangeManager
```

```
{
```

```
    int getSoundChangeStatus();
```

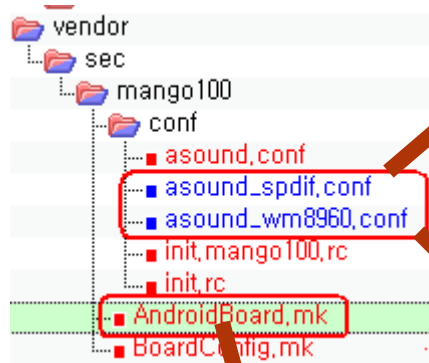
```
    int getPreSoundChangeStatus();
```

```
    void setSoundChangeMode(int mode);
```

```
}
```

<http://cafe.naver.com/embeddedcrazyboys>

Build 환경 구성



```
ctl.AndroidPlayback {
    type hw
    card 1
}
```

```
# cat info
card: 1
device: 0
subdevice: 0
stream: PLAYBACK
id: HDMI-SPDIF Playback HDMI-SPDIF Codec-0
name:
subname: subdevice #0
class: 0
subclass: 0
subdevices_count: 1
subdevices_avail: 1
# pwd
/proc/asound/card1/pcm0p
```

```
ctl.AndroidPlayback {
    type hw
    card 0
}
```

```
# pwd
/proc/asound/card0/pcm0c
# cat info
card: 0
device: 0
subdevice: 0
stream: CAPTURE
id: Tx/Rx WM8960-0
name:
subname: subdevice #0
class: 0
subclass: 0
subdevices_count: 1
subdevices_avail: 1
```

```
PRODUCT_COPY_FILES += \
```

```
vendor/sec/mango100/conf/asound.conf:system/etc/asound.conf \
vendor/sec/mango100/conf/asound_wm8960.conf:system/etc/asound_wm8960.conf \
vendor/sec/mango100/conf/asound_spdif.conf:system/etc/asound_spdif.conf \
```

안드로이드 아이콘 생성

```
<item><xliff:g id="id">soundchange</xliff:g></item>
```

\frameworks\base\core\res\res\values\arrays.xml 파일에 추가

```
<!-- Declared at res/drawable/sound_change_reload.png:0 -->  
<public type="drawable" name="sound_change_reload" id="0x01080271" />  
./out/target/common/obj/APPS/framework-res_intermediates/public_resources.xml
```

SD Card 인식 (Froyo)

- 부팅 후 아래와 같이 수정
- `dev_mount sdcard /mnt/sdcard auto /devices/platform/s3c-sdhci.0/mmc_host/mmc0`
- 또는
- `vendor/sec/mango100/conf/vold.fstab` 파일을 아래와 같이 수정
- `dev_mount sdcard /mnt/sdcard auto /devices/platform/s3c-sdhci.0/mmc_host/mmc0`

ALSA Sound 소스 받기(froyo)

#vi .repo/manifest.xml 을 수정

```
<project path="external/alsa-lib" name="platform/external/alsa-lib" />
```

```
<project path="external/alsa-utils" name="platform/external/alsa-utils" />
```

```
<project path="hardware/alsa_sound" name="platform/hardware/alsa_sound" />
```

#repo sync

#ls external

```
[icanjji@localhost froyo]$ ls external/  
alsa-lib      dhcpd        fdlibm       iptables  
alsa-utils   dnsmasq     freetype     jdiff  
apache-http  dropbear    fsck_msdos   jhead  
astl         2f494949    2f494949    2f494949
```

#ls hardware

```
[icanjji@localhost froyo]$ ls hardware/  
alsa_sound  broadcom  libhardware  libhardware_legacy
```

ALSA Sound 설정(froyo)

build/target/board/generic/BoardConfig.mk

을 추가

```
BOARD_USES_GENERIC_AUDIO := false
```

```
BUILD_WITH_ALSA_UTILS := true
```

```
BOARD_USES_ALSA_AUDIO := true
```

-asound.conf 파일을 작성 파일 시스템에 복사

```
# cp asound.conf system/usr/share/alsa_sound/
```

또는

build/target/board/generic/AndroidBoard.mk 파일에 아래와 같이 수정

```
PRODUCT_COPY_FILES += \
```

```
    build/target/board/generic/conf/asound.conf:system/etc/asound.conf
```

Patch 만들기 , 사용법

- `#diff -urN SRC DES소스 > xxx.diff`
- `#cat xxx.diff | patch -p1`
- <http://crztech.ipstime.org:8080/Release/mango100/eclair/android-menu-patch.tar> 다운받기
- `# cd mango100_eclair_2010_06_30`
- `# cat ../../mango100-eclair-menu.diff | patch -p1`
- `# cp sound_change_reload.png
frameworks/base/core/res/res/drawable/`
- `#cp sound_change_spdif.png
frameworks/base/core/res/res/drawable/`
- `# cp sound_change_wm8960.png
frameworks/base/core/res/res/drawable/`