

# Linux/Android/WinCE 対応マルチ・メディア ARM11 ボード Idea6410+LCD4. 3 の Android 2.1 マニュアル

# 株式会社日昇テクノロジー

http://www.csun.co.jp

info@csun.co.jp

2010/11/20



copyright@2011



# 修正履歴

NO	バージョン	修正内容	修正日
1	1.0	新規作成	2010/7/9
2	1. 1	USB ADB ドライバのインストールと使用を追加	2010/9/6
3	1.2	ユーザーボタンの説明を修正	2010/9/28
4	1.3	無線 LAN 設定手順を追加	2010/11/20

- ※ 使用されたソースコードは <a href="http://www.csun.co.jp/">http://www.csun.co.jp/</a>からダウンロードできます。
- ※ この文書の情報は、事前の通知なく変更されることがあります。
- ※ (株)日昇テクノロジーの書面による許可のない複製は、いかなる形態に おいても厳重に禁じられています。

ホームページ: <a href="http://www.csun.co.jp">http://www.csun.co.jp</a> メール: <a href="mailto:info@csun.co.jp">info@csun.co.jp</a> 2





第一章	Android2.1の紹介	5
1.1 /	バージョン特性	5
1.2 A	ndroid 2.1 の新規機能	5
1.3	コンパイル環境について	5
第二章	開発環境の設置	6
2.1 Ora	cle VirtualBox VMでUbuntu10.04インストール	7
2. 1. 1	Oracle VirtualBox VM インストール	7
2. 1. 2	VMマシンに Ubuntu10.04 インストール	14
2. 1. 3	Guest Additionsインストール	25
2. 2. 4	共有フォルダアクセス	27
2.2 Vmw	rare7.0上のUbuntuインストール	29
2. 2. 1	Vmware7.0のインストール	29
2.2.2	Vmware7.0上ubuntu10.04のインストール	33
2. 2. 3	Vmware toolsのインストール	53
2. 2. 4	共有フォルダの設定	57
2. 2. 5	JDK1.5のインストール	61
2. 2. 6	その他のツールのインストール	63
第三章	Cross-compile のインストール	64
第四章	ソースのコンパイル	66
4.1 u-b	oot のコンパイル	66
4.2 カ	ーネルのコンパイル	66
4.3 an	droid2.1のコンパイル	67
第五章	イメージファイルの書き込む	69
5.1 SD	起動用カードの作成	69
5. 1. 1	注意事項	69
5. 1. 2	SD 起動用カードの作成手順	69
5.2 ub	oot の書き込み	70
5. 2. 1	SD 起動モードに設定	70
5. 2. 2	uboot. bin の書込み	70
5. 2. 2	zImage ファイルの書込み	73
第六章	Windows XP上のUSB ADB インストール	74
	B ADB に関わるカーネル設定の修正	
5. 1. 1	. USB ADB ドライバソース修正	74
5. 1. 2	. USB ADBに関するカーネル設定の修正	74
5.2 US	B ADB ドライバーインストール	77







第七章	USB ADB 使用について	83
7.1 A	ADB コマンド	83
	ファイルをアップロード/ダウンロード	
第八章	Android 開発環境構築(Windows 編)	89
8.1	Android SDK のインストール	89
8.2	Eclipse のインストール	94
8.3	ADT プラグインのインストール	96
8.4	Eclipse 上サンプルの実行	100
8.5	Android エミュレータでのデバッグ	113
8.6	ARM11 ボードの Android 実機にデバッグ	127
第九章	ARM11 ボードに Android アプリを実行	128
9.1	ARM11 ボードにアプリをインストール	128
9.2	ARM11 ボードにアプリを動かす	129
付録・	ネットワーク設定	135

メール:<u>info@csun.co.jp</u>



# 第一章 Android2.1 の紹介

- 1.1 バージョン特性
- 1、Bootloader バージョン: u-boot-1.1.6
- 2、 カーネルバージョン: 1inux2.6.29
- 3、 Android バージョン:Android 2.1
- 4、 JDK バージョン: JDK1.5 (sun-java5-jdk)
- 1.2 Android 2.1 の新規機能
- 1、 速度の改善
- 2、 Personalize 設定項目の追加
- 3、無線部分 VPN 設定の追加
- 4、 MobileNetworkSharing機能の追加
- 5、 Location と Privacy オプションの追加
- 6、 ConnecttoPC 設定の追加
- 7、 音声変換機能の強化
- 8、 ダイヤル画面の改善
- 9、 Widget の追加
- 10、 検索機能の改善

#### 1.3 コンパイル環境について

1、Linux サーバ、ubuntu10.04 をインストールしてホストとする。他は XP 環境で SSH/Telnet を通じてコンパイルする。

ホームページ: http://www.csun.co.jp

メール: info@csun.co.jp



# 第二章 開発環境の設置

開発環境のホスト環境「ubuntulo.04」は三つ方法で用意できます。

- 1. 実機で ubuntu10.04 をインストール
- 2. Vmware で仮想マシンを作成し ubuntu10.04 をインストール
- 3. Oracle VM VirtualBox で仮想マシンを作成し ubuntu10.04 をインストールリソースダウンロード:
- ◆Ubuntu10.04 ダウンロード URL: (無料) http://www.ubuntu.com/desktop/get-ubuntu/download
- ◆Oracle VM VirtualBox ダウンロードURL: (無料) ホームページ: http://www.virtualbox.org/wiki/Downloads

インストーラ:

http://download.virtualbox.org/virtualbox/3.2.6/VirtualBox-3.2.6-63112-Win.exe

◆Vmware ダウンロード URL: (ラインセンス料金必要)

メーカサイトから評価版ダウンロード:(最新版ダウンロードできるが、ユーザー登録必要)

https://www.vmware.com/jp/tryvmware/?p=vmware-workstation&lp=default

弊社サイトから評価版ダウンロード: (VMware7.0のみ、ユーザー登録不要)

http://www.dragonwake.com/download/idea6410/android2.1/VMware+7.0.zip

ホームページ: http://www.csun.co.jp メール: info@csun.co.jp 6



- 2.1 Oracle VirtualBox VM で Ubuntu10.04 インストール
- 2.1.1 Oracle VirtualBox VM インストール

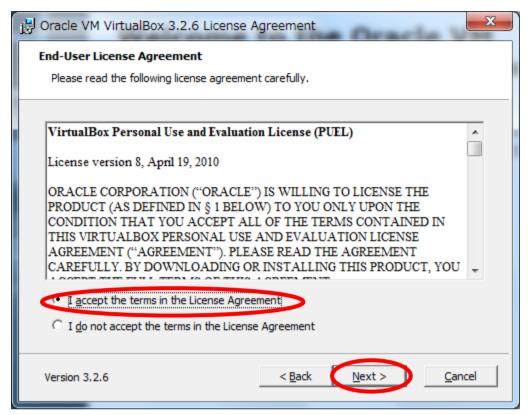
http://www.virtualbox.org/wiki/Downloads から VirtualBox をダウンロードする。 ダウンロードしたインストールファイルをクリックすると、下記画面に従って進めてください。

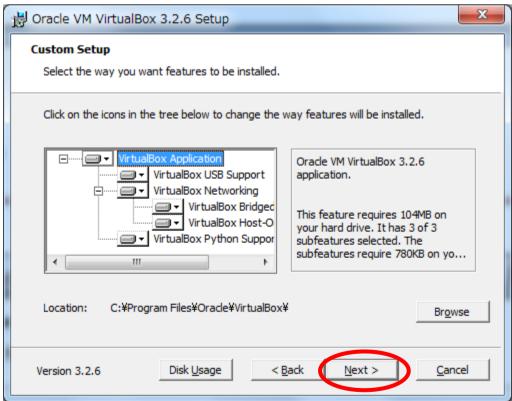


ホームページ: http://www.csun.co.jp

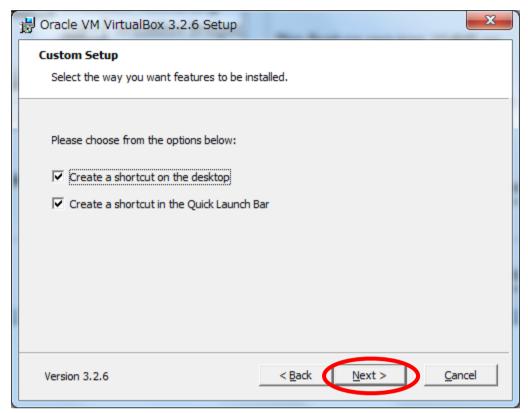
メール: info@csun.co.jp

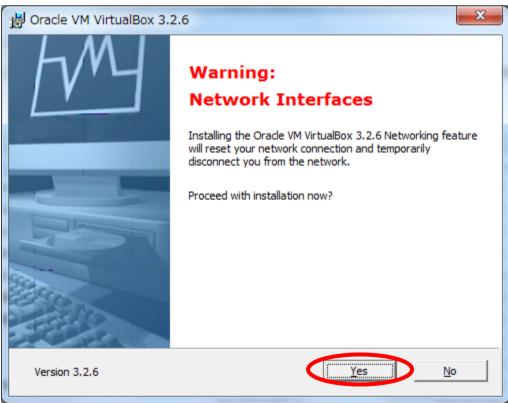




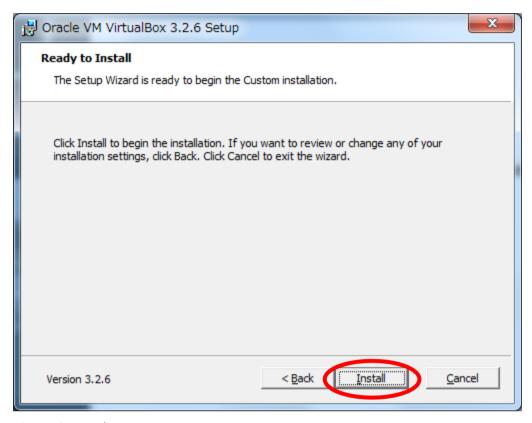




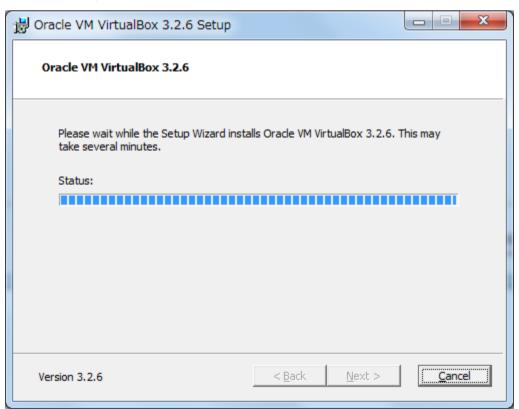








## インストール中

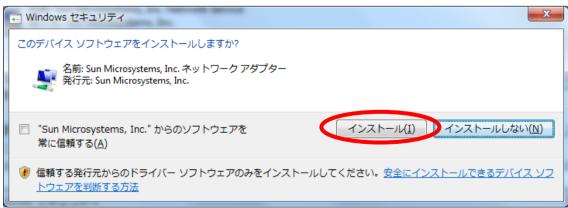




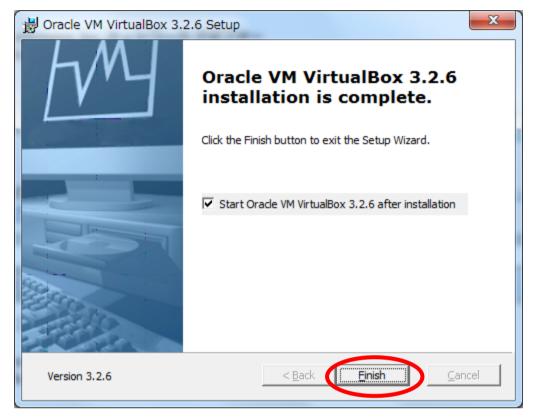




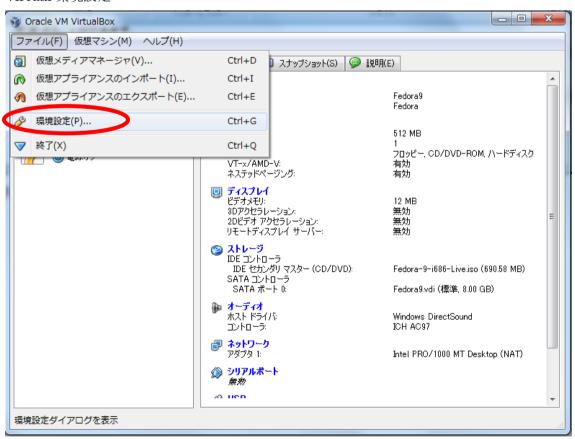




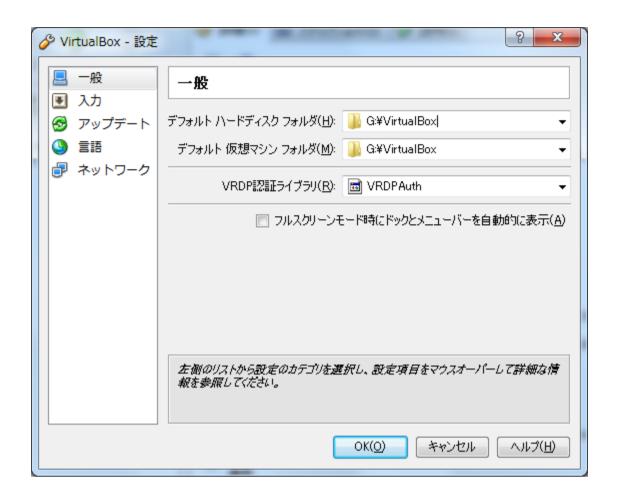




#### Virtual 環境設定



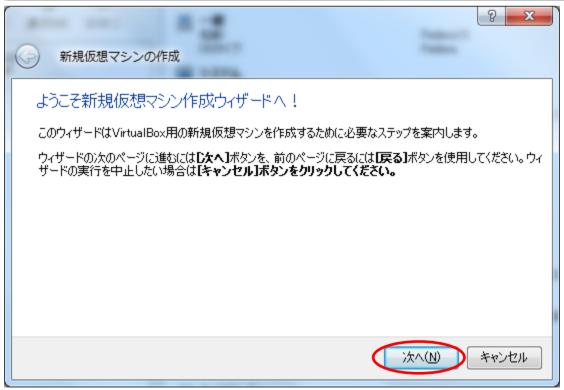




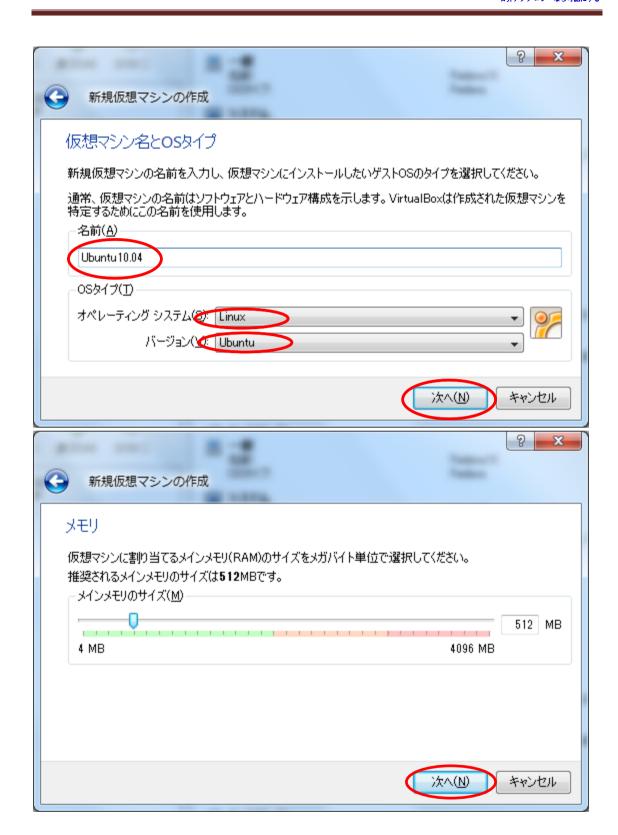


#### 2.1.2 VM マシンに Ubuntu10.04 インストール

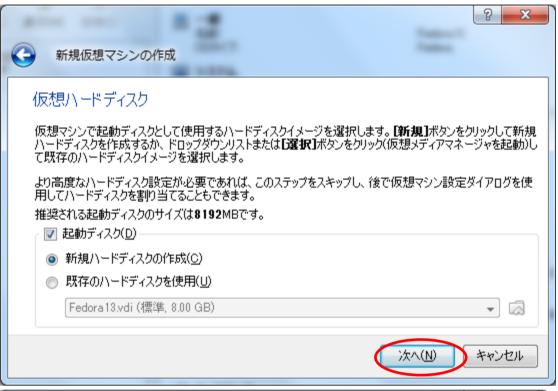


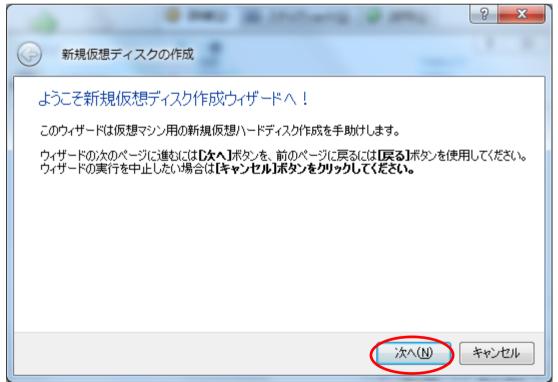




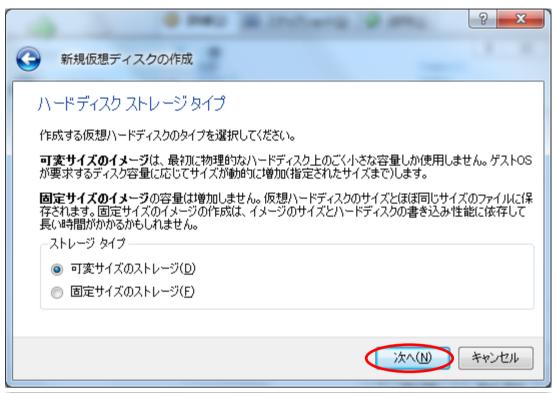


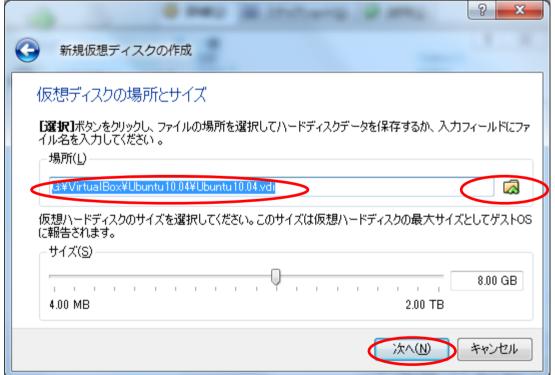




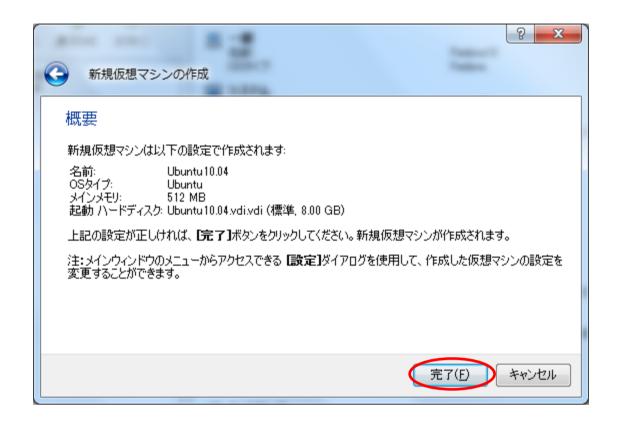








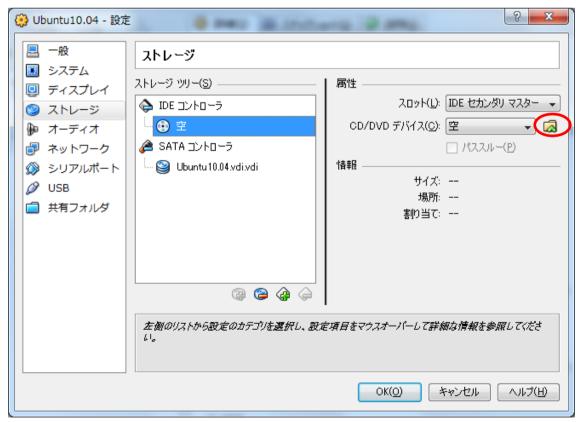




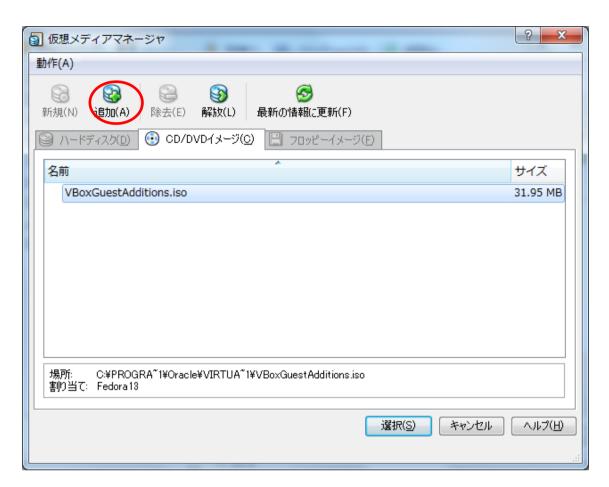


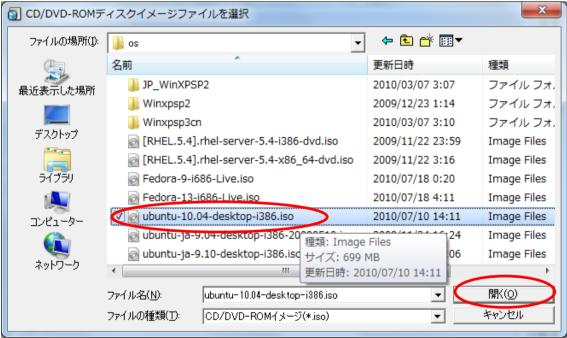
仮想マシン一覧から Ubuntu10.04 を選択して右クリックし「設定」メニューを押下



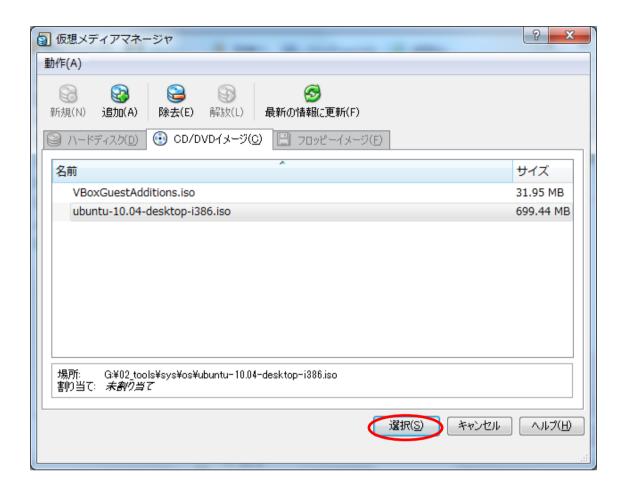




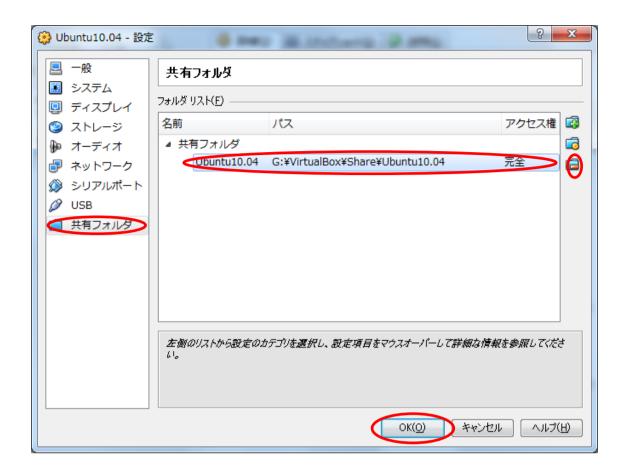






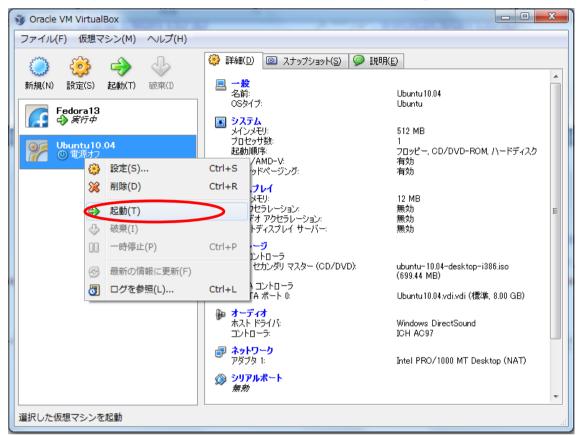




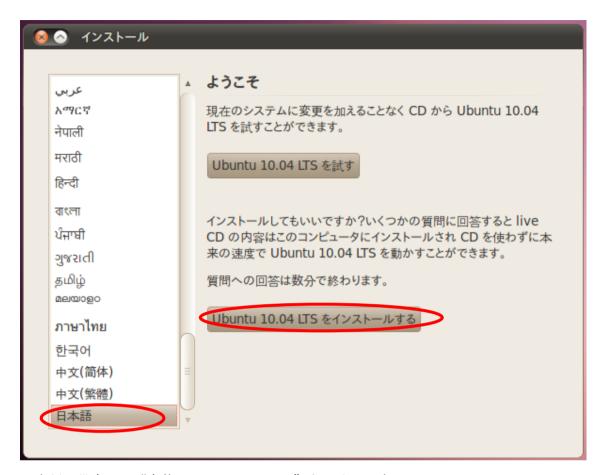




仮想マシン一覧から Ubuntu10.04 を選択して右クリックし「起動」メニューを押下







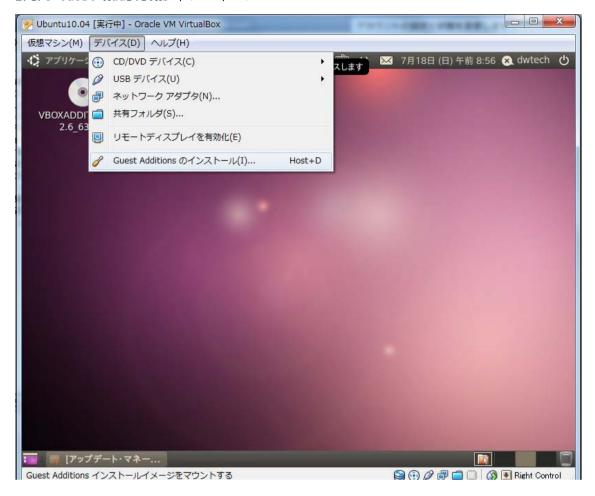
日本語に設定して"安装 Ubuntu 10.04 LTS"をクリックする。

次は場所、キーボード、ハードディスク、ユーザーネームとパスワードを設定して、インストールする。

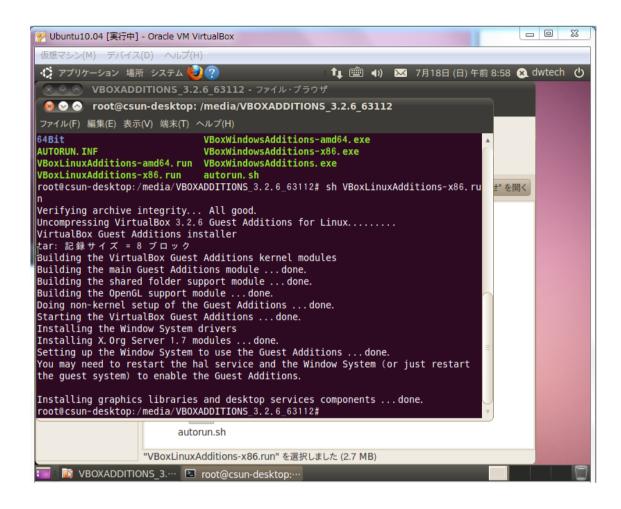
インストール完了後、ubuntu10.04 を再起動する。



## 2.1.3 Guest Additions インストール





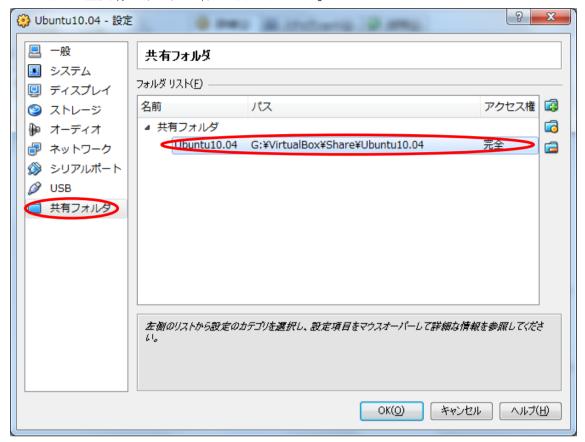




## 2.2.4 共有フォルダアクセス

前の節に VirtualBox で下記のように共有フォルダーを設定された。

VirtulaBox 上共有フォルダー名:「Ubuntu10.04」

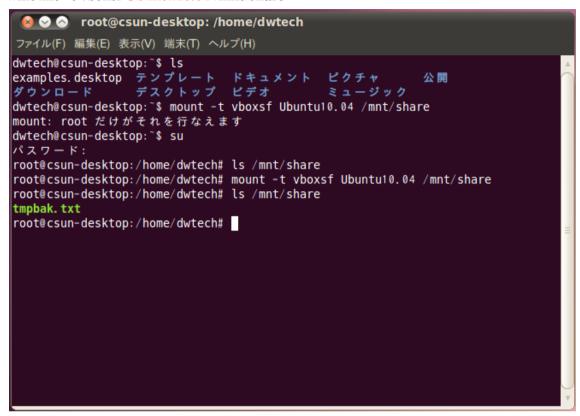


日昇テクノロジーなら可能にする

Ubuntu10.04 に共有フォルダーをアクセルするため、下記マウント必要。

#mkdir/mnt/share

#mount -t vboxsf Ubuntu10.04 /mnt/share



Ubuntu 上コンパイルツール等のインストールは本マニュアル後ろの VMWare 上 VM マシン「Ubuntu10.04」インストール手順と同じですので、そちらに下記 2 節に参照

2.2.5 JDK1.5のインストール

2.2.6 その他のツールのインストール



## 2.2 Vmware7.0 上の Ubuntu インストール

#### 2.2.1 Vmware7.0 のインストール

¥VMware7.0\_ubuntu10.04¥VMware 7.0¥フォルダにある下記ファイルをダブルクリックして 実行する。(弊社サイトから VMware をダウンロードした場合)

VMware-workstation-full-7.0.0-203739.exe

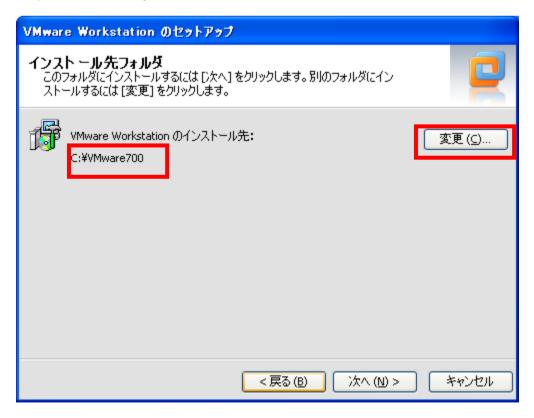


"次へ"をクリックする。



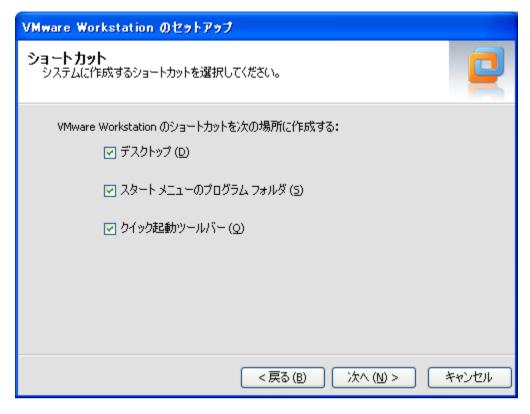


"標準"を選択する。



フォルダを選択して"次へ"をクリックする。



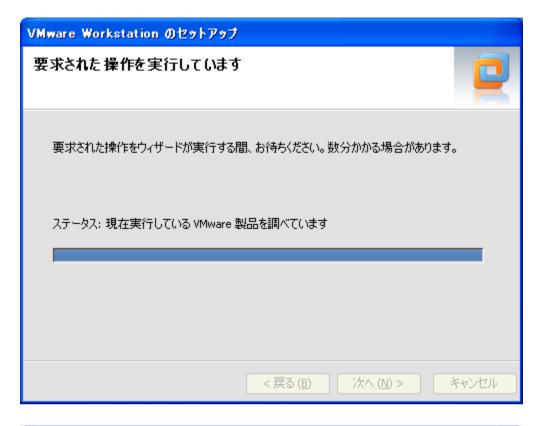


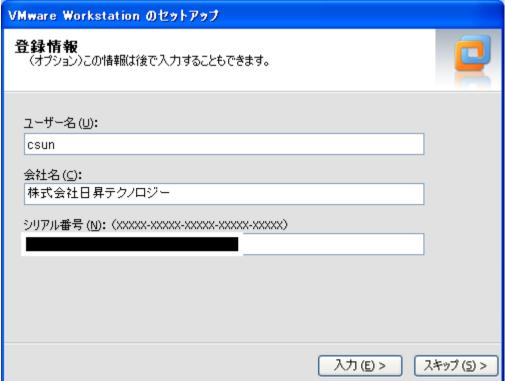
"次へ"をクリックする。



"続行" をクリックする。







登録情報とシリアルナンバーを入力して"入力(E)"をクリックする。





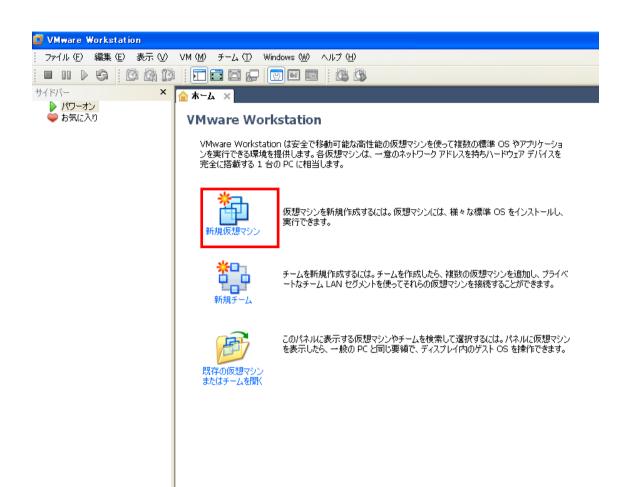
"今すぐ再起動(N)"をクリックして、PCを再起動する。

- 2.2.2 Vmware7.0上 ubuntu10.04 のインストール
- ① 新しい VM を作成する

Vmware Workstation を開く。



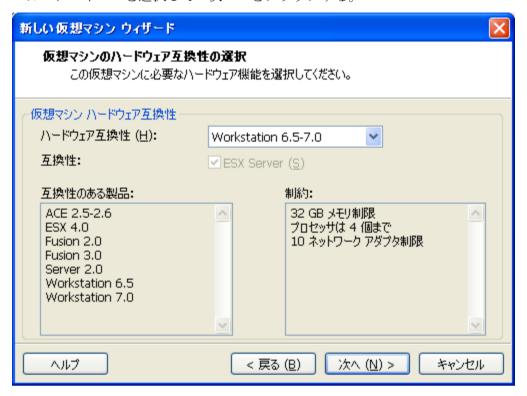
日昇テクノロジーなら可能にする



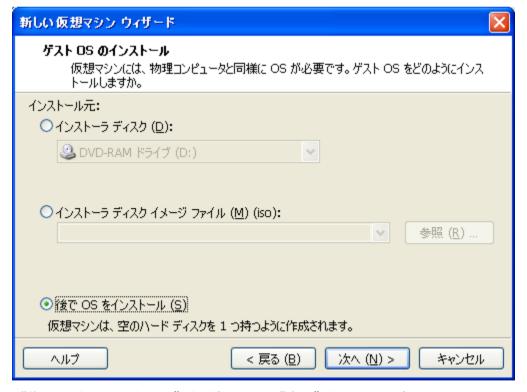




"カスタマイズ"を選択して"次へ"をクリックする。

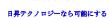


"次へ"をクリックする。



"後で OS をインストール"を選択して、"次へ"をクリックする。

※インストーラディスクイメージファイル(iso)を選択しない方が良い、起動時キーボー





ド使えない恐れがある為。

ホームページ : <u>http://www.csun.co.jp</u>

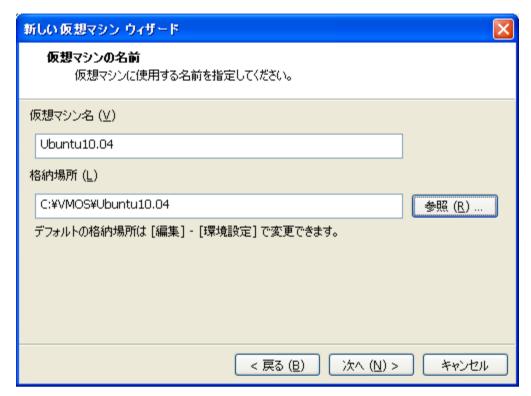
メール: <u>info@csun.co.jp</u>





上記画面の様に「ゲストOS」の所は"Linux"、Versionの所は"Ubuntu"を選択して、 "次へ"をクリックする。



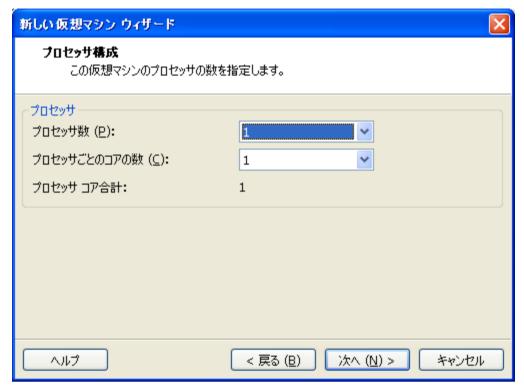


仮想マシン名: ubuntu10.04

格納場所: C:\\VMOS\\VDUntu10.04

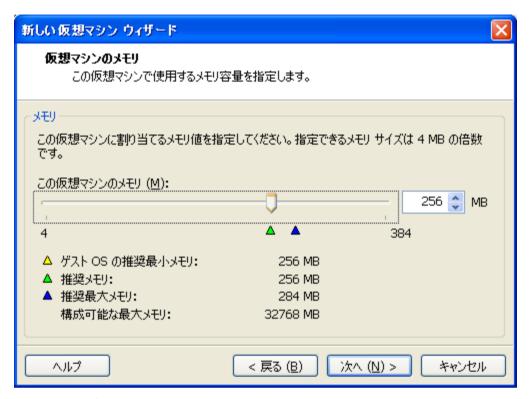
"次へ"をクリックする。





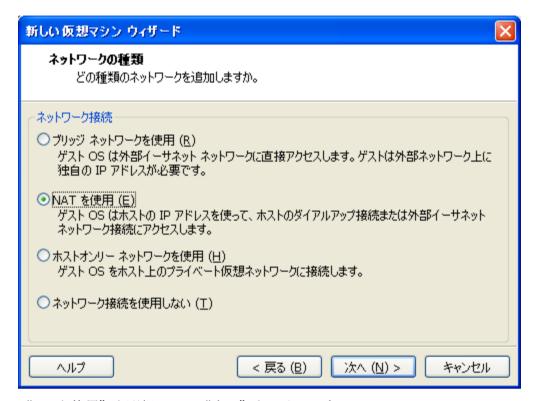
"次へ"をクリックする。





全て設定はデフォルトのまま、"次へ"をクリックする。





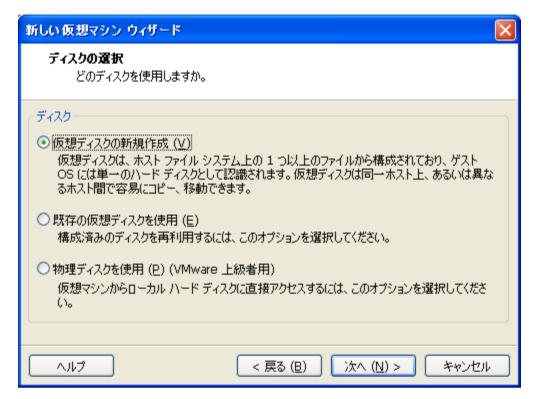
"NAT を使用"を選択して、"次へ"をクリックする。





"LSI Logic" を選択して、"次へ"をクリックする。





"仮想ディスクの新規作成" を選択して、"次へ"をクリックする。





"SCSI" を選択して、"次へ"をクリックする。

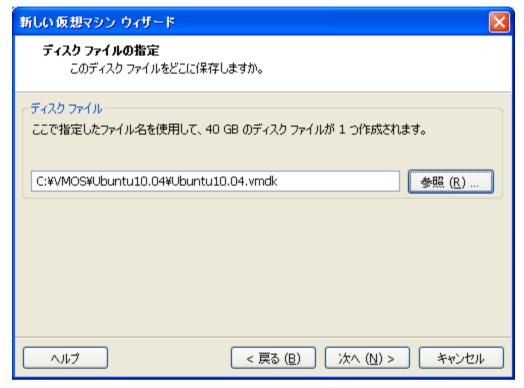




ディスク最大サイズ(S) (GB): 40GB

"仮想ディスクを単一ファイルとして格納" を選択して、"次へ"をクリックする。





VMDK フォルダを作成して、"参照"でこのフォルダを指定する。 "次へ"をクリックする。



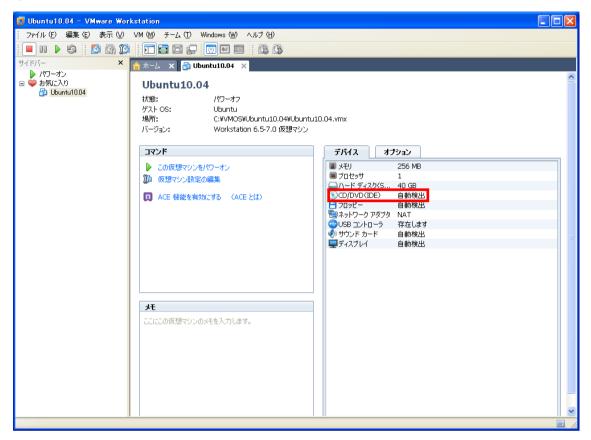


"完了"クリックして、新しい VM が作成完了。

次はこの VM で ubuntu10.04 をインストールする。

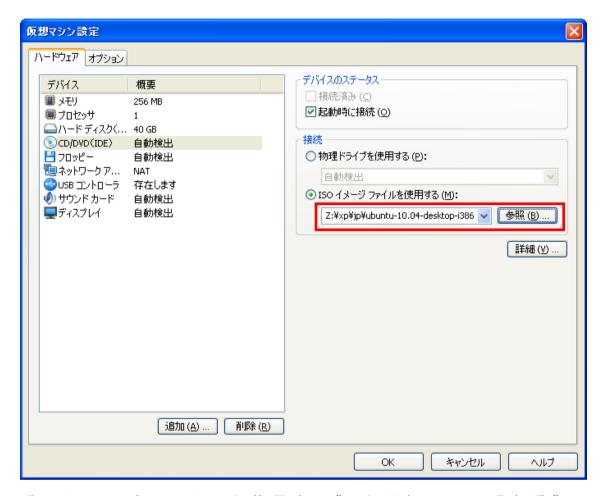


# ② ubuntu10.04 をインストールする



"CD/DVD (IDE)" をダブルクリックする。

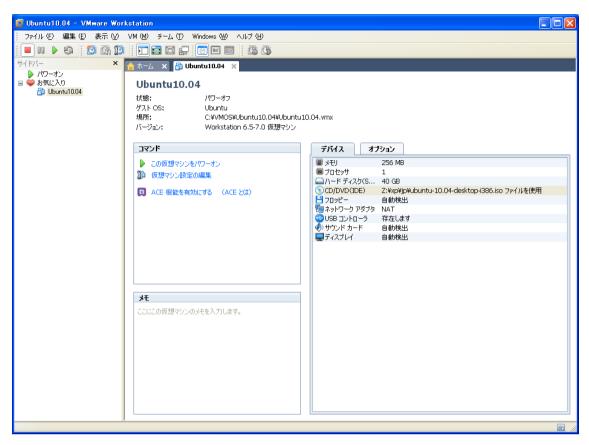




"ISO イメージファイルを使用する" を選択して、"参照"で Z:\xp\jp\ubuntu-10.04-desktop-i386.iso を選択する。

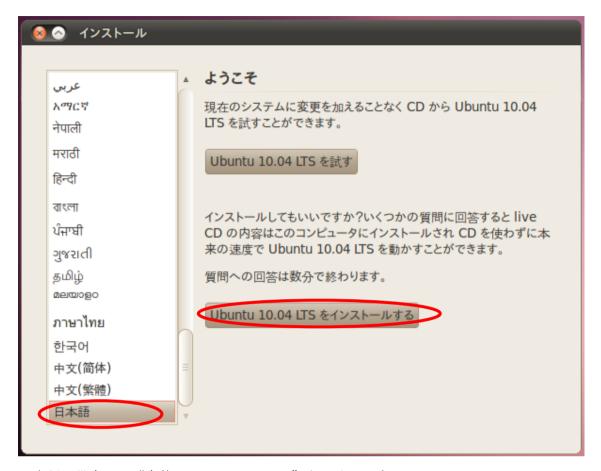
"OK" をクリックする。





"この仮想マシンをパワーオン" をクリックする。



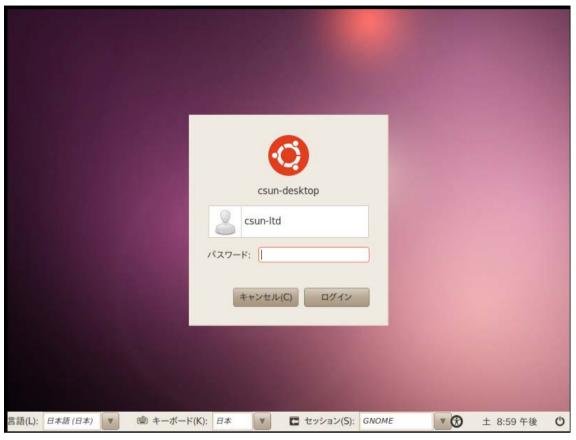


日本語に設定して"安装 Ubuntu 10.04 LTS"をクリックする。

次は場所、キーボード、ハードディスク、ユーザーネームとパスワードを設定して、インストールする。

インストール完了後、ubuntu10.04 を再起動する。





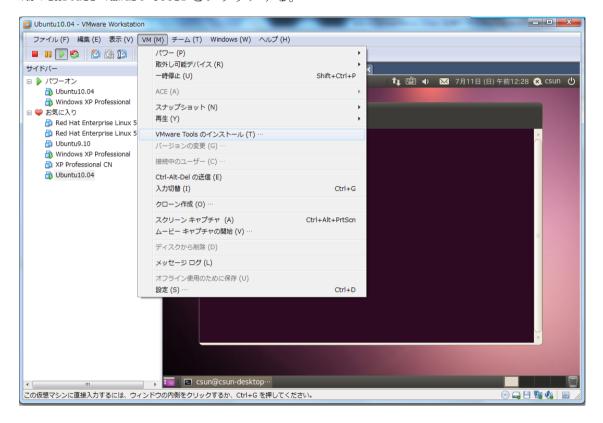
ユーザ名とパスワードを入力して起動する。



# 2.2.3 Vmware tools のインストール

Vmware Tools は VM と XP 間にファイルの転送、文字列のコピーなど Ctrl+Alt を操作しなく ても実現できる様にするツールである。

VM->Install Vmware Tools をクリックする。

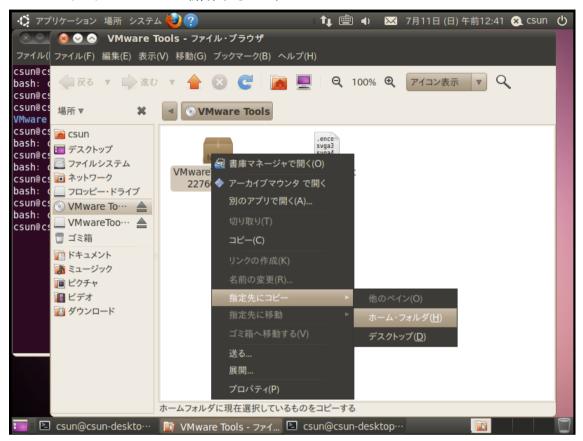








ホームフォルダーへコピー (解凍するため)



ハイパターミナルを起動する。

1、ユーザー「csun」のホームフォルダーに入る:

## #cd ~

```
csun@csun-desktop: ** cd **
csun@csun-desktop: ** ls
VMwareTools-8.1.4-227600.tar.gz テンプレート ピデオ 公開
examples.desktop デスクトップ ピクチャ
ダウンロード ドキュメント ミュージック
csun@csun-desktop: **
```

2、VmwareTools-8.1.4-227600.tar.gz を/tmp/に解凍する:

# tar xvf VMwareTools-8.1.4-227600.tar.gz -C /tmp/

```
csun@csun-desktop: $ ls

VMwareTools-8.1.4-227600.tar.gz テンプレート ビデオ 公開

examples.desktop デスクトップ ピクチャ
ダウンロード ドキュメント ミュージック

csun@csun-desktop: $ tar xvf VMwareTools-8.1.4-227600.tar.gz -C /tmp
```

3、/tmp/vmware-tools-distrib/に入る:

#cd /tmp/vmware-tools-distrib/

csun@csun-desktop:~\$ cd /tmp/vmware-tools-distrib/ csun@csun-desktop:/tmp/vmware-tools-distrib\$

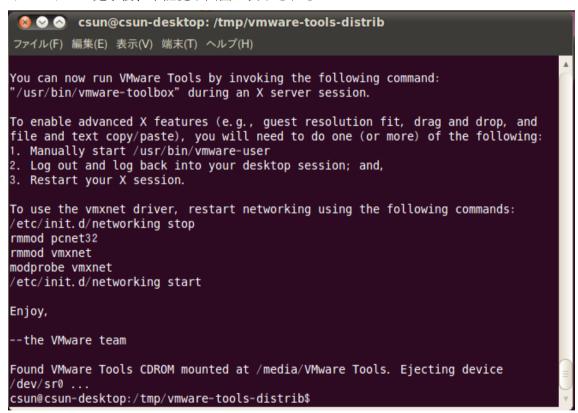
4、vmware-install.pl を実行してインストールする。

#sudo ./ vmware-install.pl

csun@csun-desktop:/tmp/vmware-tools-distrib\$ sudo ./vmware-install.pl

パスワードを入力して(画面上に表示しない)、[/usr/bin]、[/etc]などのフォルダがあったら、Enter を押して、[yes]の場合は y 押して Enter、[no]の場合は n 押して Enter を押す。

インストール完了後、下記提示画面が表示される:

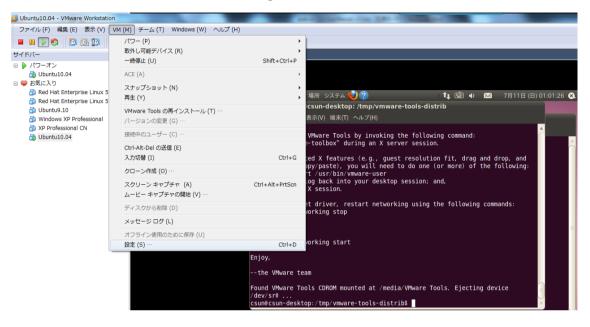




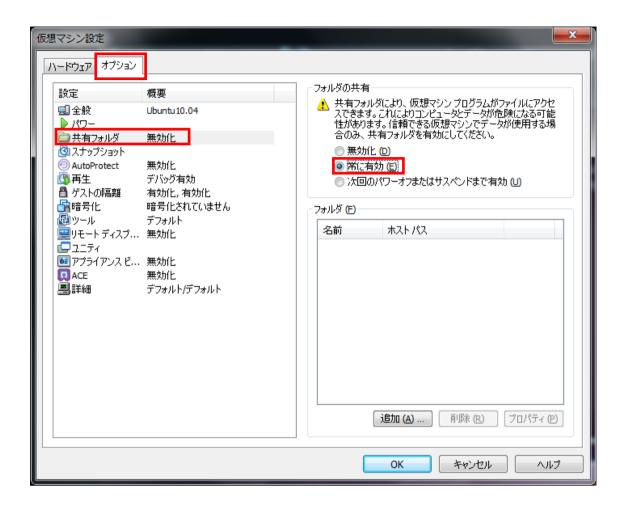
# 2.2.4 共有フォルダの設定

XP 環境でのあるフォルダを ubuntu の共有フォルダに設定できる。Ubuntu 環境でのパスは /mnt/hgfs/。

- 1、XP 環境で一つの共有フォルダを作る。例としては F ドライバで ubuntu share folder。
- 2、VMで→マウスの右クリック→setting



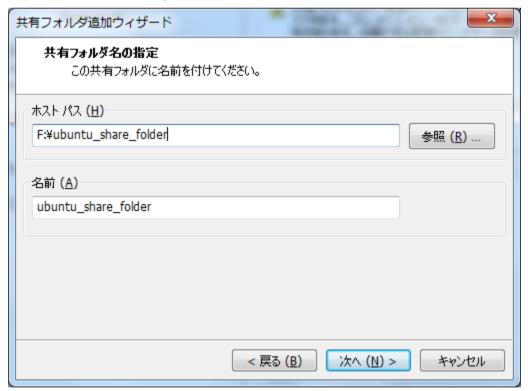








"次へ"をクリックする。



"次へ"をクリックする。





"完了"をクリックする。

設定完了後、ubuntu で共有フォルダが見える:

csun@csun-desktop:/tmp/vmware-tools-distrib\$ ls /mnt/hgfs ubuntu share folder csun@csun-desktop:/tmp/vmware-tools-distrib\$



## 2.2.5 JDK1.5のインストール

1、VMでイーサネット接続確保するうえ、先ずシステムをバージョンアップする。

# sudo apt-get update

```
🔕 📀 🔗 csun@csun-desktop: /tmp/vmware-tools-distrib
ファイル(F) 編集(E) 表示(V) 端末(T) ヘルプ(H)
ヒット http://jp.archive.ubuntu.com lucid/restricted Sources
ヒット http://jp.archive.ubuntu.com lucid/universe Packages
ヒット http://jp.archive.ubuntu.com lucid/universe Sources
ヒット http://jp.archive.ubuntu.com lucid/multiverse Packages
ヒット http://jp.archive.ubuntu.com lucid/multiverse Sources
ヒット http://jp.archive.ubuntu.com lucid-updates/main Packages
ヒット http://jp.archive.ubuntu.com lucid-updates/restricted Packages
ヒット http://jp.archive.ubuntu.com lucid-updates/main Sources
ヒット http://jp.archive.ubuntu.com lucid-updates/restricted Sources
ヒット http://jp.archive.ubuntu.com lucid-updates/universe Packages
ヒット http://jp.archive.ubuntu.com lucid-updates/universe_Sources
ヒット http://jp.archive.ubuntu.com lucid-updates/multiverse Packages
ヒット http://jp.archive.ubuntu.com lucid-updates/multiverse Sources
ヒット http://security.ubuntu.com lucid-security/main Packages
ヒット http://security.ubuntu.com lucid-security/restricted Packages
ヒット http://security.ubuntu.com lucid-security/main Sources
ヒット http://security.ubuntu.com lucid-security/restricted Sources
ヒット http://security.ubuntu.com lucid-security/universe Packages
ヒット http://security.ubuntu.com lucid-security/universe Sources
ヒット http://security.ubuntu.com lucid-security/multiverse Packages
ヒット http://security.ubuntu.com lucid-security/multiverse Sources
1,232kB を 2s で取得しました(491kB/s)
パッケージリストを読み込んでいます... 完了
csun@csun-desktop:/tmp/vmware-tools-distrib$
```

# 2、ソースを更新する

◆ /etc/apt/sources.list をsources.list.back にバックアップする。

# sudo cp /etc/apt/sources.list /etc/apt/sources.list.back

csun@csun-desktop:/tmp/vmware-tools-distrib\$ sudo cp /etc/apt/sources.list /etc/
apt/sources.list.back
csun@csun-desktop:/tmp/vmware-tools-distrib\$

◆vim で/etc/apt/sources.list を開く (まだ vim をインストールしてない場合は、sudo apt-get install vim を実行してインストールする)。

下記二行を source. list の最後にコピーする。

deb http://us.archive.ubuntu.com/ubuntu/ jaunty multiverse deb http://us.archive.ubuntu.com/ubuntu/ jaunty-updates multiverse 保存してクローズする。

# sudo vim /etc/apt/sources.list

```
deb http://us.archive.ubuntu.com/ubuntu/ jaunty multiverse
deb http://us.archive.ubuntu.com/ubuntu/ jaunty-updates multiverse
```



- ◆sudo apt-get updateを実行して、再度ソースを更新する。
- ◆sudo apt-get install sun-java5-jdk を実行して、JDK をインストールする。

csun@csun-desktop:/tmp/vmware-tools-distrib\$ sudo apt-get install sun-java5-jdk

コンフィグ画面では↓キー押して全部確認して、→キーを押して Enter キーで確定する。

# | このパッケージをインストールするためには、"Operating System Distributor | License for Java" (DLJ) バージョン 1.1 ライセンス条項に同意する必要があ | ります。同意しない場合はインストールをキャンセルします。 | DLJ ライセンス条項に同意しますか? | 〈はい〉 〈いいえ〉

"はい"を選択して、Enter キーを押す。 これで IDK のインストールが始まる。

```
⊗ S csun@csun-desktop: ~
ファイル(F) 編集(E) 表示(V) 端末(T) ヘルプ(H)
(.../odbcinst_2.2.11-21_i386.deb から) odbcinst を展開しています...
未選択パッケージ odbcinst1debian1 を選択しています。
(.../odbcinst1debian1_2.2.11-21_i386.deb から) odbcinst1debian1 を展開しています
未選択パッケージ unixodbc を選択しています。
(.../unixodbc_2.2.11-21_i386.deb から) unixodbc を展開しています...
未選択パッケージ sun-java5-bin を選択しています。
(.../sun-java5-bin_1.5.0-19-0ubuntu0.9.04_i386.deb から) sun-java5-bin を展開し
ています..
sun-java5-jre 1.5.0-19-0ubuntu0.9.04 を (.../sun-java5-jre_1.5.0-19-0ubuntu0.9.0
4_all.deb で)置換するための準備をしています ...
sun-dlj-v1-1 license has already been accepted
sun-java5-jre を展開し、置換しています...
未選択パッケージ gsfonts-x11 を選択しています。
(.../gsfonts-x11_0.21_all.deb から) gsfonts-x11 を展開しています...
desktop-file-utils のトリガを処理しています ...
python-gmenu のトリガを処理しています
Rebuilding /usr/share/applications/desktop.ja_JP.utf8.cache...
shared-mime-info のトリガを処理しています ...
fontconfig のトリガを処理しています ...
python-support のトリガを処理しています ...
man-db (2.5.7-2) を設定しています ...
Building database of manual pages
```

インストール完了。



# 2.2.6 その他のツールのインストール

#sudo apt-get install build-essential

#sudo apt-get install zliblg-dev

#sudo apt-get install flex

#sudo apt-get install libx11-dev

#sudo apt-get install gperf

#sudo apt-get install libncurses5-dev

#sudo apt-get install bison

インストール完了。



# 第三章 Cross-compile のインストール

下記良く使う二つのクロスコンパイルツールをインストールする。

- 1, arm-none-linux-gnueabi-arm-2008q3-72-for-linux.tar.bz2
- 2, cross-4.2.2-eabi.tar.bz2

/usr/local/arm/にインストールする。

お客様のニーズによって他のバージョンのツールを選択しても良いですが、下記手順をご 参照ください。

必要ツール取得:

製品と一緒同梱される DVD に下記フォルダーにある。

Linux v0.19\finux2.6.28\frac{1}{2}cross compile:

- ①cross-4. 2. 2-eabi. tar. bz2
- ②arm-none-linux-gnueabi-arm-2008q3-72-for-linux.tar.bz2

XP 環境で cross-4.2.2-eabi.tar.bz2 と

arm-none-linux-gnueabi-arm-2008q3-72-for-linux.tar.bz2 を ubuntu の共有フォルダ ubuntu\_share\_folder/toolchain/にコピーする。

```
csun@csun-desktop:/mnt/hgfs/ubuntu_share_folder/toolchain$ ls
arm-none-linux-gnueabi-arm-2008q3-72-for-linux.tar.bz2
cross-4.2.2-eabi.tar.bz2
csun@csun-desktop:/mnt/hgfs/ubuntu_share_folder/toolchain$
```

- 1、/home/csun/にtoolchainフォルダを作成する。
- 2、/mnt/hgfs/ubuntu\_share\_folder/toolchain/共有フォルダにある

cross-4.2.2-eabi.tar.bz2と

arm-none-linux-gnueabi-arm-2008q3-72-for-linux.tar.bz2を

/home/urbetter/ toolchain/フォルダにコピーする。

2、ハイパターミナルで/home/urbetter/toolchain/に入る。

```
csun@csun-desktop:/mnt/hgfs/ubuntu_share_folder/toolchain$ mkdir /home/csun/tool
schain
csun@csun-desktop:/mnt/hgfs/ubuntu_share_folder/toolchain$ cp -rf *.*
csun@csun-desktop:/mnt/hgfs/ubuntu_share_folder/toolchain$ cd
csun@csun-desktop:~$ ls
Ubuntu One
                                                     テンプレート
                                                     デスクトップ
arm-none-linux-gnueabi-arm-2008q3-72-for-linux.tar.bz2
                                                    ドキュメント
                                                     ビデオ
cross-4.2.2-eabi.tar.bz2
examples. desktop
                                                     ピクチャ
                                                     ミュージック
toolschain
ダウンロード
                                                     公開
csun@csun-desktop:~$
```



- 3、arm-none-linux-gnueabi-arm-2008q3-72-for-linux.tar.bz2をインストールする。
- **\$ tar xvf arm-none-linux-gnueabi-arm-2008q3-72-for-linux.tar.bz2** カレントフォルダに解凍されたツールを/usr/local/にコピーする。
- \$ sudo cp usr/local/arm/ /usr/local/ -a

```
csun@csun-desktop: $
csun@csun-desktop: $
csun@csun-desktop: $
sudo cp usr/local/arm/ /usr/local/ -a
[sudo] password for csun:
csun@csun-desktop: $
```

- 4、cross-4.2.2-eabi.tar.bz2をインストールする。
- **\$ tar xvf cross-4.2.2-eabi.tar.bz2**カレントフォルダに解凍されたツールを/usr/local/armにコピーする。
- \$ sudo cp 4.2.2-eabi/ /usr/local/arm/ -a

```
csun@csun-desktop:~$ sudo cp 4.2.2-eabi/ /usr/local/arm/ -a
csun@csun-desktop:~$
```

5、インストールした結果を確認する。

```
csun@csun-desktop:~$ ls /usr/local/arm
4.2.2-eabi arm-none-linux-gnueabi
csun@csun-desktop:~$
```

二つとも正しいフォルダにインストールされている。

# 第四章 ソースのコンパイル

## ソースは下記三つ:

1, u-boot : ut6410-uboot-v2.0-20101004.tgz

2 kernel: ut6410-kernel-v2.0-20101004.tgz

3, android2. 1: ut6410-android2. 1-v2. 0-20101004. tgz

上記ソースは下記から取得できる

\*製品と一緒同梱される DVD に下記フォルダーにある。

Linux v0. 19\(\frac{1}{2}\) and roid -2. 1\(\frac{1}{2}\) ut 6410 - and roid 2. 1\(\frac{1}{2}\) source:

ソースコードを/home/urbetter/ut6410/にコピーする。

# 4.1 u-boot のコンパイル

# u-bootは二つある:

- 1、 SDboot. bin: SDカードに書き込んで、SDカードからの起動を実現する。
- 2、 u-boot.bin: nand flashに書き込んで、nandからの起動を実現する。

二つとも同じソースコードで一箇所の定義で区別する:

Include/configs/smdk6410.h

```
447
448 #define CONFIG_BOOT_NAND //boot from nandflash
449 //#define CONFIG_BOOT_MOVINAND //boot from SD card
450
```

\$tar xvf ut6410-uboot-v2.0-20101004.tgz

\$cd ut6410-uboot-v2.0-20101004

\$make distclean

\$make smdk6410\_config

\$make

実行後、カレントフォルダにu-boot.binファイルが生成される。上記定義でSDカードに設定した場合はSDboot.binにネームを変更する。

## 4.2 カーネルのコンパイル

Sourceのフォルダに既にコンフィグしている. configがある。

UT\_LCD43C. config: 4.3インチ (480×272) 液晶タイプに適用

UT\_LCD7B. config: 7/10.2インチ (800×480) 液晶タイプに適用

\$tar xvf ut6410-kernel-v2.0-20101004.tgz

\$cd ut6410-kernel-v2.0-20101004

\$make clean

\$cp UT\_LCD43C.config .config



#### \$make

実行後生成されたzImageはarch/arm/boot/フォルダにある

# 4.3 android2.1のコンパイル

# 下記コマンドを実行する:

1、ソースコードを解凍する。

\$tar xvf ut6410-android2.1-v2.0-20101004.tgz

2、ソースコードフォルダに入る。

\$cd ut6410-android2. 1-v2. 0-20101004

- 3、環境変数を設定する(※二つの点の間スペースがある)。
- \$. ./build/envsetup.sh
- 4、オプションを設定する。

#### \$tapas

csun@csun-desktop: /ut6410/ut6410-android2.1-v2.0\$ . ./build/envsetup.sh csun@csun-desktop: /ut6410/ut6410-android2.1-v2.0\$ tapas Build for the simulator or the device?

- Device
- 2. Simulator

# Which would you like? [1] ∏

Enter キーを押す。

# Build type choices are:

- release
- 2. debug

Which would you like? [1]

Enter キーを押す。

# Which product would you like? [ut6410]

Enter キーを押す。

## Variant choices are:

- user
- 2. userdebug
- 3. eng

Which would you like? [eng]

設定結果:



PLATFORM\_VERSION\_CODENAME=REL
PLATFORM\_VERSION=2.1-update1
TARGET\_PRODUCT=ut6410
TARGET\_BUILD\_VARIANT=eng
TARGET\_SIMULATOR=false
TARGET\_BUILD\_TYPE=release
TARGET\_ARCH=arm
HOST\_ARCH=x86
HOST\_OS=linux
HOST\_BUILD\_TYPE=release
BUILD\_ID=ECLAIR

5、 ソースコードをコンパイルする。

#### \$mm

実行後生成された image ファイルは out/target/product/ut6410/フォルダにある。

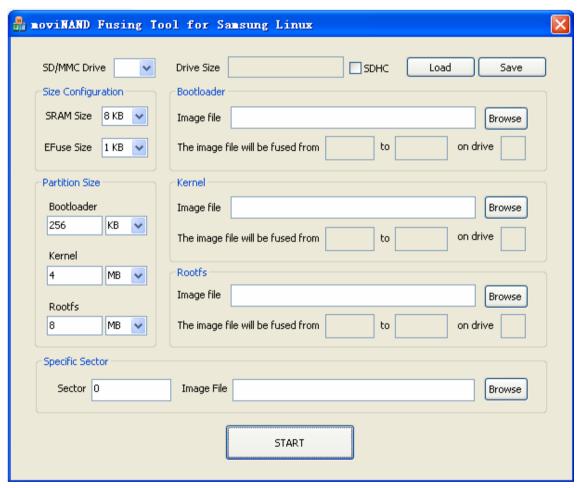
- 6、 yaffs イメージファイル ut6410\_root. img を作成する。
- \$./make\_ut6410\_yaffs2\_image.sh

カレントフォルダに ut6410\_root.img ファイルが生成される。

ロ見テクリョジーから可能にする

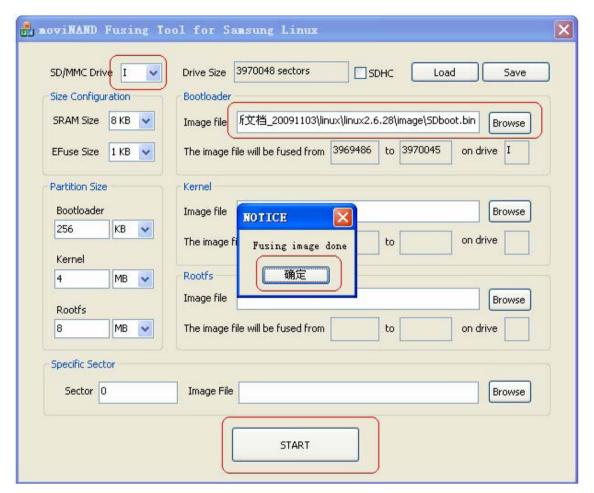
# 第五章 イメージファイルの書き込む

- 5.1 SD 起動用カードの作成
- 5.1.1 注意事項
- 1、2G Bytes 以内のSDカードをご利用ください。
- 1G/2G Kingston 或いは Sandisk の SD カードをお勧めします。
- 2、SDbootをSDカードに書込み時、SDカードリーダをご利用ください。一部のノートPC内蔵のリーダは正常に書き込めない恐れがあります。
- 5.1.2 SD 起動用カードの作成手順
- 1、SD カードをSD カードリーダを通じて、windows xp 環境でFAT32 にフォーマットする。
- 2、windows xp 環境で moviNAND\_Fusing\_Tool. exe を実行する。



- 3、 SDboot. bin を SD カードに書き込む
- ◆SD/MMC Driver の所に Windows XP 環境で SD カードのドライブを選択する。
- ◆Image file の所 "Browse"をクリックして SDboot. bin を選択する。
- ◆ "START" をクリックする。





- ◆書込み成功の場合 "Fusing image done" の提示画面が表示する。「確定」クリックして 完了。
- ※書込み成功しても、SDカードには書き込んだデータが見えない、容量も変更しない。

## 5.2 uboot の書き込み

SD カードから SDboot を起動して、SDboot のコマンドで USB OTG ポート通じて u-boot. bin を nand flash に書き込む。

## 5.2.1 SD 起動モードに設定

idea6410の設定:SDブート:SW1を「1111」に設定

# 5.2.2 uboot.bin の書込み

DNW でダウンロードして "u-boot. bin"を下記の操作条件のように書き込む。

\*Android をインストール前に、OS は WinCE がなった場合、Eboot で「A」コマンドですべて Nand Flash をフォーマットしてください。



# 1.【操作条件】

- ① SDブート状態:1111
- ②SD1 カードを ARM11 ボードに入れてください。
- ③付属 USB ケーブルで PC と ARM11 を接続し、PC と接続したら、USB ドライバをインストールしてください。USB Driver 場所: ¥tools¥usb-driver (secusb2. sys、secusb2. inf)

U-Boot を起動する (「Hit any key to stop autoboot」が出来たら、Enter キーを押す)

```
U-Boot 1.1.6 (Aug 3 2009 - 09:41:16) for SMDK6410
************
     UT-S3C6410/idea6410 SD boot v0.19
     ShenZhen Urbetter Technology
     Http://www.urbetter.com
**********
CPU:
        S3C6410@532MHz
        Fclk = 532MHz, Hclk = 133MHz, Pclk = 66MHz, Serial = CLKUART (SYNC Mode)
        SMDK6410
Board:
DRAM:
        128 MB
Flash:
NAND:
        256 MB
*** Warning - bad CRC or NAND, using default environment
In:
        serial
Out:
        serial
Err:
        serial
        Not Found CS8900@0x18800300
Net:
Hit any key to stop autoboot: 0
SMDK6410 #
SMDK6410 #
SMDK6410 #
```



2. コマンド「dnw c0008000」を実行(最後 Enter キーを入力)

```
SMDK6410 #
SMDK6410 #
SMDK6410 # dnw c0008000
OTG cable Connected!
Now, Waiting for DNW to transmit data
```

3. dnw を起動

USB Port -> Transmit -> Transmit

4. u-boot.binを開く

```
SMDK6410 #
SMDK6410 # dnw c0008000
OTG cable Connected!
Now, Waiting for DNW to transmit data
Download Done!! Download Address: 0xc0008000, Download Filesize:0x30000
Checksum is being calculated.
Checksum O.K.
SMDK6410 #
SMDK6410 #
```

5. ubootを nand flash 0に書き込む

コマンド「nand erase 0 40000」、「nand write c0008000 0 40000」を実行 (Enterを入力)

```
SMDK6410 #
SMDK6410 #
SMDK6410 # nand write c0008000 0 100000
NAND write: device 0 offset 0x0, size 0x100000
1048576 bytes written: OK
SMDK6410 #
```

書込み終了後、SW1を1100 に設定してnandflash起動モードに設定する。 ボードを再起動するとnandflashから起動する。

日星テクノロジーから可能にする

### 5.2.2 zImage ファイルの書込み

### \*Nand Flash モードで再起動して、uboot と同じ方法で書き込む

- 1. コマンド「dnw c0008000」を実行(最後Enterキーを入力)
- 2. dnw を起動 USB Port -> Transmit -> Transmit
- 3. zImageを開く
- 4. nand erase 900000 300000 nand write c0008000 900000 300000

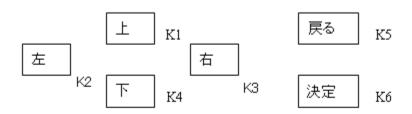
#### 5.2.3 androidの書込み

"ut6410\_root.img" を書き込む。

- 1. dnw c0008000
- 2. dnw を起動 USB Port -> Transmit -> Transmit
- 3. ut6410\_root.imgを開く
- 4. nand erase C00000 F400000
  nand write.yaffs c0008000 C00000 xxxxx
  xxxxxは実際ダウンロードしたファイルのサイズ。

Androidを初めて起動する場合、時間がかかります。起動後、1分間でスリープ状態に入ります。そのため、起動後、設定を変更必要です。

Androidのメイン画面で、「K6」(決定ボタン)を押し、 "Settings"を選択し、 "Sound & Display"ー》 "Screen timeout"ー》 "15 seconds" \*ARM11ユーザーボタンはAndroidが起動の際に操作イメージを下記の図に参照





## 第六章 Windows XP上のUSB ADB インストール

#### ノート:

2010 年 9 月前に Android2.1 をインストールされた場合のみ、USB ADB ドライバを独自 インストール必要ですが、2010 年 9 月以降購入された場合、新しいカーネルソース(イメ ージファイル)に USB ADB ドライバを既に入れられますので、本章を飛ばしてもよい。

### 5.1 USB ADB に関わるカーネル設定の修正

5.1.1. USB ADB ドライバソース修正

ADB をサポートするため、カーネルの USB ドライバソースを下記のように修正。

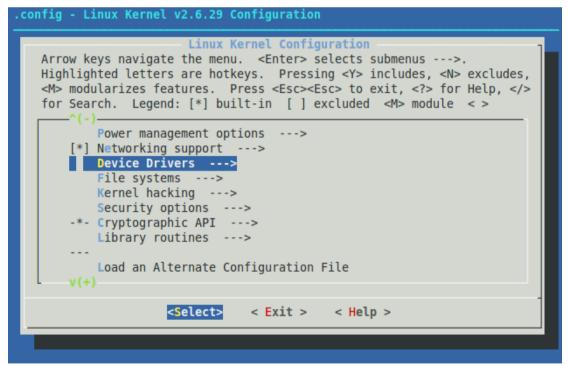
①USB ADB ドライバをダウンロード

http://www.dragonwake.com/download/idea6410/android2.1/Android2.1\_USB\_ADB\_Driver.zip

②上記ダウンロードしたファイルを解凍し、中身の「gadget. zip」を解凍してカーネルの ソースフォルダー「drivers¥usb¥gadget」を差し替え

#### 5.1.2. USB ADB に関するカーネル設定の修正

Device Drivers →





USB support →

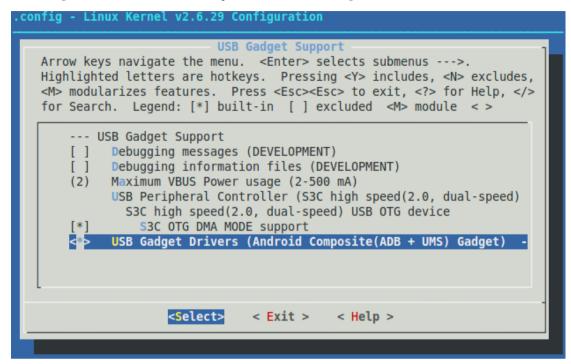
```
config - Linux Kernel v2.6.29 Configuration
                             Device Drivers
  Arrow keys navigate the menu. <Enter> selects submenus --->.
  Highlighted letters are hotkeys. Pressing <Y> includes, <N> excludes,
  <M> modularizes features. Press <Esc> to exit, <?> for Help, </>
  for Search. Legend: [*] built-in [ ] excluded <M> module < >
          Graphics support --->
      <*> Sound card support --->
      [*] HID Devices --->
      [*] USB support --->
      <*> MMC/SD/SDIO card support --->
      < > Sony MemoryStick card support (EXPERIMENTAL) --->
      [ ] Accessibility support --->
      [*] LED Support --->
      <*> Switch class support --->
      <*> Real Time Clock --->
                    <Select>
                               < Exit > < Help >
```

USB Gadget Support ->

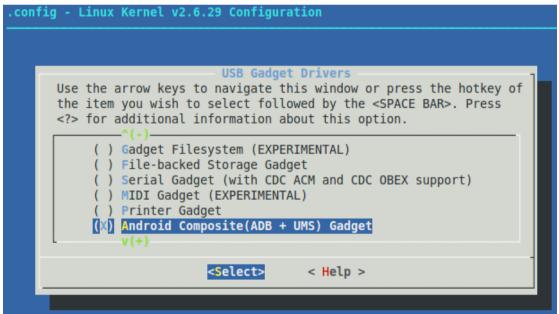
```
.config - Linux Kernel v2.6.29 Configuration
                               USB support
   Arrow keys navigate the menu. <Enter> selects submenus --->.
   Highlighted letters are hotkeys. Pressing <Y> includes, <N> excludes,
   <M> modularizes features. Press <Esc> to exit, <?> for Help, </>>
   for Search. Legend: [*] built-in [ ] excluded <M> module < >
       ^(-)-
       < >
            Apple Cinema Display support
            USB LD driver
            PlayStation 2 Trance Vibrator driver support
             IO Warrior driver support
            USB testing driver
       < >
             iSight firmware loading support
             USB VST driver
       <*> USB Gadget Support --->
            *** OTG and related infrastructure ***
            GPIO based peripheral-only VBUS sensing 'transceiver'
                     <Select>
                                < Exit >
                                            < Help >
```



USB Gadget Drivers (Android Composite (ADB+UMS) Gadget) ->



Android Composite (ADB+UMS) Gadget



保存し、カーネルをコンパイルし、zImage ファイルを ARM11 ボードにもう一回書込む \* 「<u>第四章 ソースのコンパイル</u>」と合わせてコンパイル場合、コンパイル、書込みが一回のみ

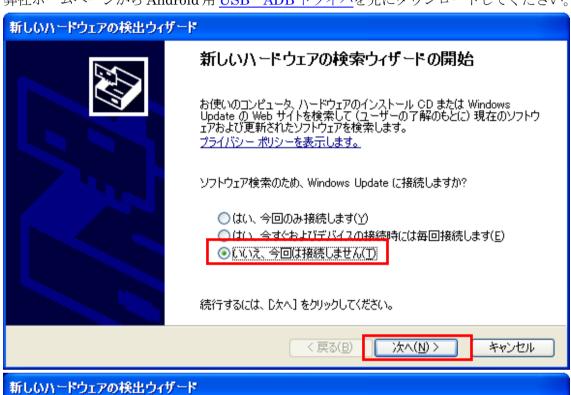


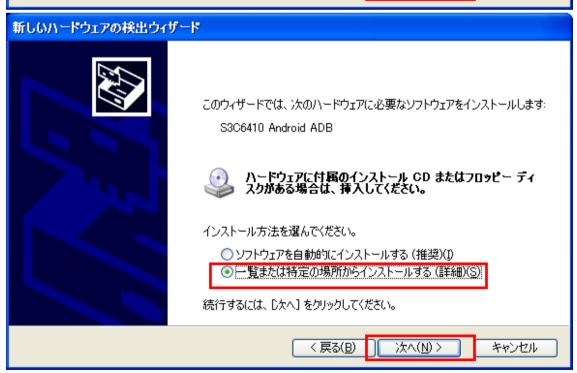
日昇テクノロジーなら可能にする

### 5.2 USB ADB ドライバーインストール

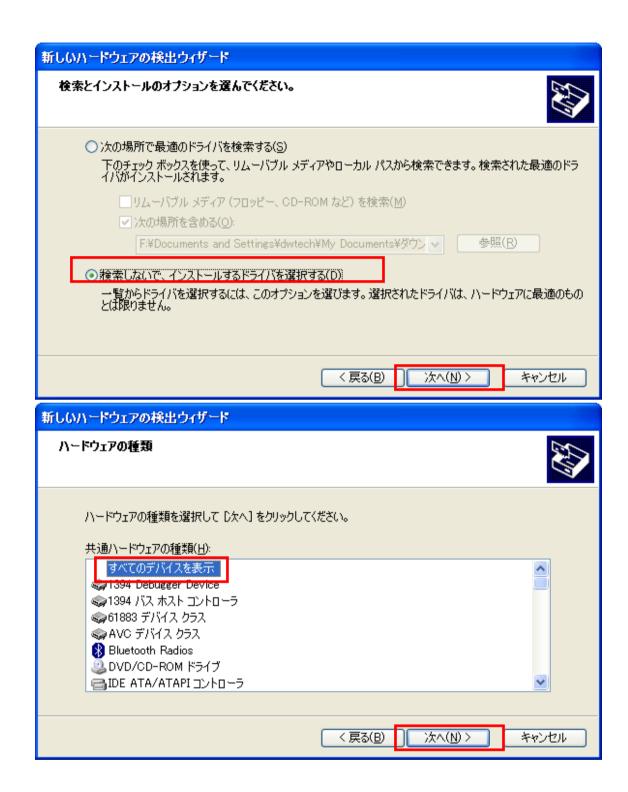
USB ケーブルで Idea6410 ボードの Mini-B USB インタフェースが PC の USB ポートと接続し、電源を入れ、Android を Nand Flash モードで起動させて、S3C6410 Android ハードウエアを見つかれ、ドライバーのインストールを下記のウィンドウのように要求される

弊社ホームページから Android 用 USB ADB ドライバを先にダウンロードしてください。

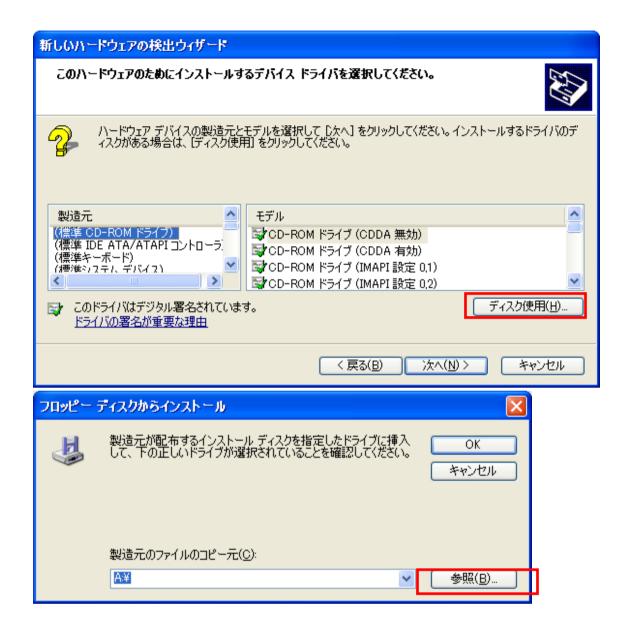




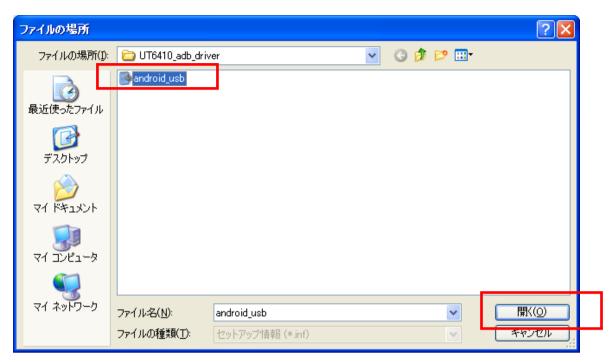


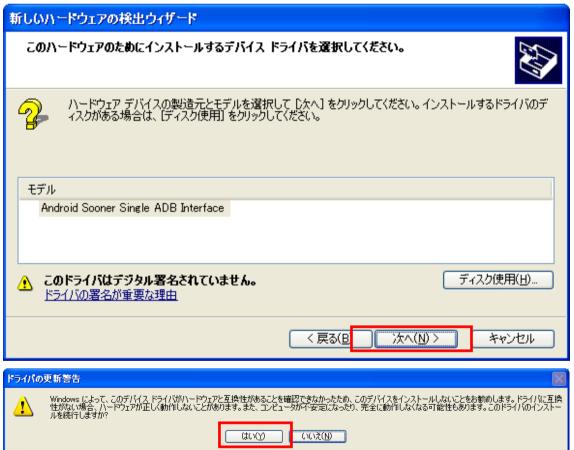












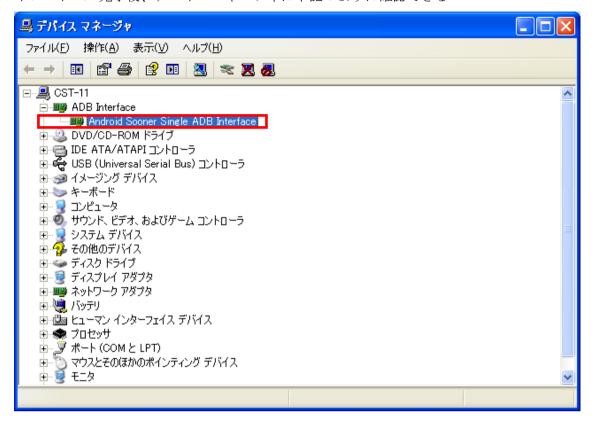








インストール完了後、デバイスマネージャに下記のように確認できる



Android 起動後、メッセージ欄に「USB connected」というメッセージが見られる



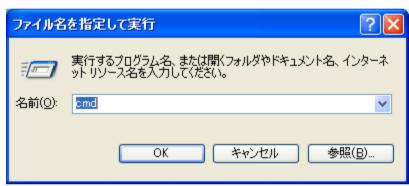
# 第七章 USB ADB 使用について

### 7.1 ADB コマンド

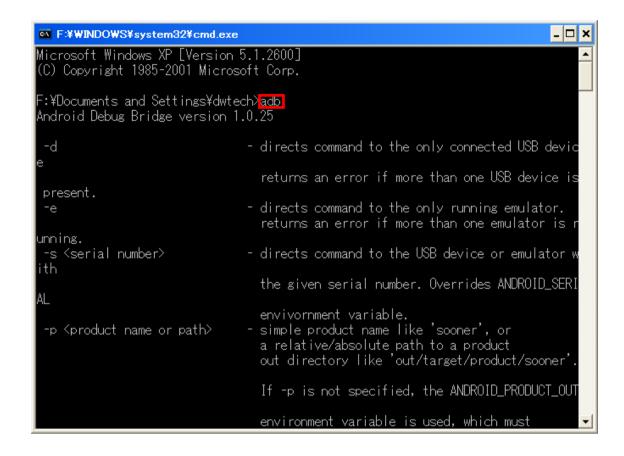
UT6410\_adb\_driver フォルダーから下記ファイルを「C:\Windows\system32」にコピーする

- ① adb. exe
- ② AdbWinApi.dll











adb は(Android のデバッグブリッジ)は、Android から一般的なデバッグツールとして提供して、我々はデバイスや携帯電話のシミュレータのステータスを管理できます。また、次の操作を行うことができます:

- 1、デバイスや携帯電話のシミュレータコードを素早くアップグレード、 例えば、Android のアプリケーションやシステムなどを更新
- 2、デバイス上でシェルコマンドとして実行
- 3、対象のポートで機器や携帯電話エミュレータを管理;
- 4、デバイスや携帯電話のエミュレータでファイルをコピーあるい貼り付け

ここではいくつかの一般的な操作があります:

①エミュレータにアプリケーションをインストールする:

adb install <path to apk></path to apk>

残念ですが、削除コマンドはないので、アプリケーションのアンインストールは手動で実 施

adb shell

cd /data/app

rm app.apk

②デバイスエミュレータのシェルに入る

adb shell

上記のコマンドで、デバイス、またはエミュレータのシェル環境に入る、 この Linux Shell では、さまざまな Linux シェルコマンドを実行できる。 一回ひとつ Linux コマンドを実施する場合、

adb shell [command]

例: adb shell dmesg カーネルのデバッグ情報を出力

③ポートをリリース

任意のポート番号をホストからシミュレータやデバイスへリクエストのポートとして 設定できる。

例:

adb forward tcp:5555 tcp:8000



日昇テクノロジーなら可能にする

### ④ファイルをコピーできる

1つのデバイスからコピーあるいはデバイスへファイルをコピーできる

1) ファイルあるいデバイスをシミュレータのディレクトリにコピー adb push <source> <destination></destination></source> 例:

adb push test.txt /tmp/test.txt

2) デバイスからファイルをコピー

adb pull <source> <destination></destination></source>
例:

adb pull /addroid/lib/libwebcore.so

⑤エミュレータあるいはデバイスインスタンスを検索

現在のデバイスインスタンスリストを取得と、該当インスタンスの状態を取得

adb devices

- ⑥バグレポートを参照
  - adb bugreport
- ⑦無線通信のログを記録
- 一般的に、無線通信のログは非常に大きく、実行時にはレコードには必要ありませんが、 レコードの取得も設定できる

adb shell

logcat -b radio

⑧デバイス ID とシリアル番号を取得

adb get-product
adb get-serialno

⑨データベース SQLite3 をアクセス

adb shell

sqlite3

\*



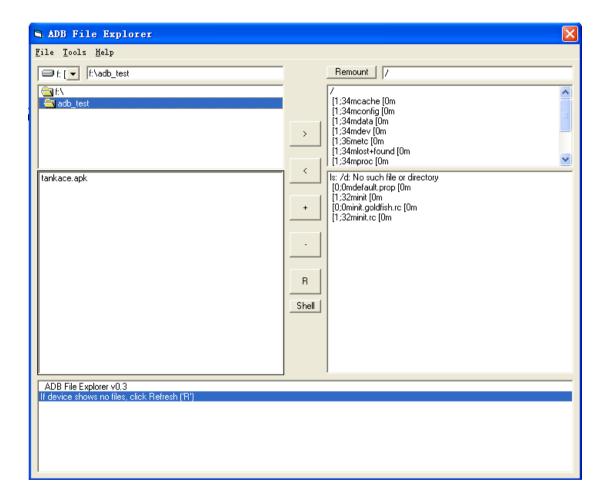
### 7.2 ファイルをアップロード/ダウンロード

「ADB File Explorer v03. exe」というツールを利用して Android デバイスへファイルをアップロード、また、Android でデバイスからファイルをダウンロードできる

1. ファイルをダウンロード

左: PC のパス(任意のパスを選択可能)

右: Android デバイスのパス (デフォルト「\」)





ダウンロードする際に、右のファイルを選べ、「<」ボタンを押下、「Yes」を選択してファイルは Android デバイスから PC の「F:\adb test」にダウンロードできる



### 2. ファイルをアップロード

左側からファイルを選べ、「>」ボタンを押下、ファイルは Android デバイスの「/」にアップロードした。



シリアルポート (ハイパーターミナル) で結果を確認



# 第八章 Android 開発環境構築(Windows 編)

AndroidとはGoogleを中心とした世界34社がオープン携帯プラットフォームを目指す団体Open Handset Allianceが開発 しているモバイルプラットフォームです。ここでは、そのプラットフォーム上での開発の仕方を紹介します。

まず、開発環境を整えます。ここでは Windows 上で Eclipse を利用して開発します。必要なアプリケーションは下記のとおりになります。

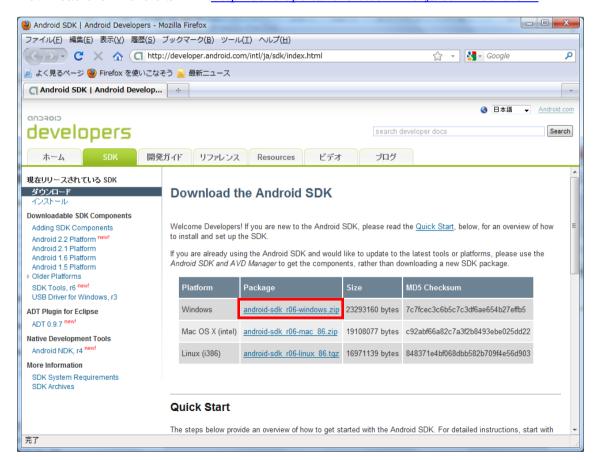
- Android SDK
- Eclipse
- Android Development Tools プラグイン

それぞれのインストール方法を次に示します。

### 8.1 Android SDK のインストール

下記からダウンロードします。

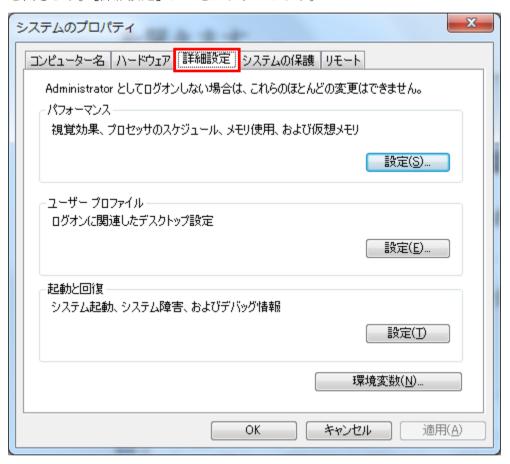
Download the Android SDK: http://developer.android.com/intl/ja/sdk/index.html





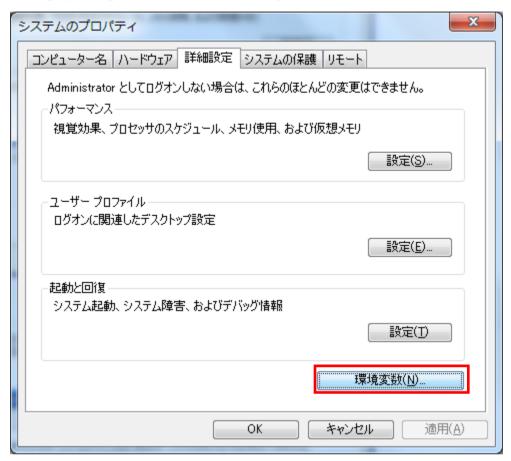
ダウンロードしたファイルを解凍し、解凍されたフォルダを任意の場所におきます。ここでは仮にそのフォルダを「 $\$SDK\_HOME$ 」とします。そして、そのパスを通す必要があります。

1. マイコンピュータを右クリックし [プロパティ] を選択して、[システムのプロパティ] を開きます。[詳細設定] タブをクリックします。



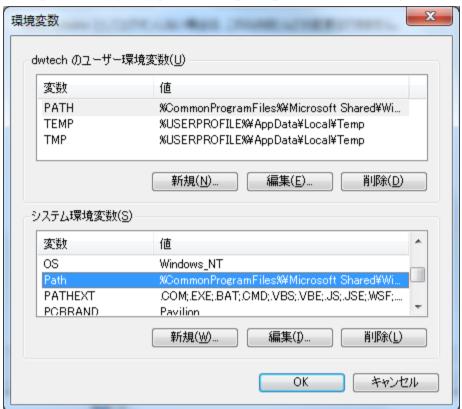


2. [環境変数] ボタンをクリックします。

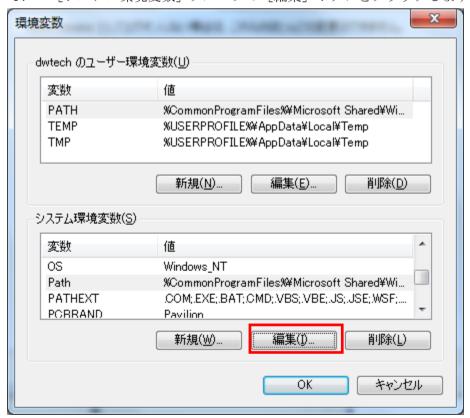




3. [システム環境変数] リストの [Path] をクリックします。

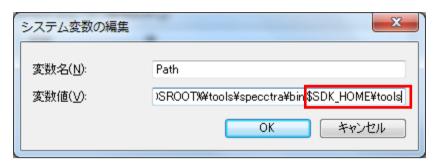


4. [システム環境変数] グループの[編集] ボタンをクリックします。

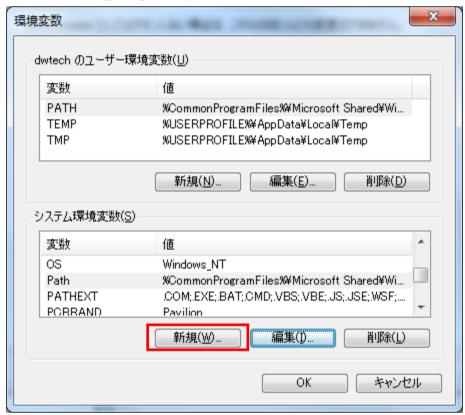




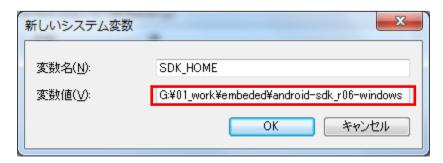
5. [変数値] テキストボックスの末尾に「\$SDK HOME¥tools」を追加します。



6. 変数「\$SDK\_HOME」の定義を追加



「 $\$SDK\_HOME$ 」が「 $G:\$01\_work\$embeded\$android-sdk\_r06-windows$ 」であれば、「 $G:\$01\_work\$embeded\$android-sdk\_r06-windows$ 」となります。

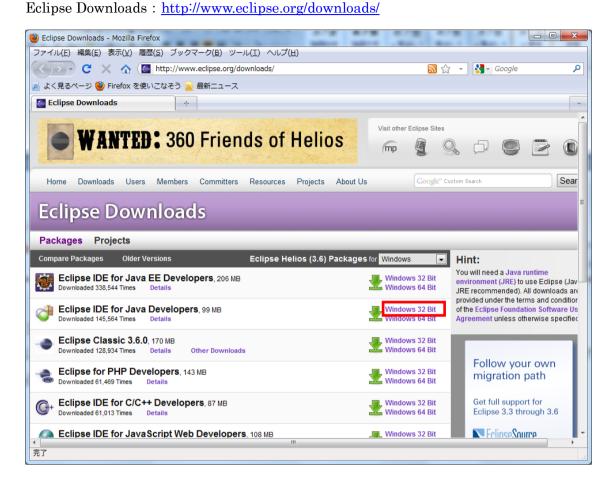


7. [OK] を押して、それぞれのダイアログボックスを閉じます。

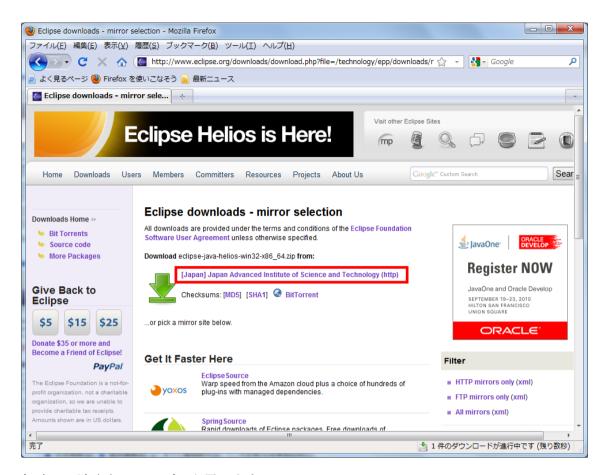


## 8.2 Eclipse のインストール

次に Eclipse をインストールします。最新版の Eclipse を下記よりダウンロードします。





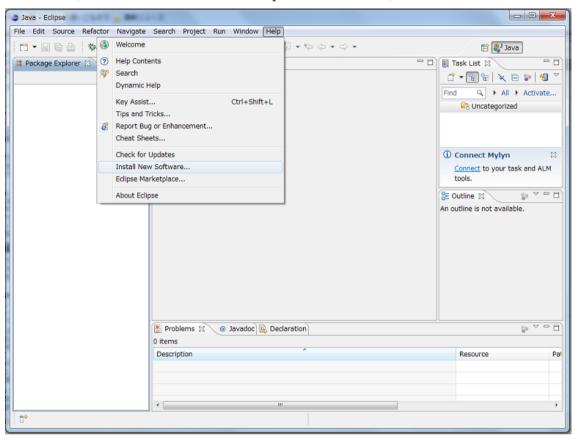


解凍して適当なフォルダに配置します。



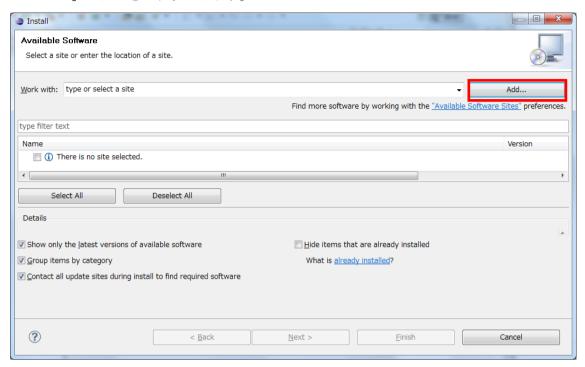
### 8.3 ADT プラグインのインストール

1. Android アプリケーションを効率よく開発するために Elipse のプラグインをインストールします。 先ほどインストールした Eclipse を起動し ます。

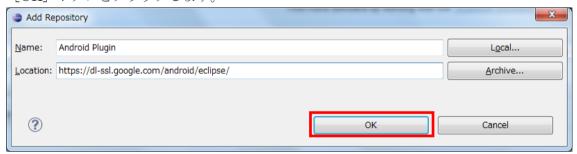




2.「Add」ボタンをクリックします。

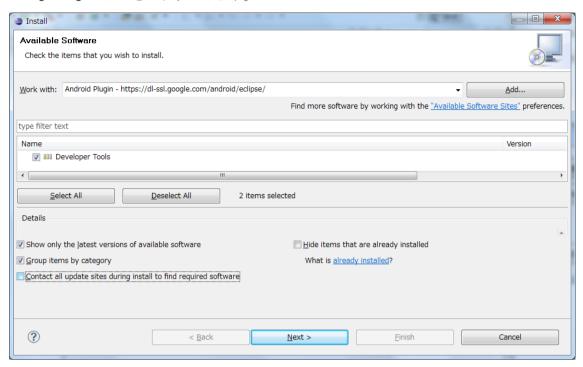


3. 新しいダイアログが開くので、Name 欄にリモートサイトの名前(例えば、Android Plugin) を入力します。URL 欄には「https://dl-ssl.google.com/android/eclipse/」を入力 し、 [OK] ボタンをクリックします。

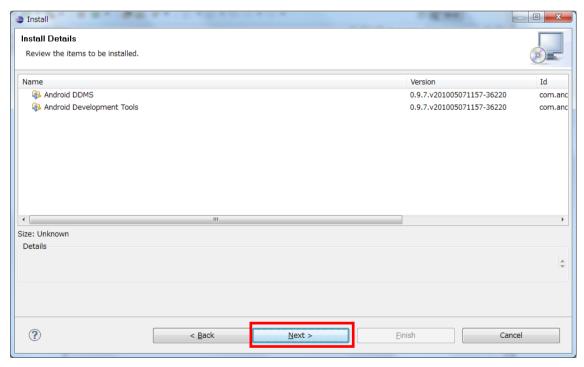




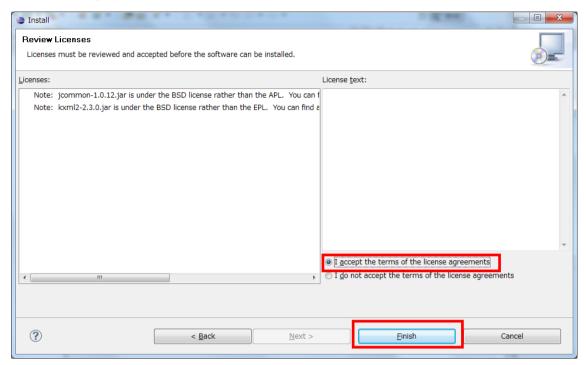
4. [Next] ボタンをクリックします。



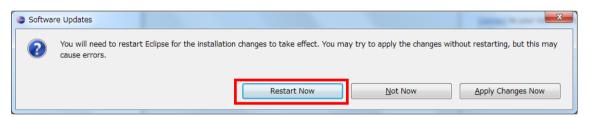
5. [Next] ボタンをクリックします。







7. 再起動するかどうかを聞かれるダイアログが表示されるので、[Yes] ボタンをクリックします。





日昇テクノロジーなら可能にする

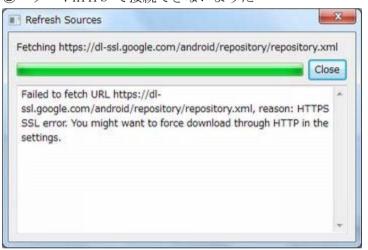
### 8.4 Eclipse 上サンプルの実行

試しに、SDKに付属しているサンプルを実行してみましょう。

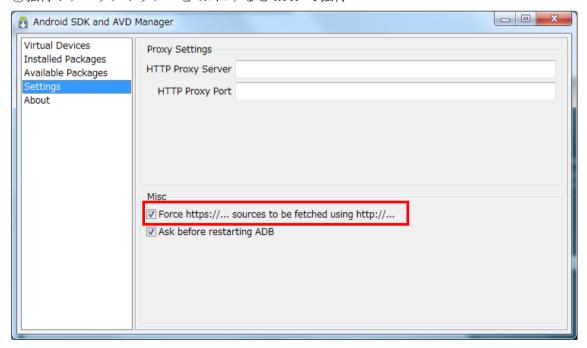
- 1. SDK マネジャー(サンプルを含む)をバージョンアップします。
- ①コマンドラインから「.¥android.bat update sdk」を実行します。
- あるいは「G:¥01\_work¥embeded¥android·sdk\_r06-windows」の直下「"SDK Setup.exe"」 プログラムを実行します。



②エラー: HTTPS で接続できないようだ



③強行:チェックボックスを ON にすると HTTP で強行

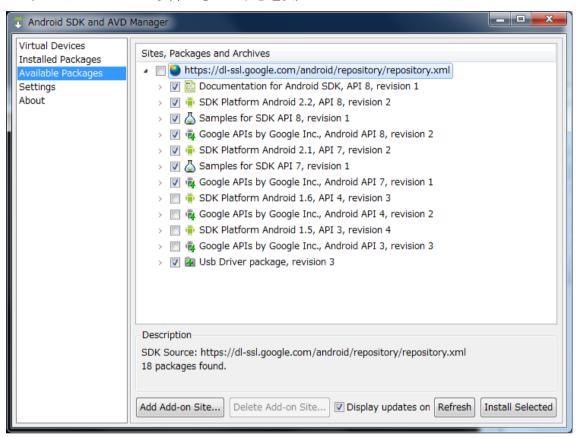






④必要のパッケージを選択してインストールします。

(ここに Android2.1 以降のもののみを選択)



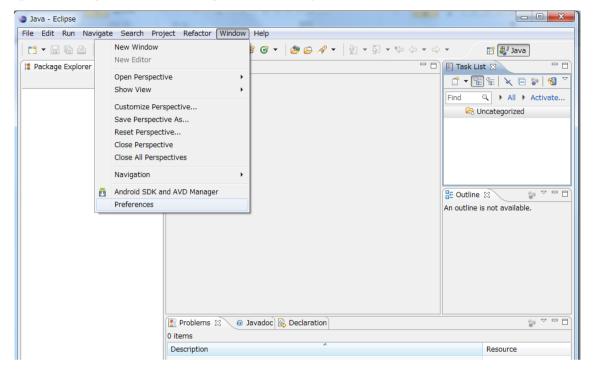
ホームページ: <a href="mailto:http://www.csun.co.jp">http://www.csun.co.jp</a> メール: <a href="mailto:info@csun.co.jp">info@csun.co.jp</a>







- 2. SDK をバージョンアップ完了後、SDK を設定します。
- ①「Window」→「Preferences」を選択します。



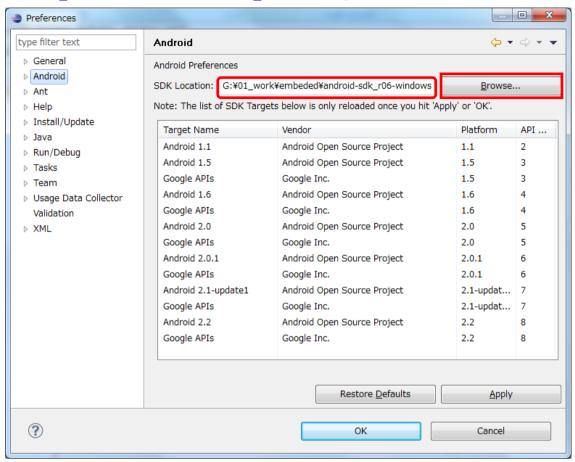
ホームページ: http://www.csun.co.jp  $\cancel{1}$  -  $\cancel{1}$ : info@csun.co.jp





②SDK のインストール先場所を指定します。

G:¥01 work¥embeded¥android-sdk r06-windows



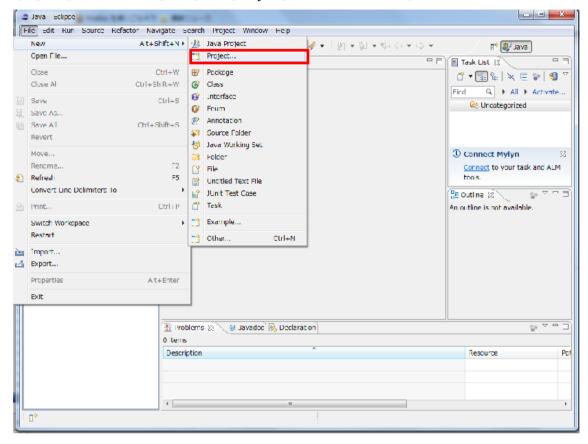
ホームページ: <a href="mailto:http://www.csun.co.jp">http://www.csun.co.jp</a> メール: <a href="mailto:info@csun.co.jp">info@csun.co.jp</a>







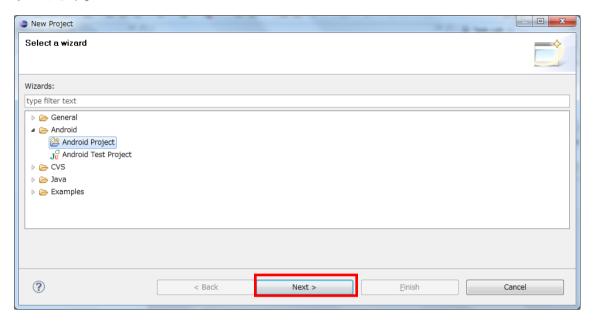
- 3. プロジェクトの作成
- ① [File] メニューの [New] [Project] をクリックします。



ホームページ: <a href="http://www.csun.co.jp">http://www.csun.co.jp</a>  $\cancel{x} - \cancel{\nu}$ : info@csun.co.jp



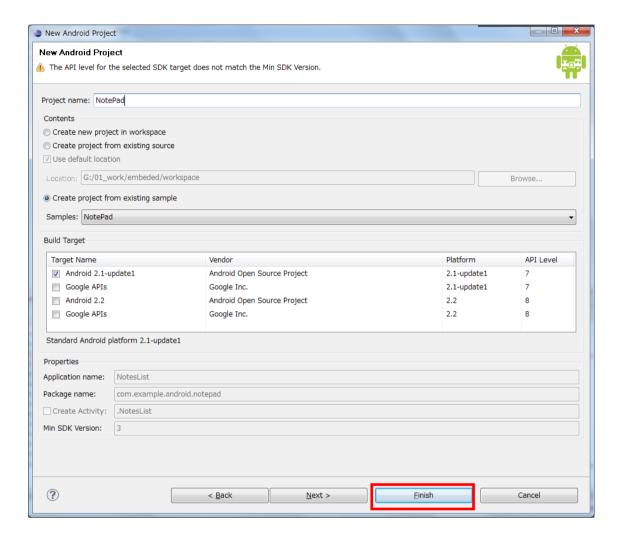
②ダイアログボックスが開くので [Android Project] を選択して、[Next >] ボタンをクリックします。



ホームページ: <a href="mailto:http://www.csun.co.jp">http://www.csun.co.jp</a> メール: <a href="mailto:info@csun.co.jp">info@csun.co.jp</a>



③ [Project Name] に「NotePad」と入力し、[Create Project from existing sample] を、 [Samples] に「NotePad」というサンプルを、Target Name に「Android 2.1-update1」を選択し、[Finish] ボタンをクリックします。

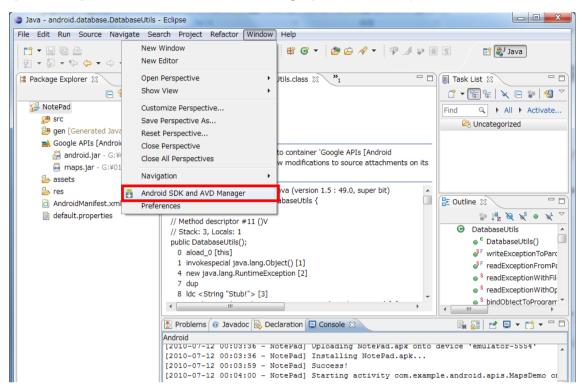


ホームページ: <a href="mailto:http://www.csun.co.jp">http://www.csun.co.jp</a> メール: <a href="mailto:info@csun.co.jp">info@csun.co.jp</a>

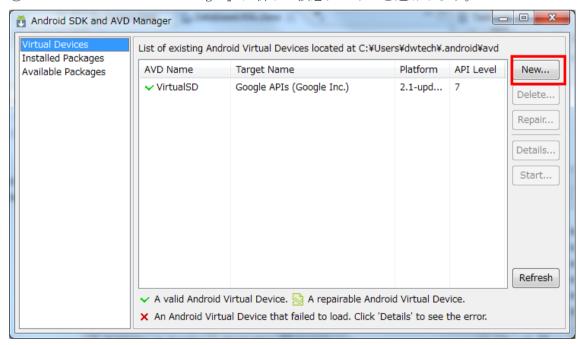
日昇テクノロジーなら可能にする



- 4. 「Virtual Devices」をインストールします。
- ①「Window」→「Android SDK and AVD Manager」を選択します。



②「Android SDK and AVD Manager」に新しい仮想デバイスを追加します。



ホームページ: http://www.csun.co.jp メール: info@csun.co.jp



日昇テクノロジーなら可能にする

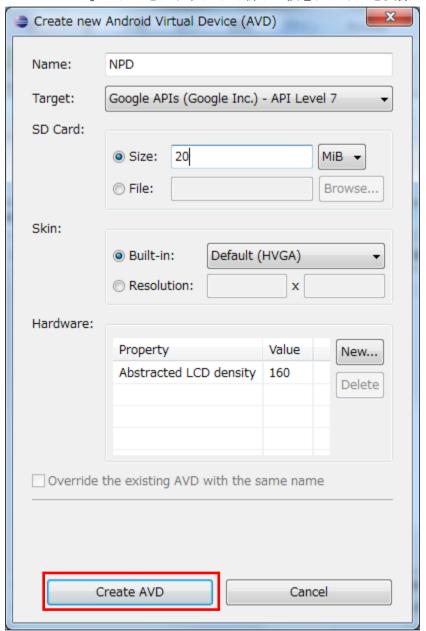
③新しい仮想デバイスを設定します。

Name: VirtualSD

Target: Google APIs(Google Inc.)-API Level 7

SD Card Size: 20M

「Create AVD」ボタンをクリックして新しい仮想デバイスを実際に作成します。

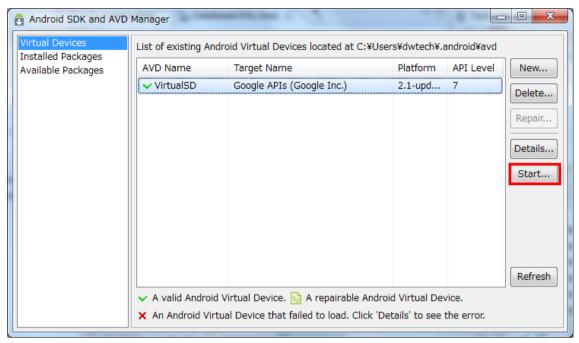


ホームページ: <a href="mailto:http://www.csun.co.jp">http://www.csun.co.jp</a> メール: <a href="mailto:info@csun.co.jp">info@csun.co.jp</a>



## ④仮想デバイスを起動します。

「Start...」ボタンをクリックし仮想デバイスを起動します。



#### 起動中の模様:



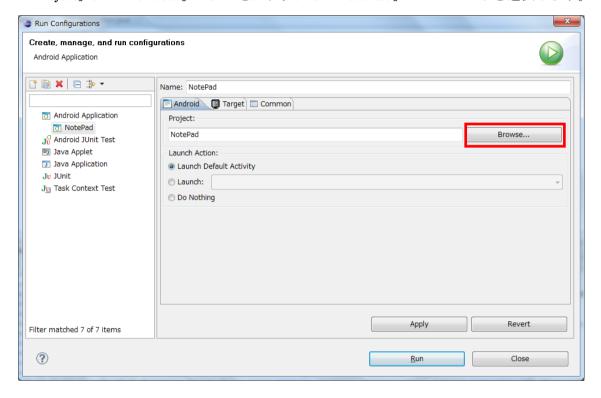






- 5. アプリケーションの実行
- ① [Run] メニューの [Run Configuration] をクリックします。
- ② [Android Application] を選択して、右ボタンメニューから [コンテキスト] メニュー
- の [New] をクリックし、[Name] テキストボックスを「NotePad」にします。

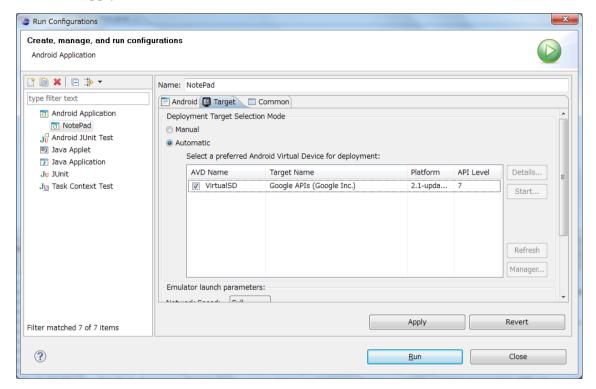
「Project」に「Browse...」ボタンをクリックし「NotePad」プロジェクトを選択します。



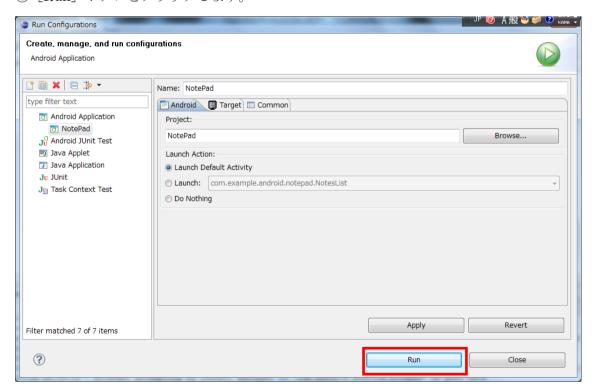




③「Target」タブにこの前作成した仮想デバイス「VirtualSD」をチェックします。 最後、「Apply」ボタンを押し設定を適用します。



④ [Run] ボタンをクリックします。





⑤エミュレータが起動し、アプリが実行されるので適当に操作してみてください。

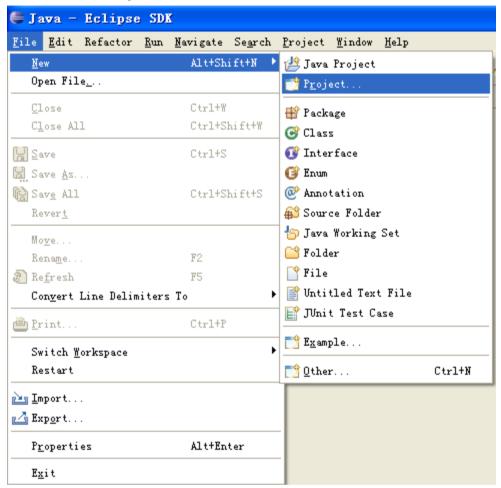


以上で開発ができる準備が整いました。



# 8.5 Android エミュレータでのデバッグ

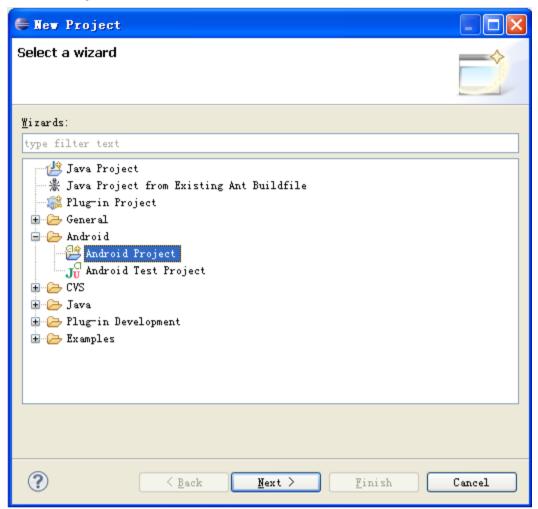
File -> New ->Project



 $\mathcal{A} - \mathcal{I} \mathcal{V} : \underline{\mathsf{info@csun.co.jp}}$ 



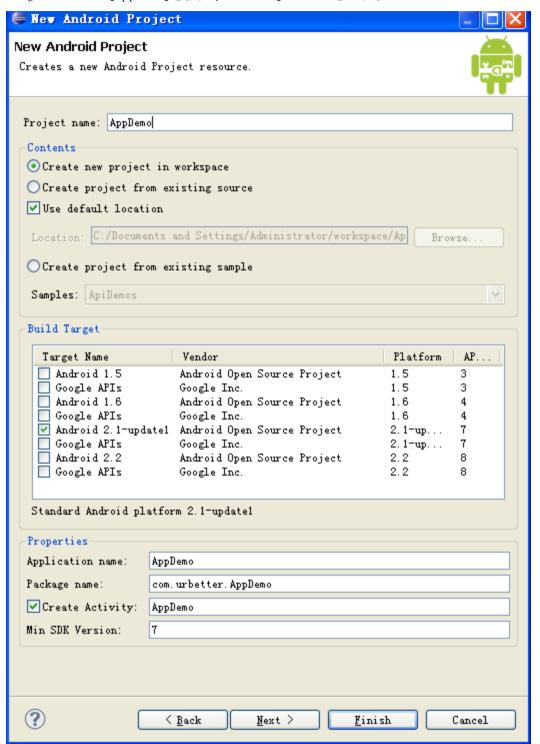
Android Project -> Next

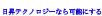


ホームページ: <a href="http://www.csun.co.jp">http://www.csun.co.jp</a>

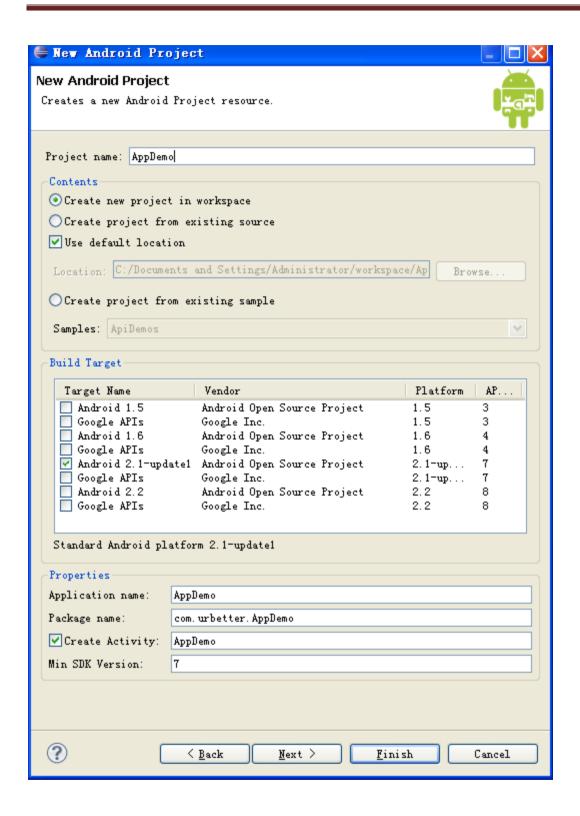


Project Name に[AppDemo]を記入、「Finish」ボタンをクリック



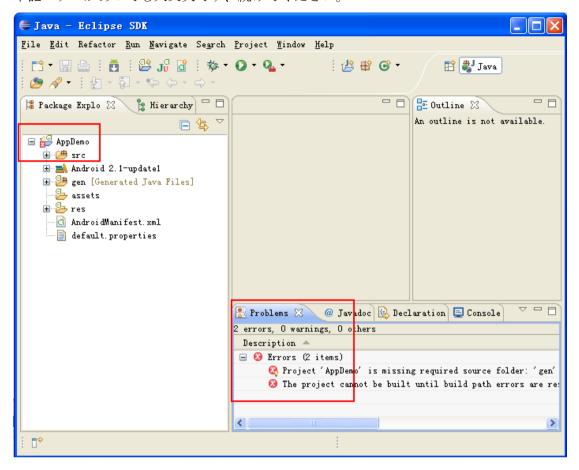








下記エラーがあっても大丈夫です、続けてください。

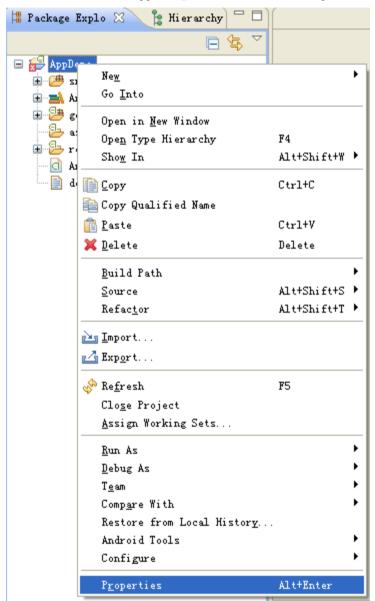


ホームページ: http://www.csun.co.jp  $\mathcal{A} - \mathcal{I} \mathcal{V}$ : info@csun.co.jp



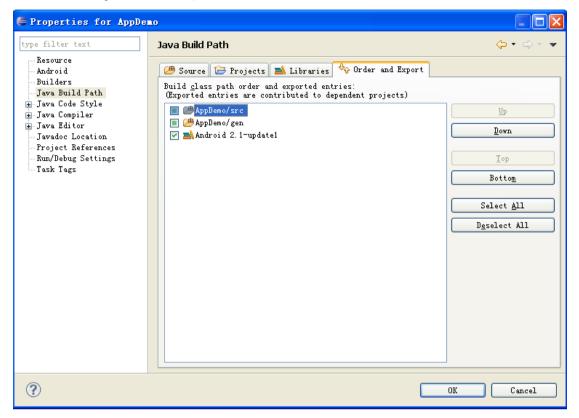


プロジェクト名前「AppDemo」を右クリック-> Properties





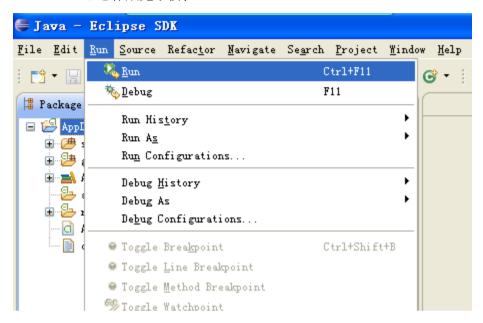
android 2.1-upate を選択し, ok ボタンをクリック



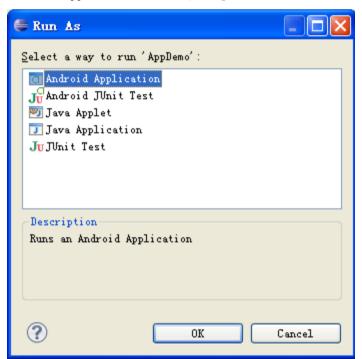




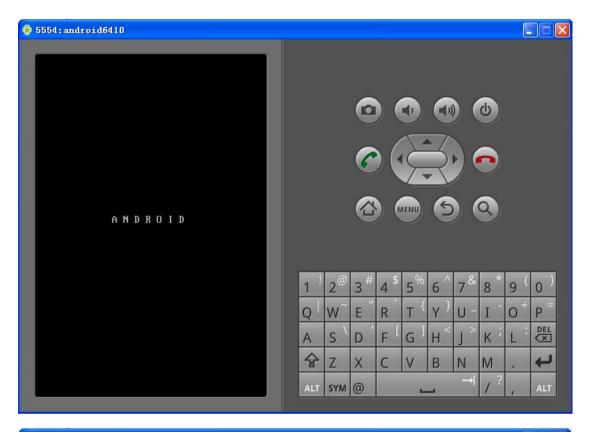
プロジェクトを作成完了後、Run -> Run



Android Application を選べ、「OK」ボタンを押下











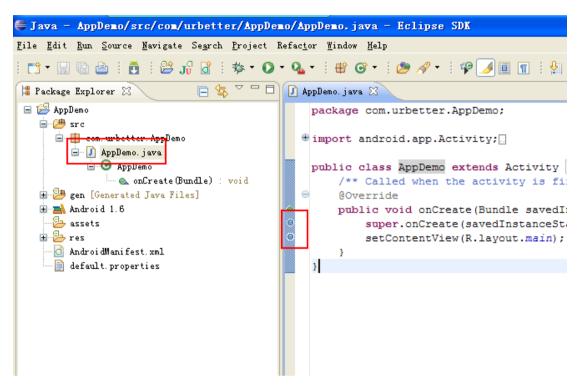


上記はAVD名前が「android6410」としている実行結果

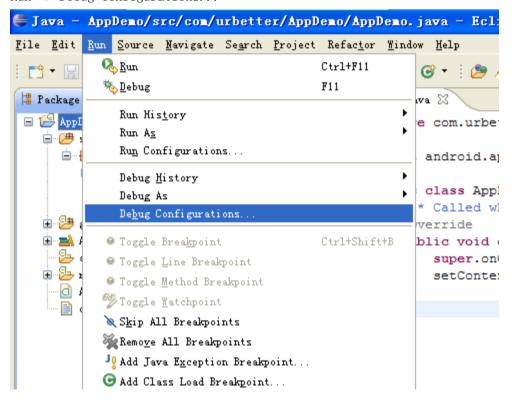




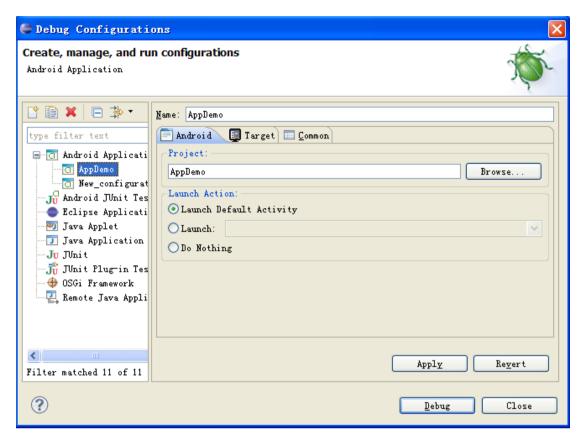
- 8.6 Android エミュレータでデバッグ
- 1. ブレークポイントを設定



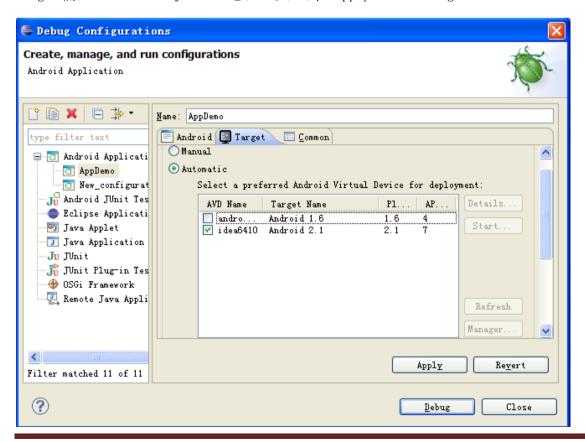
Run -> Debug Configurations...







Target 欄に「Android2.1」の AVD をチェックし、"Apply" -> "Debug"



ホームページ: http://www.csun.co.jp

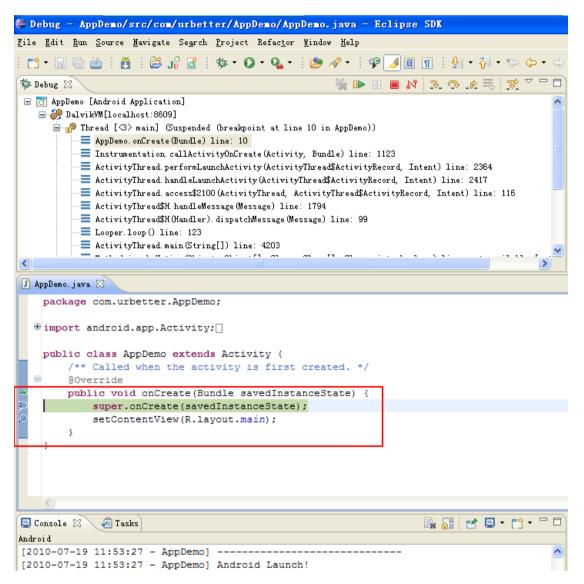
 $\mathcal{A} - \mathcal{N}$ : info@csun.co.jp



ホームページ: <a href="http://www.csun.co.jp">http://www.csun.co.jp</a> メール: <u>info@csun.co.jp</u>



#### ブレークポイントのところに止まる



「F6」キーを押しステップでデバッグ

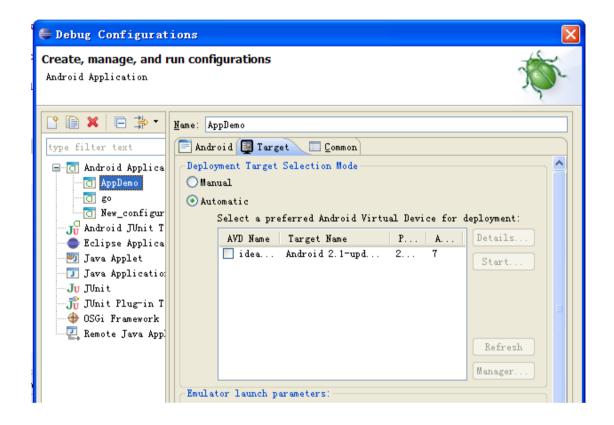


#### 8.6 ARM11 ボードの Android 実機にデバッグ

ARM11 ボードに電源を入れて Android を起動させ、USB ケーブルで PC と接続し、下記のようなメッセージが現れた。

"USB debugging connected"

AMR11 ボードの Android 実機上のデバッグはエミュレータでのデバッグと殆ど同じ、 デバッグ時、AVD を選択しなければ実機デバッグに入る





# 第九章 ARM11 ボードに Android アプリを実行

9.1 ARM11 ボードにアプリをインストール

Android のアプリケーションは、apk ファイル(Android package file)と呼ばれる zip 形式のアーカイブファイル。Android アプリケーションをインストールする方法は次の 3 パターン。

- 1. Android Market からインストール (開発アプリをインストールする方法ではない)
  - 1) [Application] タブから [Android Market] を選択
  - 2)全てのアプリケーションはカテゴリ毎に分かれているので、好きなアプリを選択してインストール。もちろん検索も可能。Android Marketには、人気のあるアプリケーションが紹介されている。
- 2. micro SD カードからインストール
  - 1)[Android Market]から"Apps Installer"や"ApkInstaller"等のインストーラをインストール
  - 2)実機にインストールしたいアプリケー ション (apk ファイル) を micro SD カード内に コピー。コピー場所はどこでも OK
  - 3)インス トールした[Apps Installer]や[ApkInstaller]等を起動
  - 4)表示されるメッセージに従ってインス トール
- 3. Android SDK 付属の Android Debug Bridge (adb)からインストール
  - 1) Windows PC と実機を USB ケーブルで接続。
    USB ドライバ は、第五章にすでにインストールされた。
  - 2)実機 で、[Application]タブから[設定]を選択して、[アプリケーション]-[提供元不明のアプリ]を有効にする
  - 3) "adb install <apk のフルパス>"を実行

ホームページ: http://www.csun.co.jp メール: info@csun.co.jp



#### 9.2 ARM11 ボードにアプリを動かす

本マニュアルは 7.1 の 2 番のインストール方法に基づき、開発した Android アプリを ARM11 ボードに実行します。

- 1) 実機にインストールしたいアプリケー ション (apk ファイル) を micro SD カード内に コピー。(ここに Android SDK 付属のサンプル「Snake」をコピーした)
- 2) シリアポートー>USB ケーブルを利用し下記のように ARM11 と PC を接続します。 \*シリアポートー>USB ケーブルがない場合、弊社二つ製品を利用できます。
  - ①USB Open-JTAG+RS232: http://www.csun.co.jp/SHOP/200905191.html
  - ②USB RS232 変換ケーブル(D サブオス):

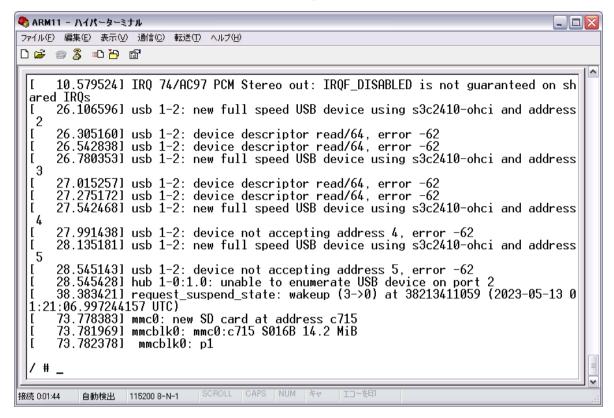
http://www.csun.co.jp/SHOP/2010040601.html

ハイパーターミナル設定:



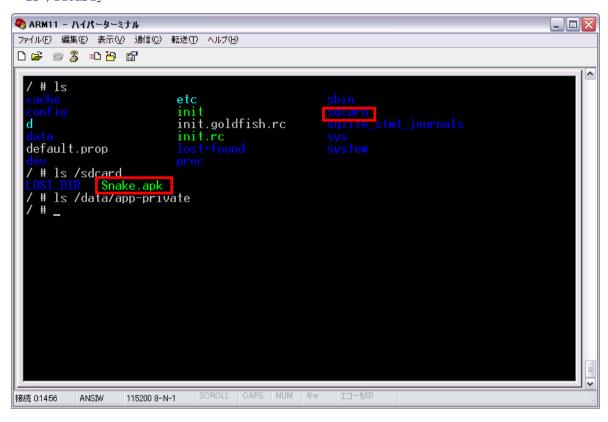


ARM11 ボードが起動出来た様子: (起動後「Enter」を 2-3 回押し root としてログイン)





3) SD カードを ARM11 ボードに差し込み、Android アプリをインストールします。
①SD カードを ARM11 ボードに差し込んで、ハイパーターミナルから SD 内容を確認します。
「1s /sdcard」





②インストール前、ARM11 ボードの Android のアプリケーション一覧に「Snake」アプリがない事を確認 (Home→メニューー>アプリケーション一覧)

操作キー: P49 を参照

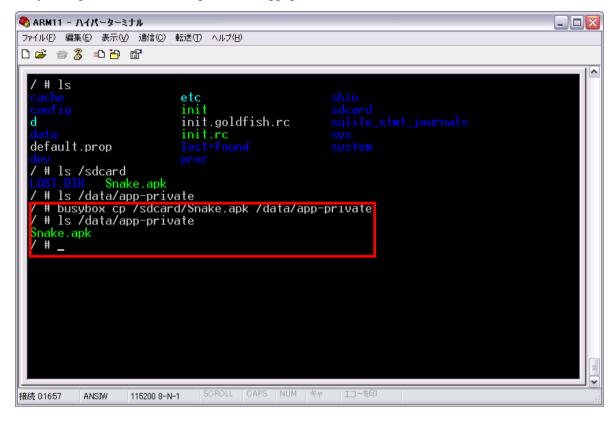




③アプリをインストール

インストールコマンド (コピー):

busybox cp /sdcard/Snake.apk /data/app-private





#### ④インストール済の事を確認



#### ⑤アプリを実行

「Snake on a Phone」アイコンをクリックしアプリ実行 起動中の様子





## 付録:ネットワーク設定

- 一、有線 LAN を設定
  - 1. 起動前にLANケーブルを付ける場合、IPは自動取得されますので、特に設定必要がありません。
  - 2. 起動後LANケーブルを付ける場合、手動で設定必要です。
    - ① LANケーブルを接続する時、

```
#
#
# eth0: link up, 100Mbps, full-duplex, lpa 0x45E1
# <mark>|</mark>
```

② ネットワークの状況を調べ、コマンド「netcfg」

```
# netcfq
10
         UP
                                 255.0.0.0
                                                  0x00000049
                127.0.0.1
eth0
         UP
                0.0.0.0
                                 0.0.0.0
                                                  0x00001043
tunl0
         DOWN
                0.0.0.0
                                 0.0.0.0
                                                  0x00000080
gre0
         DOWN
                0.0.0.0
                                 0.0.0.0
                                                  0x00000080
```

上記の結果により、IPは取れてない状況です。

- ③ コマンド「netcfg eth0 up」を発行
- ④ コマンド「netcfg eth0 dhcp」を発行
- ⑤ コマンド「netcfg」をもう一回発行

```
# netcfq
10
         UP
                127.0.0.1
                                 255.0.0.0
                                                  0x00000049
eth0
         UP
               192.168.1.113
                                 255.255.255.0
                                                  0x00001043
tun10
         DOWN
                                                  0x00000080
                0.0.0.0
                                 0.0.0.0
gre0
         DOWN
                0.0.0.0
                                 0.0.0.0
                                                  0x00000080
```

IPは割り当てられたことを明らかにした、ネットワークは接続できます。



最後: Android起動後の様子









## 二、無線LANを設定

- 1. 無線LANを探す
  - ① ARM11ボードをAndroidで起動してからメニューボタンを押下





② メニュー画面で「Setting」を押下、「Wireless & network」を選択し、「Wi-Fi」を押下 \*この手順には無線LANを使用できるようにする







③ 無線LANが見つかったら、下記の画面に表示される。



無線LANがある場合、ARM11ボードに見つからないであれば、 下記コマンドをハイパーターミナル(シリアルポートを通してPCと接続)で 実行してください。

#netcfg eth1 up

#iwlist eth1 scan

上記コマンドを発行された場合、無線LAN(AP)が出るはずですが、出ない場合、 無線LANの信号の強さをチェックしてください。

④番を実行した後、①~③を再度実施してください。



#### 2. 無線LANを設定

1番の手順を実施した後、セキュリティがある無線LANに対して、「Wi-Fi Settings」をクリックし、画面の指示に従ってキーを入力してください。

パブリックの無線LANの場合、自動的に無線LANを接続されます、何にも設定必要がありません。





#### 3. 無線LAN接続に関しての検証

上記設定は終わったら、ホームに戻して「Browser」を起動し、下記のような画面が出て来れば、無線LANを接続出来たことを明らかにします。無線LANでお使いましょう。

