



# Andes Game Platform Porting

Andes Technology



Architecture for Next-generation  
Digital Engines for SoC

# Outline



- ❖ Porting guide
- ❖ System Architecture
- ❖ Package dependency
- ❖ Game package details
- ❖ Performance issue
- ❖ The result of playing game on Andes platform
- ❖ The ways of enhancement performance

# Getting Started



## ❖ Environment

- Ubuntu 9.10
- BSP 2.0
- Andes tool chain v1.3.3
- A working target

# Porting Guide



- ❖ Demo open source applications porting for Andes platform
- ❖ There are following steps
  - Modify config.sub
  - configure and make

# Environment settings for Andes tool chain



## ❖ Set the location of your tool chain

- `source bashrc.nds32le-linux-V0`
- `export`  
`ANDESIGHT_ROOT=/home/path/toolchains/nd`  
`s32-elf-n1213-43u1h`
- `export`  
`PATH=$ANDESIGHT_ROOT/bin:$PATH`

# Modify config.sub



❖ Find the line below "Some are omitted here ..."

```
    | mn10200 | mn10300 \  
    | mt \  
    | msp430 \  
+   | nds32 | nds32le \  
    | nios | nios2 \  
    | ns16k | ns32k \  
    | or32 \  
    |
```

# Modify config.sub



- ❖ Find the line below "Recognize the basic CPU types with company name."

```
@@ -351,6 +352,7 @@  
    | mmix-* \  
    | mt-* \  
    | msp430-* \  
+   | nds32-* | nds32le-* \  
    | nios-* | nios2-* \  
    | none-* | np1-* | ns16k-* | ns32k-* \  
    |
```

# Configure



## ❖ Using build scripts

```
# Andes toolchain setting
export ANDESIGHT_ROOT=/home/path/toolchains/nds32-elf-n1213-43u1h
export PATH=$ANDESIGHT_ROOT/bin:$PATH

# PREFIX folder
export PREFIX=/my/path/nds32
#export CROSS_COMPILE="nds32le-linux-" #新版
export CROSS_COMPILE="nds32-elf-"
which ${CROSS_COMPILE}gcc &> /dev/null || export CROSS_COMPILE="nds32-elf"
export HOST=${CROSS_COMPILE%-}
export BUILD="i686-linux"
export TARGET=${CROSS_COMPILE%-}
```



# Configure



## ❖ Assign Andes toolchains

```
# Cross Toolchain
export BUILD_CC=gcc
export CC="${TARGET}-gcc"
export CXX="${TARGET}-g++"
export AR="${TARGET}-ar"
export AS="${TARGET}-as"
export RANLIB="${TARGET}-ranlib"
export LD="${TARGET}-ld"
export STRIP="${TARGET}-strip"
```

# Deploy



- ❖ Copy the folder of your building path to SD card
- ❖ Set environment variable of library

# Make and Install



- ❖ The compile time error can find in this step

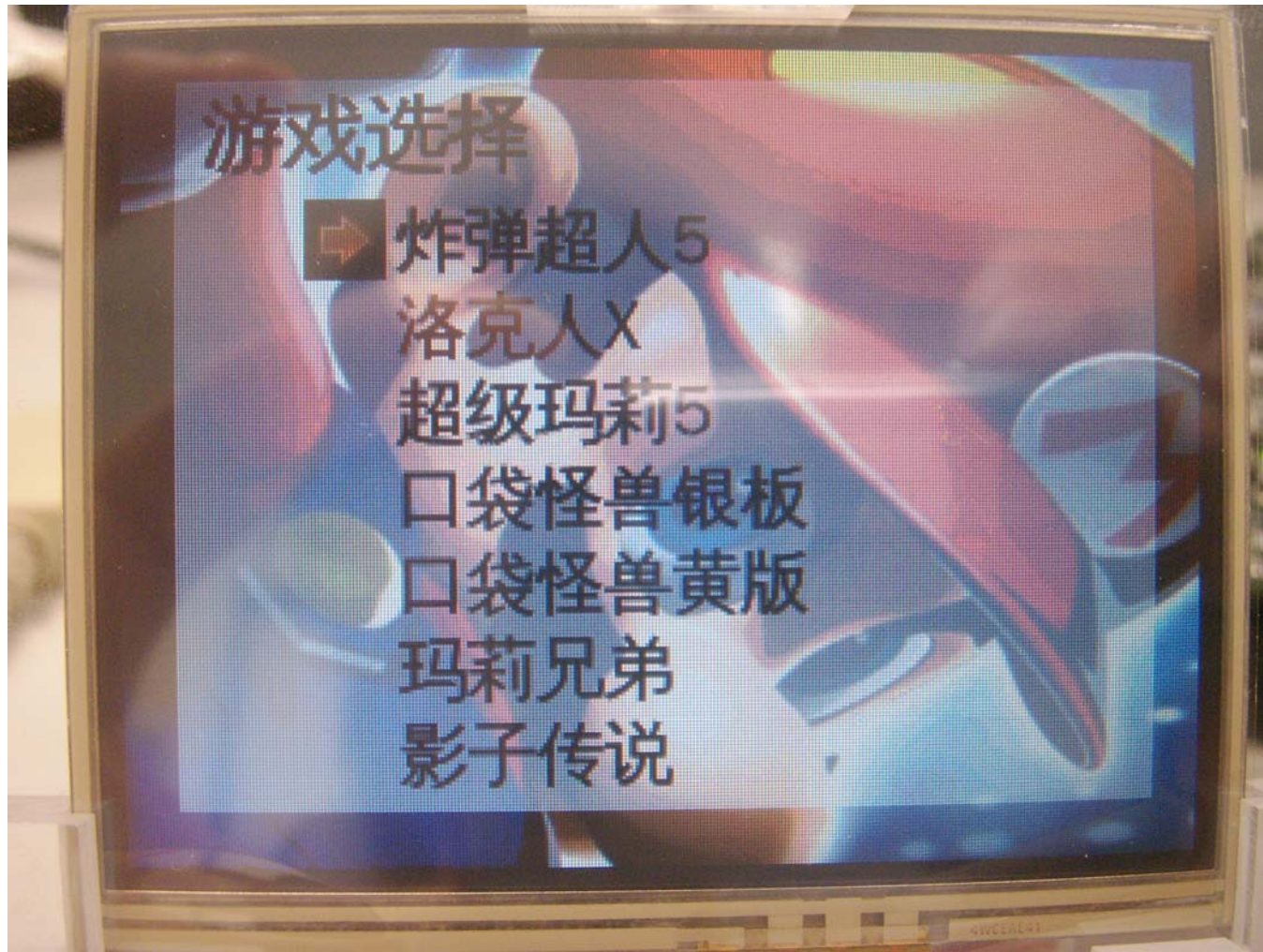
```
./configure --host=$HOST --build=$BUILD --prefix=$PREFIX  
make  
#make install DESTDIR=${DEPLOY_DIR}  
make install DESTDIR=${PREFIX}
```

# Game platform



- ❖ Demo how to play games on Andes platform
- ❖ Emulate a hardware architecture of a game system
- ❖ A game emulator will be composed of the following modules
  - A CPU emulator or CPU simulator (the two terms are mostly interchangeable in this case)
  - A memory subsystem module
  - Various I/O devices emulators

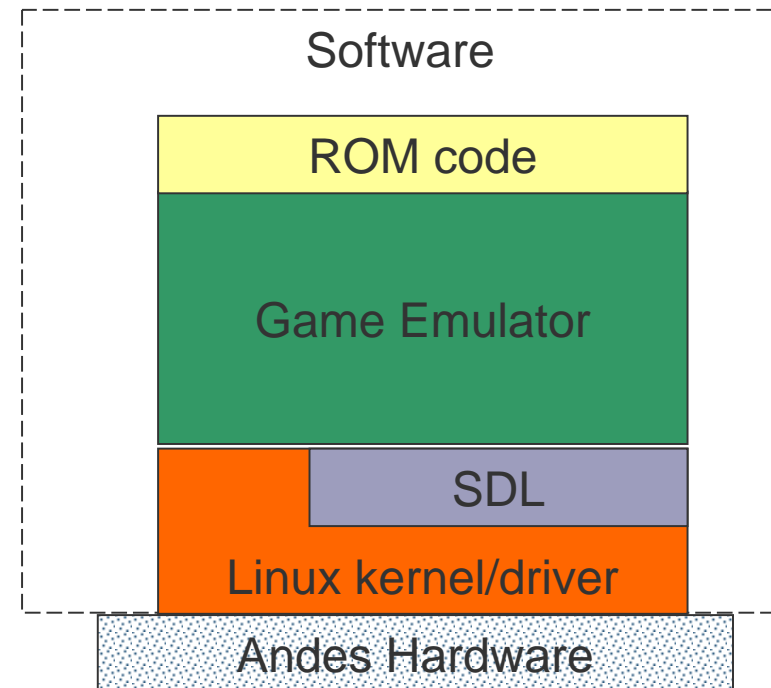
# Game Menu



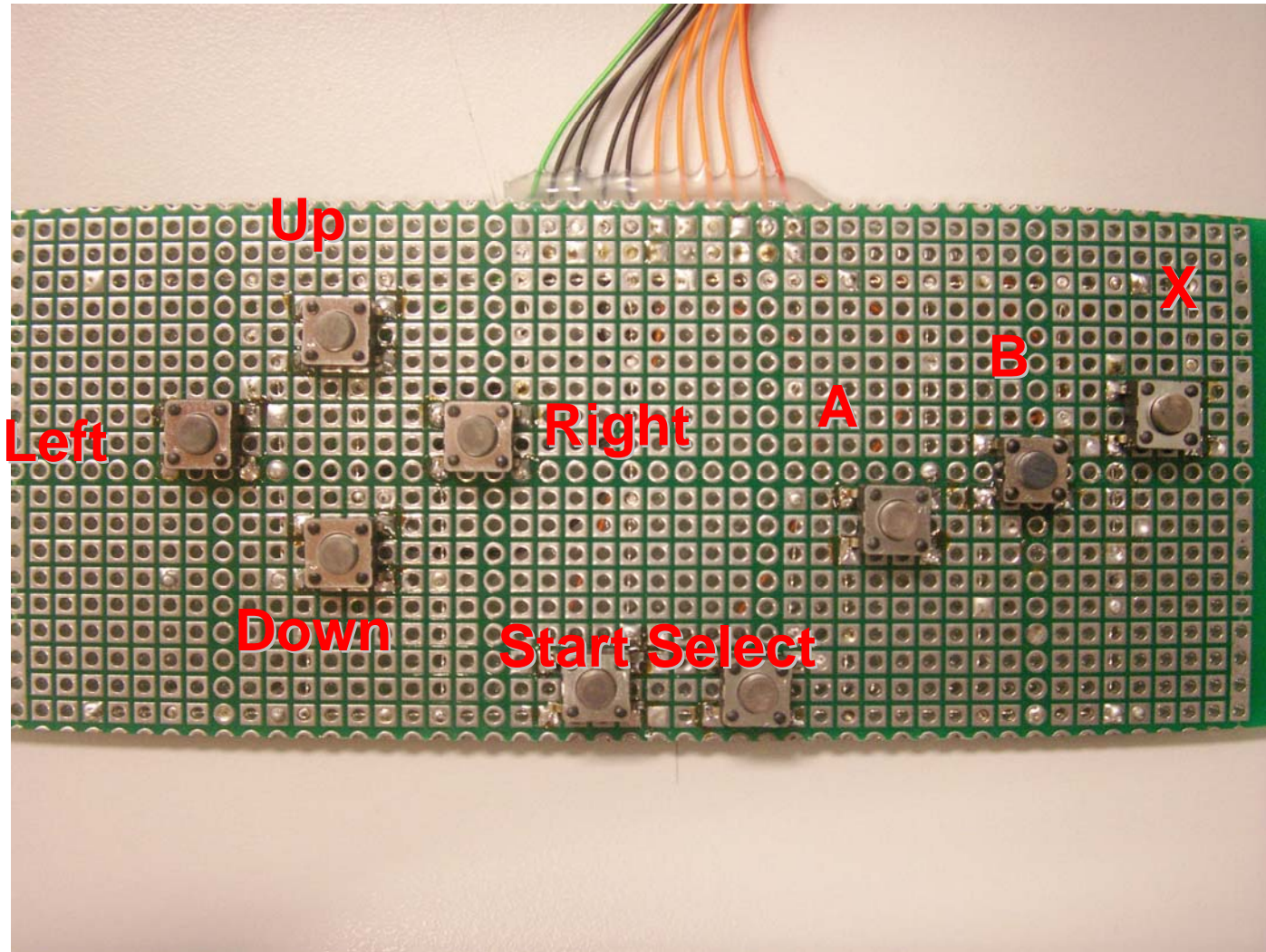
# System Architecture



- ❖ ROM code
- ❖ Game Emulator
  - Game Boy
  - Gameboy Advance
  - Nintendo Entertainment System
  - Super Nintendo Entertainment System
- ❖ Simple DirectMedia Layer (SDL)
  - Display on frame buffer
  - Audio
- ❖ Linux kernel/drivers
  - Audio driver
  - Joy Stick/GPIO
  - SD card driver



# Joy Stick



# Package dependency



- ❖ InfoNES
- ❖ VisualBoyAdvance 1.7.2
  - SDL 1.2.14
  - libpng 1.2.40
  - zlib 1.2.3
- ❖ Snes9x
  - SDL 1.2.14
- ❖ Gnuboy 1.0.3
  - SDL 1.2.14



# Gnuboy Performance Issue



- ❖ No floating point code whatsoever.
- ❖ Use fixed point or better yet exact analytical integer methods as opposed to any approximation.
- ❖ Assembly code implementation of CPU emulation

# Game Emulator



- ❖ Game Boy
- ❖ Nintendo Entertainment System
- ❖ Super Nintendo Entertainment System

# Game Boy(1/4)

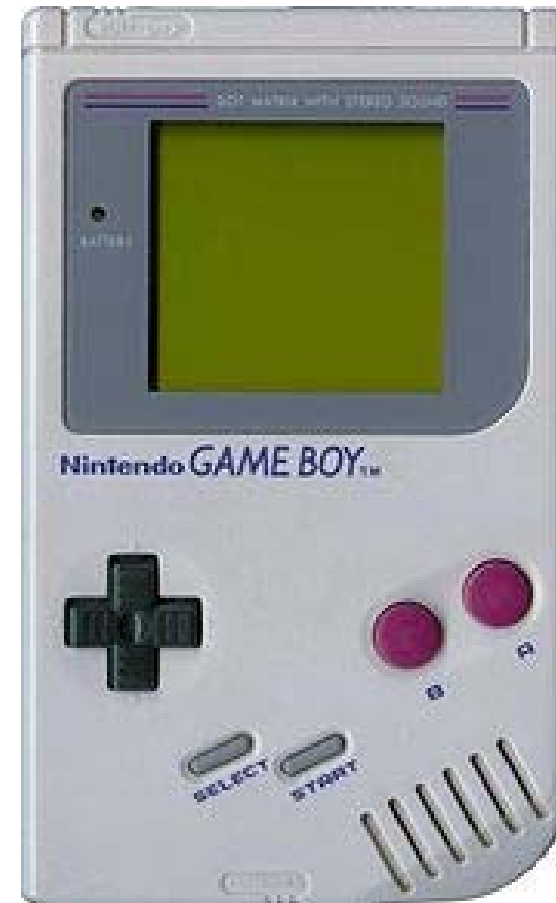


- ❖ An 8-bit handheld video game console developed and manufactured by Nintendo
- ❖ It was released in Japan on April 21, 1989 (1989-04-21)
- ❖ It was created by Gunpei Yokoi and Nintendo's Research and Development 1
  - The same staff who had designed the Game & Watch series as well as several popular games for the NES.

# Game Boy(2/4)



- ❖ Directional pad
- ❖ Four operation buttons
  - "A"
  - "B"
  - "SELECT"
  - "START"



# Game Boy(3/4)



## ❖ CPU

- Custom 8-bit Sharp LR35902 core at 4.19 MHz
- The core also contains integrated sound generation

## ❖ RAM

- 8 kB internal S-RAM

## ❖ Video RAM

- 8 kB internal

## ❖ ROM

- On-CPU-Die 256-byte bootstrap; 256 kb, 512 kb, 1 Mb, 2 Mb, 4 Mb and 8 Mb cartridges

# Gnuboy Performance Issue



- ❖ No floating point code whatsoever.
- ❖ Use fixed point or better yet exact analytical integer methods as opposed to any approximation.
- ❖ Assembly code implementation of CPU emulation

# Game Boy(4/4)



## ❖ Sound

- 2 square waves, 1 programmable 32-sample 4-bit PCM wave, 1 white noise, and one audio input from the cartridge

## ❖ Display

- Reflective LCD 160 × 144 pixels

## ❖ Screen size

- 66 mm (2.6 in) diagonal

## ❖ Power

- 6 V, 0.7 W (4 AA batteries provide ~14~35 hours)

## ❖ Dimensions: 90 mm (W) x 148 mm (H) x 32 mm (D) / 3.5" x 5.8" 1.3" (in)

# Nintendo Entertainment System



## ❖ CPU

- Ricoh 2A03 8-bit processor (MOS Technology 6502 core)

## ❖ Controller input

- 2 controller port

## ❖ Best-selling game

- Super Mario Bros.

## ❖ Predecessor

- Color TV Game

## ❖ Successor

- Super Nintendo Entertainment System





# Nintendo Entertainment System



## ❖ Game

- Super Mario Bros.
- The Legend of Kage



# Super Nintendo Entertainment System



- ❖ Built on 16-bit architectures and offered improved graphics and sound over the 8-bit NES



# Super Nintendo Entertainment System



## ❖ CPU reference

- Clock Rates (NTSC)
  - Input: 21.47727 MHz
- Bus
  - 3.58 MHz, 2.68 MHz, or 1.79 MHz
- Clock Rates (PAL)
  - Input: 21.28137 MHz
- Bus
  - 3.55 MHz, 2.66 MHz, or 1.77 MHz
  - 24-bit and 8-bit address buses, 8-bit data bus

# Super Nintendo Entertainment System



## ❖ Additional Features

- DMA and HDMA
- Timed IRQ
- Parallel I/O processing
- Hardware multiplication and division

# Super Nintendo Entertainment System



## ❖ CPU

- 16-bit 65c816 Ricoh 5A22 3.58 MHz

## ❖ Best-selling game

- Super Mario World
- Donkey Kong Country

## ❖ Predecessor

- Nintendo Entertainment System

## ❖ Successor

- Nintendo 64

# Super Nintendo Entertainment System



## ❖ Game

- Super Mario World
- Mega Man X
- Super Bomberman 5



# Five Inch Version



# Resource usage and profiling



- ❖ VisualBoyAdvance on Intel(R) Core(TM)2 Quad CPU Q6600 @ 2.40GHz
- ❖  $192/100 \Rightarrow 1.92$  Core



系統監控

監控(M) 編輯(E) 檢視(V) 說明(H)

系統 程序 資源 檔案系統

最後 1、5、15 分鐘的平均負載 : 3.12, 1.80, 0.83

程序名稱	狀態	% CPU ^	記憶體	虛擬記憶體	Nice	等待頻道
VisualBoyAdvance	執行中	192	34.2 MiB	142.0 MiB	0	0
screenshot	睡眠中	0	2.8 MiB	35.7 MiB	0	poll_sched
anome-system-monitor	執行中	8	4.9 MiB	56.1 MiB	0	0



# The result of playing game on Andes platform



Emulator name	CPU consumption rate	DRAM consumption rate	smoothness 0~100
Game boy	30~5%	4%	90
Game boy Advance	98%	32%	20
NES	98%	3%	55
SNES	98%	16%	90

# The ways of enhancement performance



- ❖ Game emulator study and optimization
- ❖ Eliminate floating point calculation
- ❖ Close hardware emulation

# Andes Platform Porting Experiment

Andes Technology



Architecture for **N**ext-generation  
Digital **E**ngines for **S**oC

# Outline



- ❖ Porting Diff package
  - Creating working platform
  - Getting package
  - Adding Andes architecture
  - Configure
  - Make
  - Deploy

# About Diff



- ❖ Diff is a file comparison utility
- ❖ Outputs the differences between two files.
- ❖ Show the changes between a file and a former version of the same file.
- ❖ The output is called a diff or a patch
- ❖ Like the use of the word "grep" for describing the act of searching, the word diff is used in jargon as a verb for calculating any difference.

# Getting started



- ❖ Make sure you have
  - Linux platform
  - Tool-chains
  - A folder name nds32 under your home folder

# Getting package



## ❖ apt-get source diff

```
檔案(E) 編輯(E) 檢視(V) 終端機(T) 求助(H)
liu@ysliu-desktop:~/tmp/diff$ apt-get source diff
正在讀取套件清單... 完成
正在重建相依關係
正在讀取狀態資料... 完成
需要下載 692kB 的原始套件檔。
下載 :1 http://debian.nctu.edu.tw karmic/main diffutils 2.8.1-13 (dsc) [920B]
下載 :2 http://debian.nctu.edu.tw karmic/main diffutils 2.8.1-13 (tar) [666kB]
下載 :3 http://debian.nctu.edu.tw karmic/main diffutils 2.8.1-13 (diff) [25.0kB]
取得 692kB 用了 0s (1,151kB/s)
gpgv: 由 2009年05月30日 (週六) 00時03分39秒 CST 建立的簽章, 使用 DSA 金鑰 ID 3EF23CD6
gpgv: 無法檢查簽章: 找不到公鑰
dpkg-source: warning: failed to verify signature on ./diffutils_2.8.1-13.dsc
dpkg-source: info: extracting diffutils in diffutils-2.8.1
dpkg-source: info: unpacking diffutils_2.8.1.orig.tar.gz
dpkg-source: info: applying diffutils_2.8.1-13.diff.gz
liu@ysliu-desktop:~/tmp/diff$
```







# Using Scripts



## ❖ ./build\_nds.sh

```
檔案(E) 編輯(E) 檢視(V) 终端機(T) 求助(H)
liu@ysliu-desktop:~/tmp/diff/diffutils-2.8.1$ ./build_nds.sh
/home/liu/Andestech/AndeSight133//toolchains/nds32le-linux-glibc-V0/bin/nds32le-
checking for a BSD-compatible install... /usr/bin/install -c
checking whether build environment is sane... yes
checking for gawk... no
checking for mawk... mawk
checking whether make sets ${MAKE}... yes
checking for nds32le-linux-strip... nds32le-linux-strip
checking for gawk... (cached) mawk
checking for nds32le-linux-gcc... nds32le-linux-gcc -I/home/liu/nds32/include
checking for C compiler default output... a.out
checking whether the C compiler works... yes
checking whether we are cross compiling... yes
checking for suffix of executables...
checking for suffix of object files... o
checking whether we are using the GNU C compiler... yes
checking whether nds32le-linux-gcc -I/home/liu/nds32/include accepts -g... yes
```

# If You Forget to Add Andes Arch.



檔案(E) 編輯(E) 檢視(V) 終端機(T) 求助(H)

```
checking for stdlib.h... (cached) yes
checking for string.h... (cached) yes
checking sys/file.h usability... yes
checking sys/file.h presence... yes
checking for sys/file.h... yes
checking time.h usability... yes
checking time.h presence... yes
checking for time.h... yes
checking for unistd.h... (cached) yes
checking for struct stat.st_blksize... yes
checking for struct stat.st_rdev... yes
checking for ptrdiff_t... yes
checking for ssize_t... yes
checking for unsigned long long... yes
checking for uintmax_t... yes
checking for msgfmt... no
checking for gmsgfmt... :
checking for xgettext... no
checking for msgmerge... no
checking build system type... Invalid configuration `nds32-linux': machine `nds32' not recognized
configure: error: /bin/bash config/config.sub nds32-linux failed
make: *** No targets specified and no makefile found. Stop.
make: *** No rule to make target `install'. Stop.
liu@ysliu-desktop:~/tmp/diff/diffutils-2.8.1$
```

# Configure



檔案(E) 編輯(E) 檢視(V) 終端機(T) 求助(H)

```
checking for doprnt... no
checking whether strerror is declared... yes
checking whether strerror_r is declared... yes
checking for strerror_r... yes
checking whether strerror_r returns char *... yes
checking for ANSI C header files... (cached) yes
checking for function prototypes... yes
checking for setlocale... yes
checking for locale.h... (cached) yes
checking for stdlib.h... (cached) yes
checking for string.h... (cached) yes
checking for isascii... yes
checking for iswprint... yes
checking whether mbrtowc and mbstate_t are properly declared... yes
checking for limits.h... (cached) yes
checking stddef.h usability... yes
checking stddef.h presence... yes
checking for stddef.h... yes
checking for stdlib.h... (cached) yes
checking for string.h... (cached) yes
checking wchar.h usability... yes
checking wchar.h presence... yes
checking for wchar.h... yes
checking wctype.h usability... yes
```

# Compile



```
檔案(E) 編輯(E) 檢視(V) 終端機(T) 求助(H)
    depfile='.deps/util.Po' tmpdepfile='.deps/util.TPo' \
    depmode=gcc3 /bin/bash ../config/depcomp \
    nds32le-linux-gcc -I/home/liu/nds32/include -DLOCALEDIR=\"/home/liu/nds3
-fomit-frame-pointer -funroll-loops -O3 -c `test -f util.c || echo './`util.c
source='diff3.c' object='diff3.o' libtool=no \
    depfile='.deps/diff3.Po' tmpdepfile='.deps/diff3.TPo' \
    depmode=gcc3 /bin/bash ../config/depcomp \
    nds32le-linux-gcc -I/home/liu/nds32/include -DLOCALEDIR=\"/home/liu/nds3
-fomit-frame-pointer -funroll-loops -O3 -c `test -f diff3.c || echo './`diff3.c
source='sdiff.c' object='sdiff.o' libtool=no \
    depfile='.deps/sdiff.Po' tmpdepfile='.deps/sdiff.TPo' \
    depmode=gcc3 /bin/bash ../config/depcomp \
    nds32le-linux-gcc -I/home/liu/nds32/include -DLOCALEDIR=\"/home/liu/nds3
-fomit-frame-pointer -funroll-loops -O3 -c `test -f sdiff.c || echo './`sdiff.c
In file included from ../lib/regex.h:2,
    from util.c:27:
../lib/posix/regex.h:532:1: warning: "__restrict_arr" redefined
In file included from /home/liu/Andestech/AndeSight133/toolchains/nds32le-linux-
    from /home/liu/Andestech/AndeSight133/toolchains/nds32le-linux-
    from system.h:61,
    from diff.h:23,
    from util.c:23:
/home/liu/Andestech/AndeSight133/toolchains/nds32le-linux-glibc-V0/libexec/./nd
initon
```

# Deploy



- ❖ Copy all the files and folder under `$HOME/nds32` folder
- ❖ Diff usage
  - `diff [diff opts] file1 file2`

# Thank You!!!



Architecture for **N**ext-generation  
Digital **E**ngines for **S**oC