Have Fun on Andes Platform — Game Emulator an overview

劉昱賢 thomasysliu at COSCUP 2010

Outline

- Who am I
- Introduction
- What is Andes Platform
- System Architecture
- Game package details
- Porting how-to
- Performance issue
- Have fun
- Q&A

Who am I

- 交通大學電子工程系學生
- 對嵌入式系統領域稍有涉獵
- Thomas 之前於晶心科技，一間台灣 32 位元處理器的 IC 設計公司打工
- 在台灣心的開發平台，都靠 Game Emulator 騙吃騙喝
- 熱於探索應用自由軟體的應用模式與機會。

Introduction

- 生活不忘娛樂，來點輕鬆的小遊戲增添一些歡樂吧！
- 使用 Andes Core™ N1213 的 platform SoC 所建構的發展系統硬體主板
- 搭配 3.5 吋 320×240 畫質 Panel，展示了此平台開發掌上型遊戲機，遊戲軟體與 Andes Core 的完美結合
- 跟大家分享一些在 Embedded Linux 上執行 Emulator 的使用心得

Introduction

- 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

What is Andes Platform

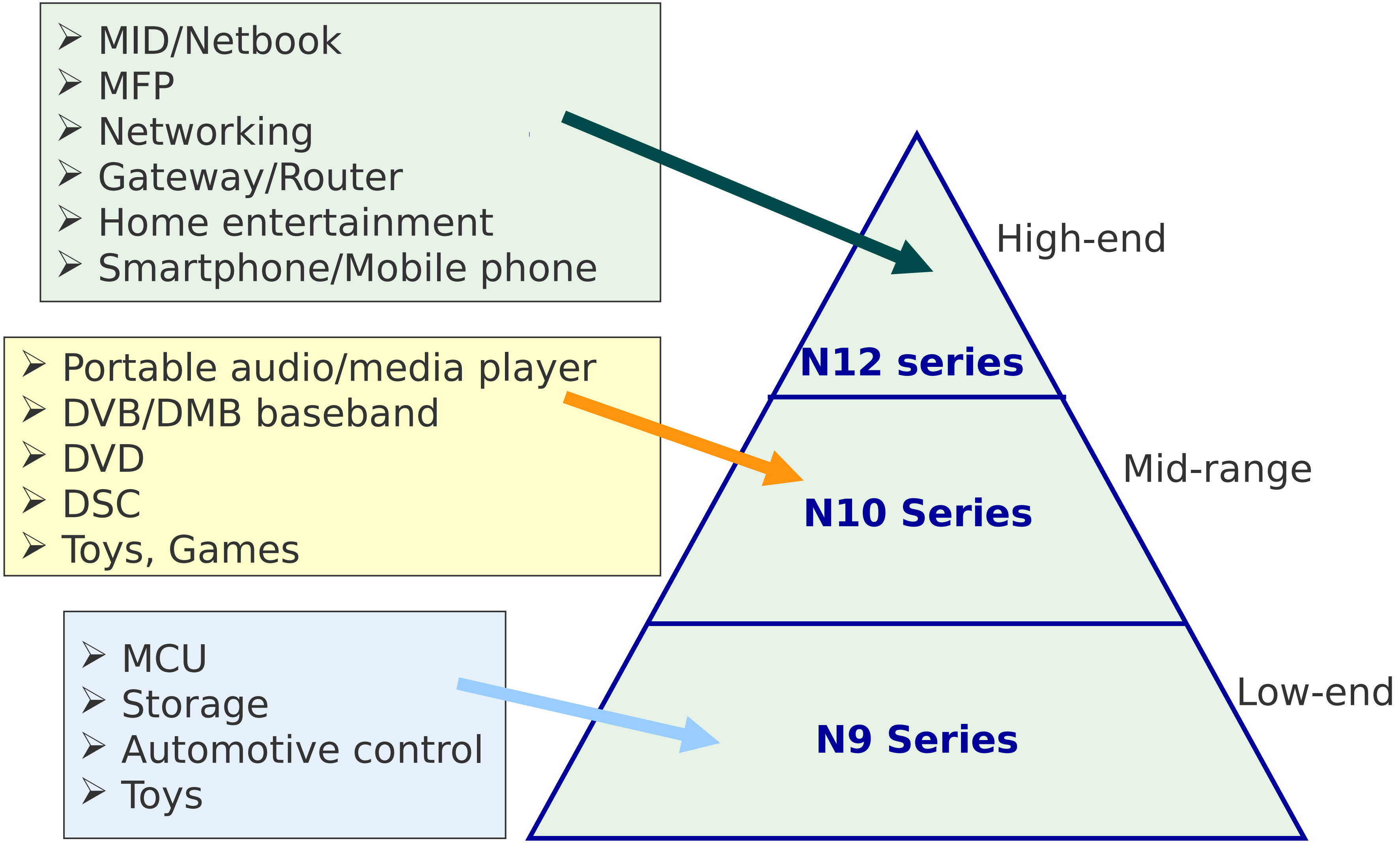
- 晶心科技 (Andestech)
- Andes Technology Corporation was founded in the Hsinchu Science Based Industrial Park（ SiSoft Research Center ）in the first half of 2005.
- Andestech devote in developing high-performance/low-power 32-bit processors and its associated SoC platforms to serve the rapidly growing embedded system applications worldwide.

What is Andes Platform

What is Andes Platform

- NDS32 is a new 32-bit RISC architecture invented by Andestech.com .
- It has a 16-bit/32-bit mixed-length instruction set to achieve optimal
- system performance, code density, and power efficiency.
- NDS32 also provide N9, N10, N12 different CPU core families for soft-core and hard-core SoC design.

AndeSoft™ Market Segments



- MID/Netbook
- MFP
- Networking
- Gateway/Router
- Home entertainment
- Smartphone/Mobile phone

- Portable audio/media player
- DVB/DMB baseband
- DVD
- DSC
- Toys, Games

- MCU
- Storage
- Automotive control
- Toys

N12 series

High-end

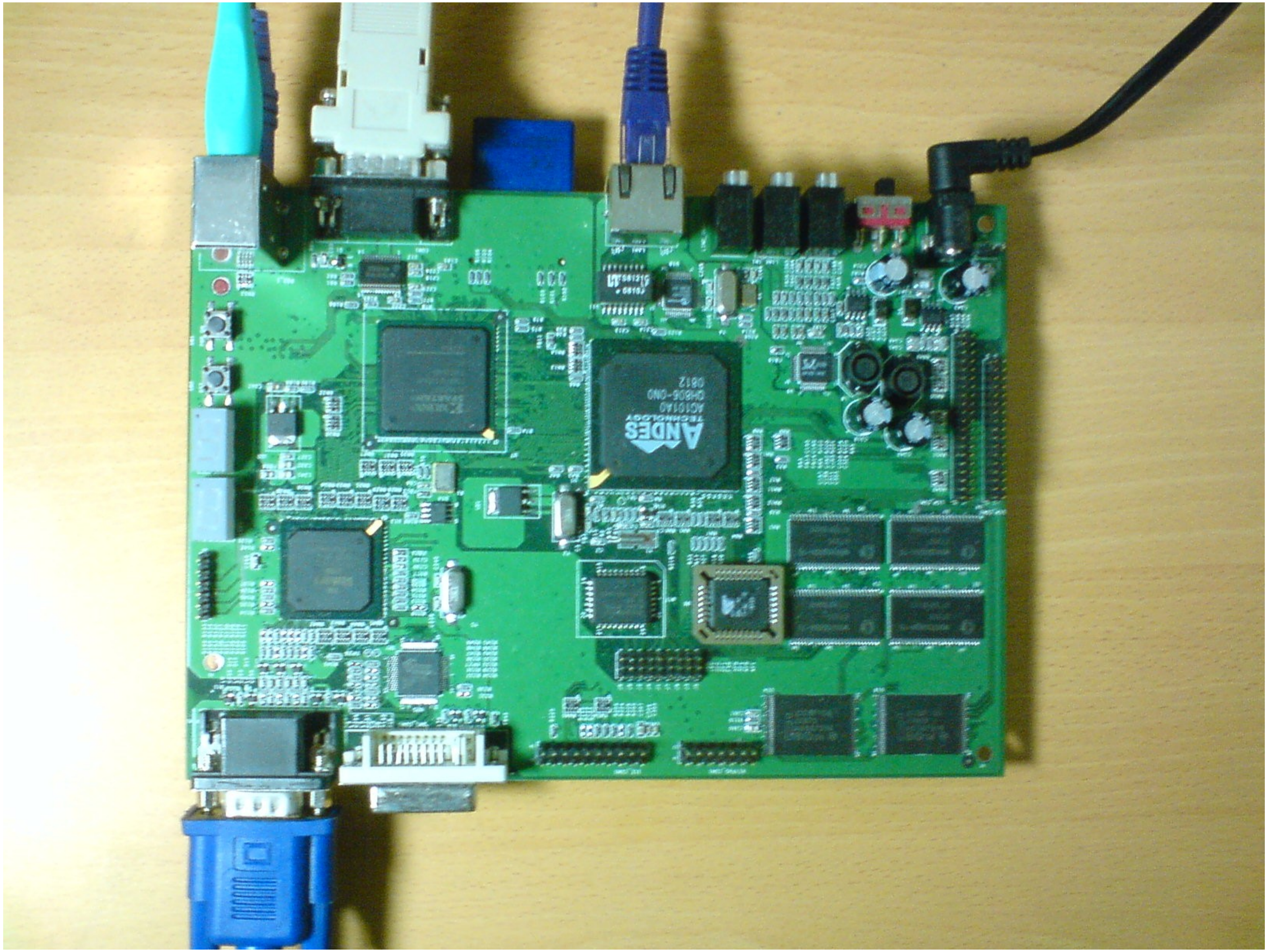
N10 Series

Mid-range

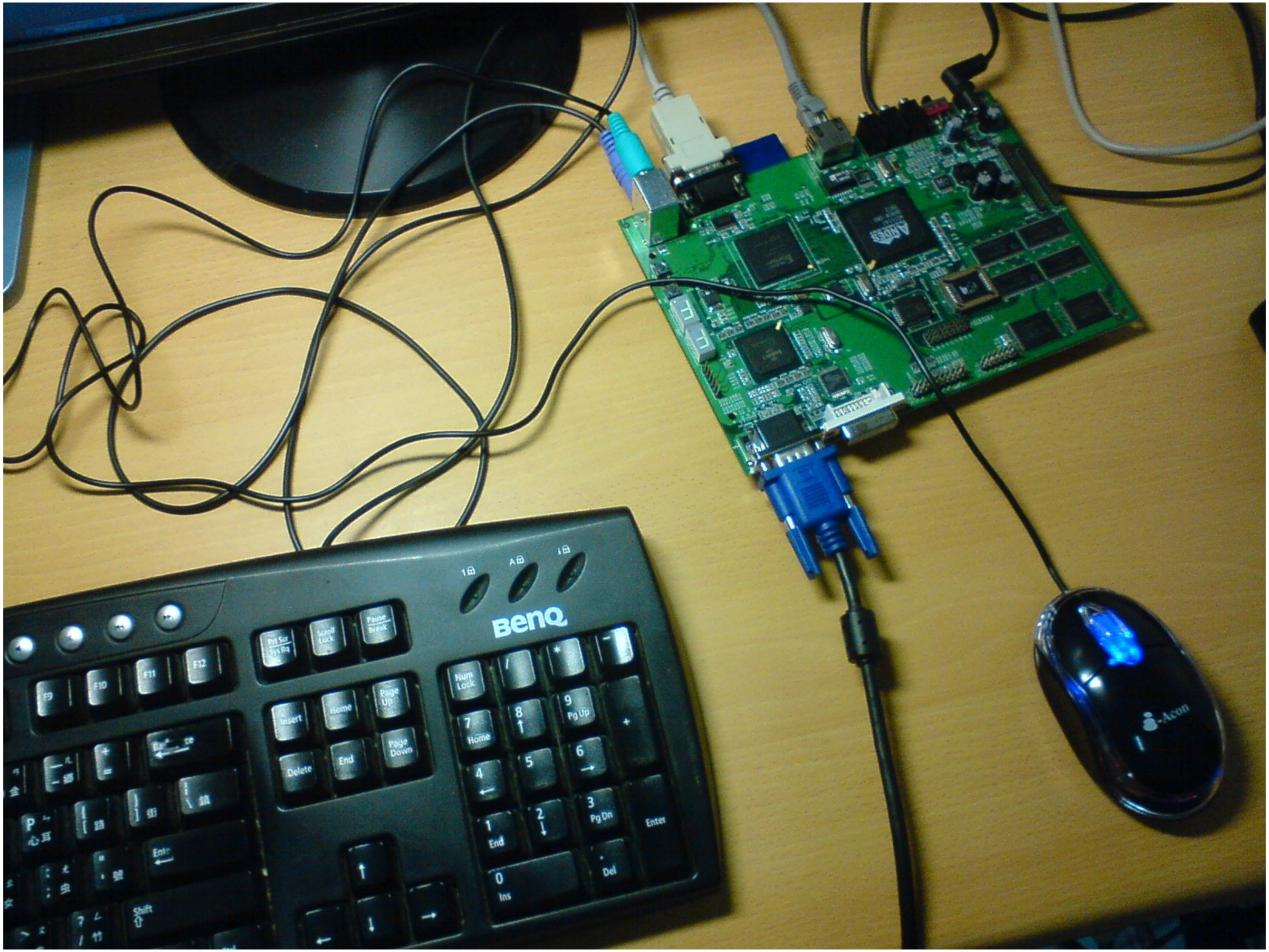
N9 Series

Low-end

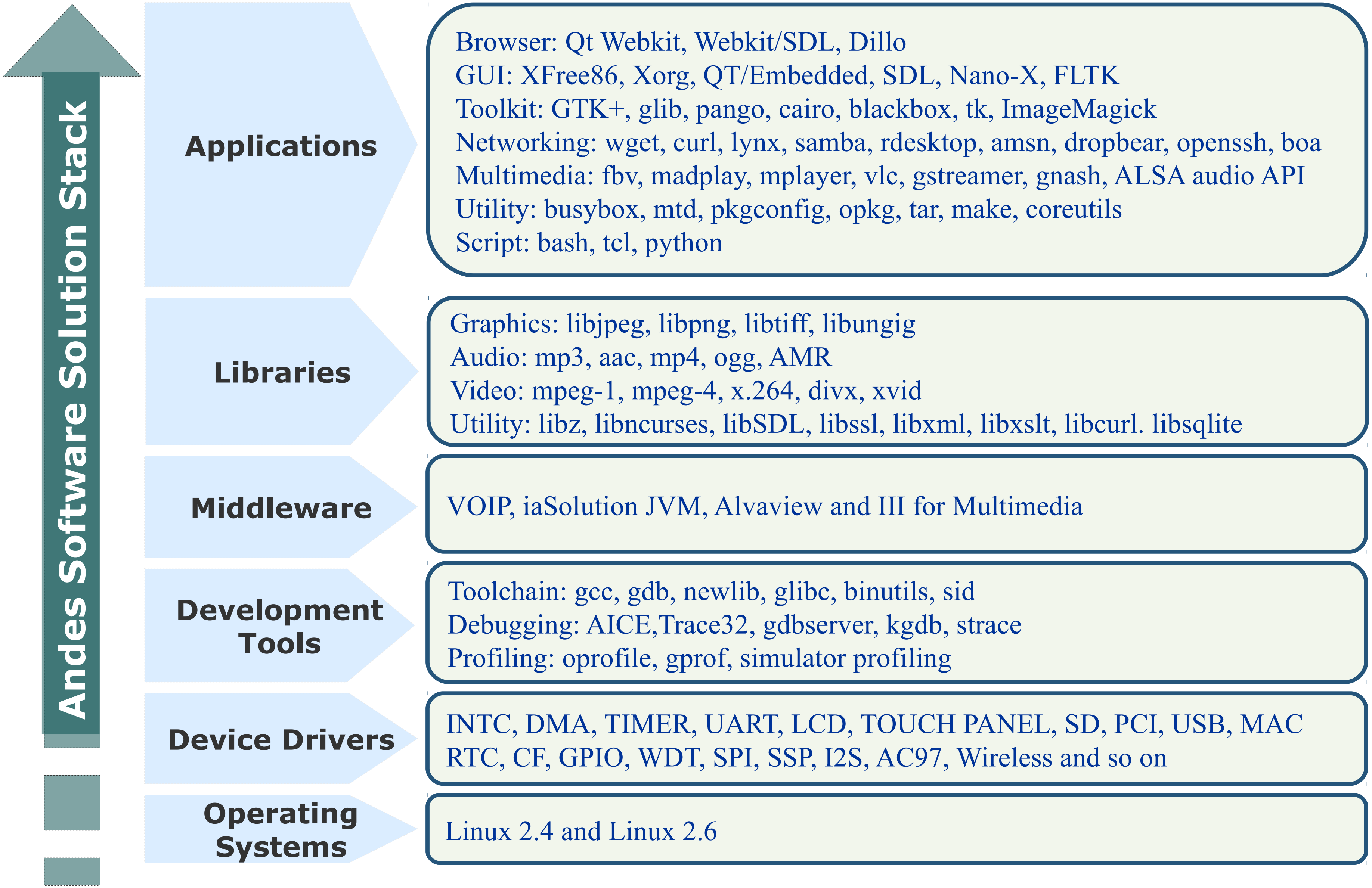
What is Andes Platform



What is Andes Platform



AndeSoft™: Linux Solutions



Applications

Browser: Qt Webkit, Webkit/SDL, Dillo
GUI: XFree86, Xorg, QT/Embedded, SDL, Nano-X, FLTK
Toolkit: GTK+, glib, pango, cairo, blackbox, tk, ImageMagick
Networking: wget, curl, lynx, samba, rdesktop, amsn, dropbear, openssh, boa
Multimedia: fbv, madplay, mplayer, vlc, gstreamer, gnash, ALSA audio API
Utility: busybox, mtd, pkgconfig, opkg, tar, make, coreutils
Script: bash, tcl, python

Libraries

Graphics: libjpeg, libpng, libtiff, libungig
Audio: mp3, aac, mp4, ogg, AMR
Video: mpeg-1, mpeg-4, x.264, divx, xvid
Utility: libz, libncurses, libSDL, libssl, libxml, libxslt, libcurl. libsqlite

Middleware

VOIP, iaSolution JVM, Alvaview and III for Multimedia

Development Tools

Toolchain: gcc, gdb, newlib, glibc, binutils, sid
Debugging: AICE,Trace32, gdbserver, kgdb, strace
Profiling: oprofile, gprof, simulator profiling

Device Drivers

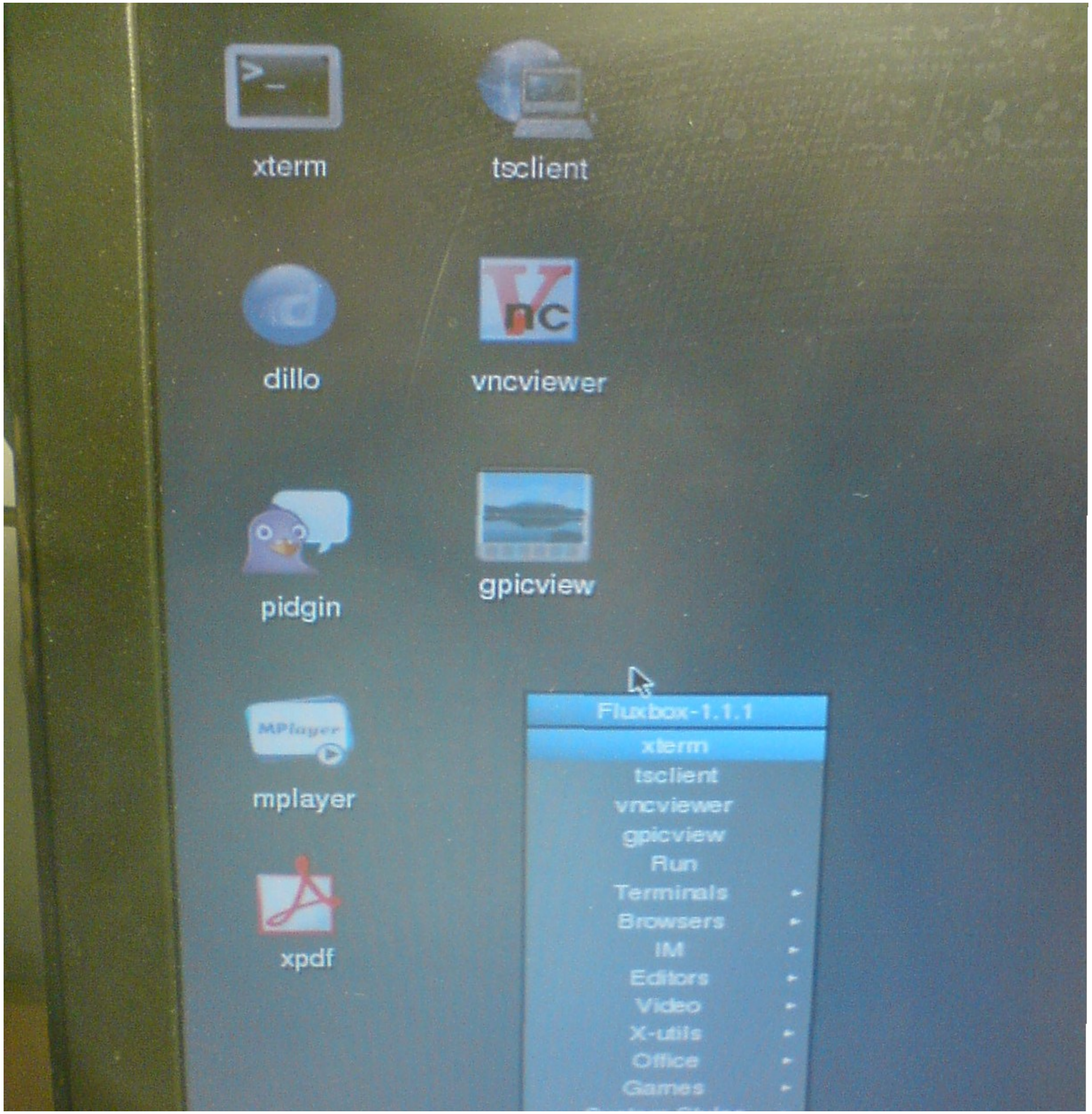
INTC, DMA, TIMER, UART, LCD, TOUCH PANEL, SD, PCI, USB, MAC
RTC, CF, GPIO, WDT, SPI, SSP, I2S, AC97, Wireless and so on

Operating Systems

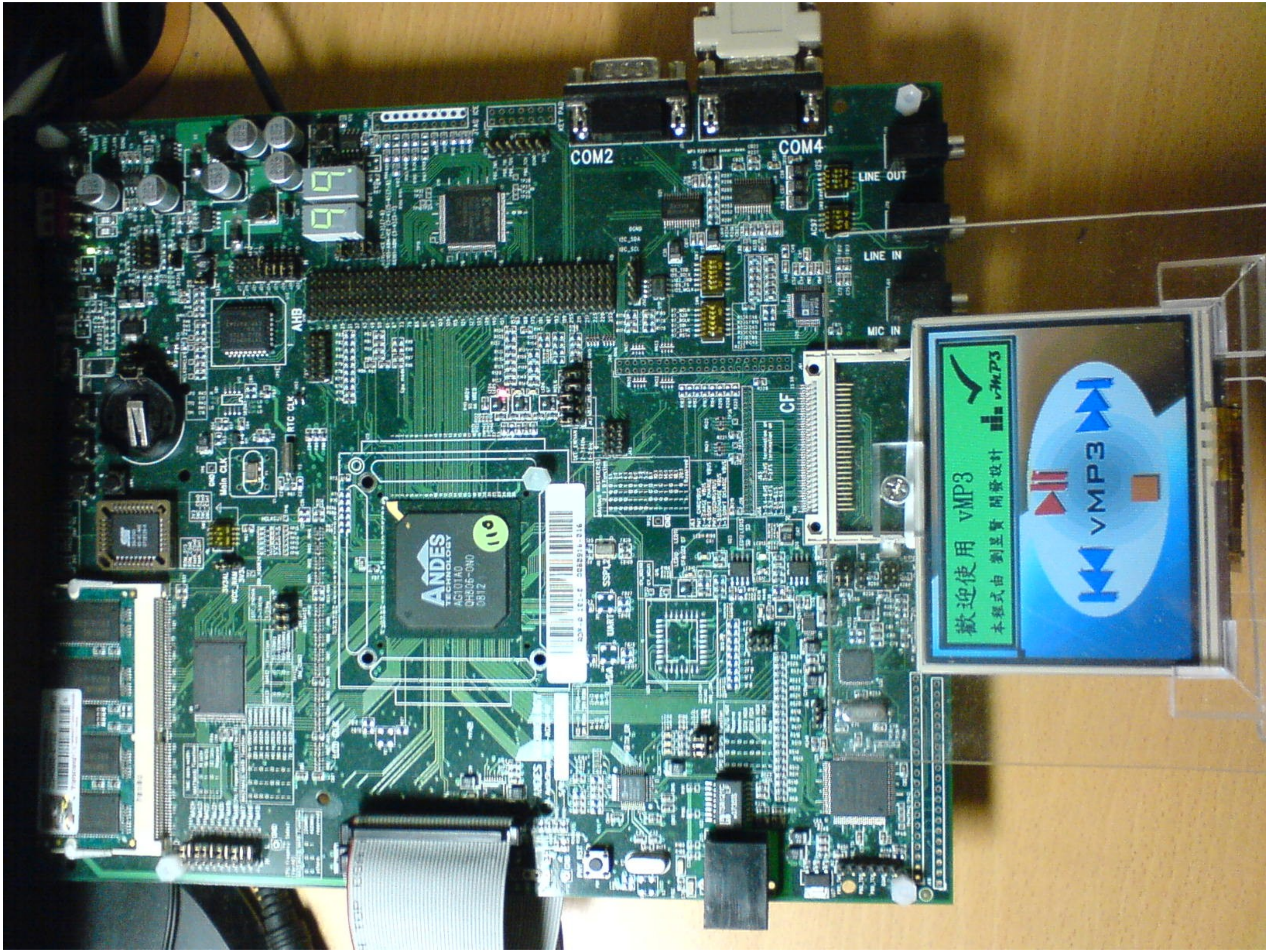
Linux 2.4 and Linux 2.6

What is Andes Platform

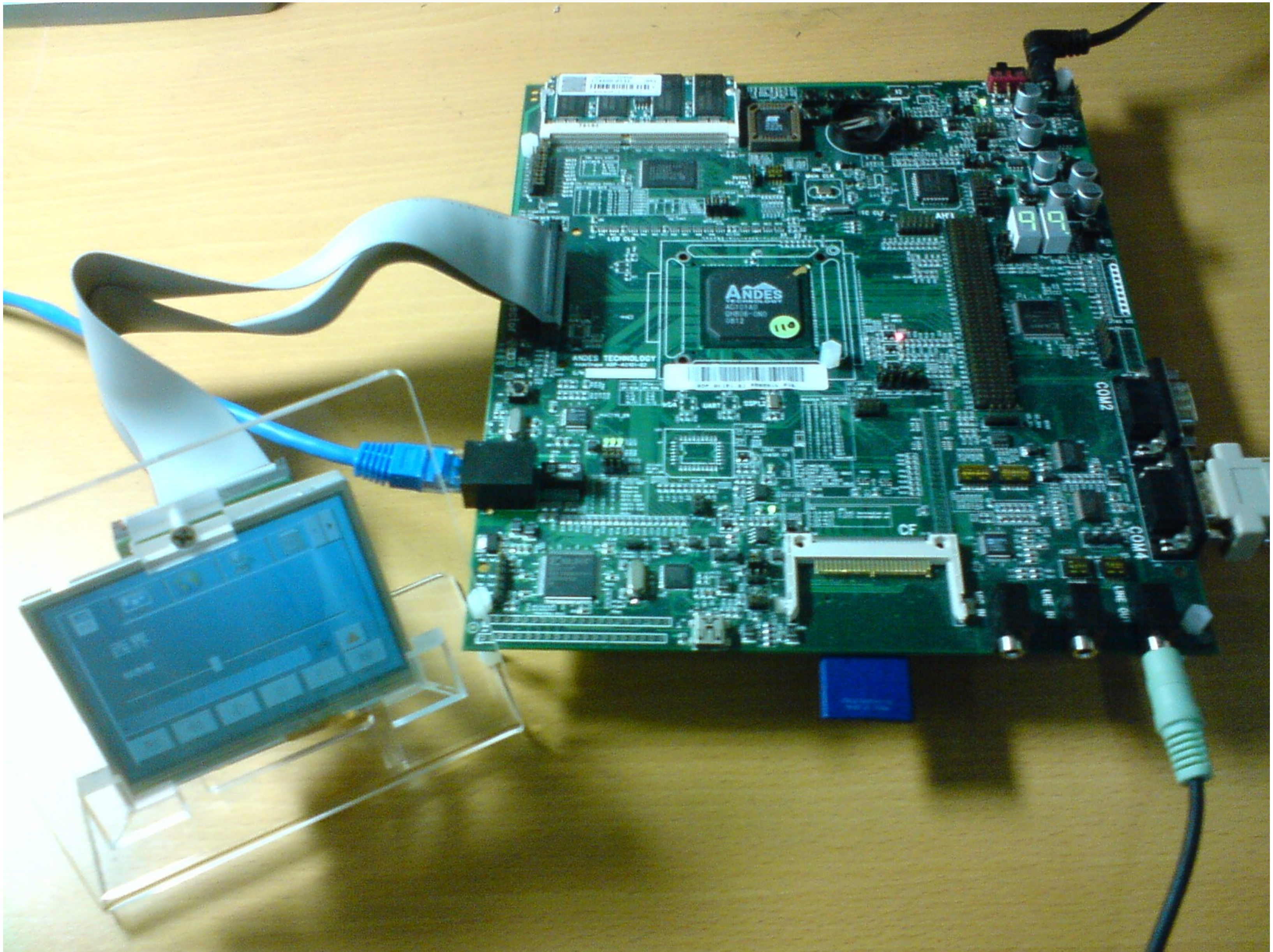
- Xwindow
- gtk
- Fluxbox
- 上 MSN
- 看影片
- 看 pdf



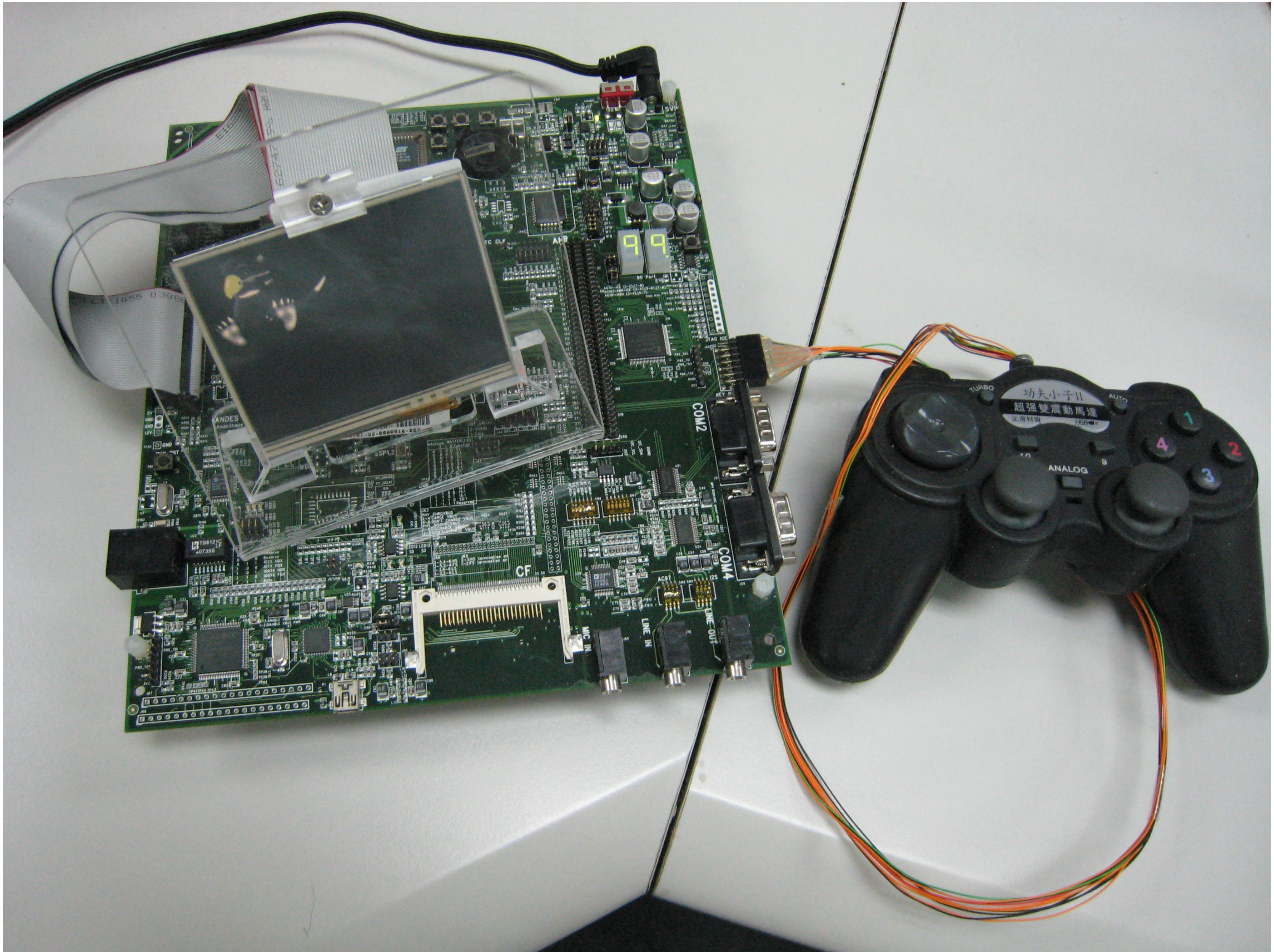
What is Andes Platform



What is Andes Platform

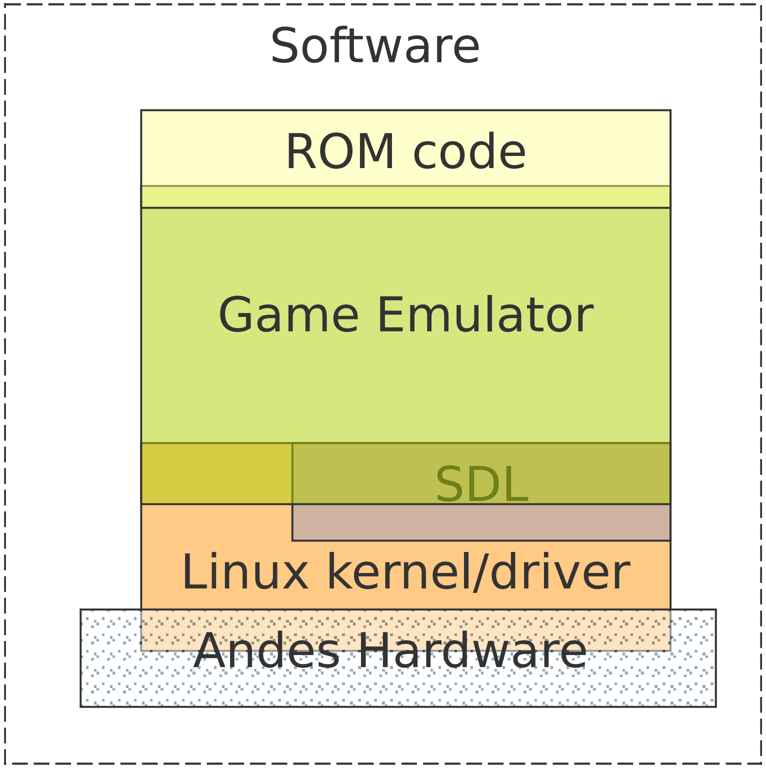


Andes Game Platform



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



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

Game Emulator

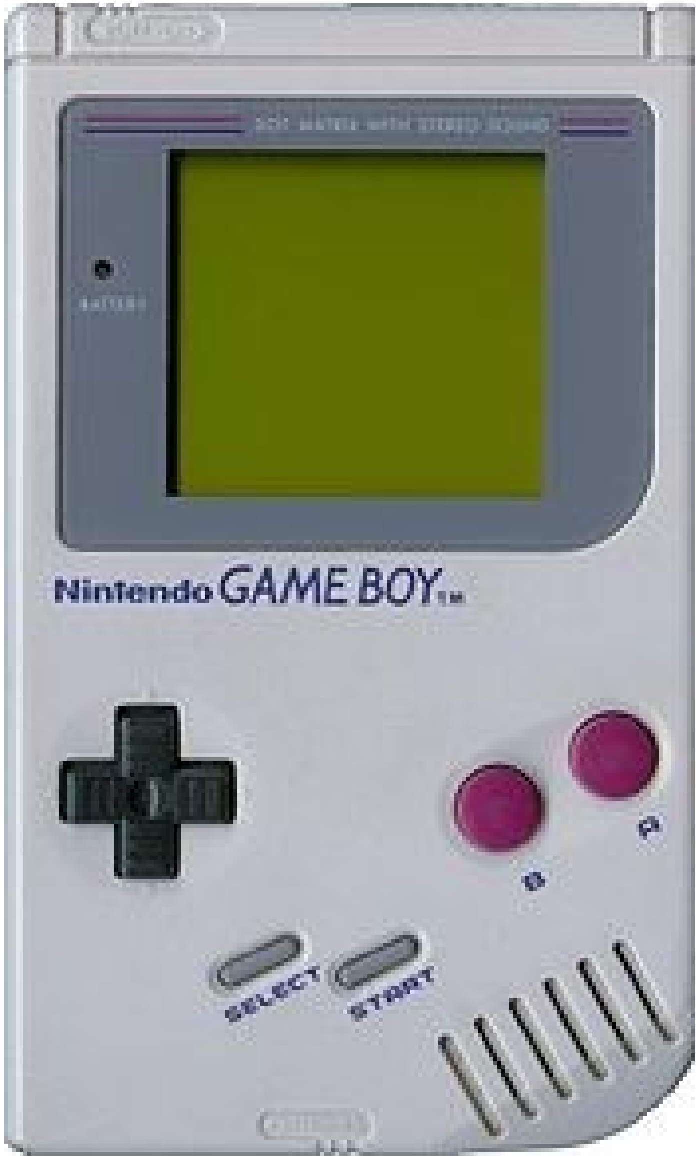
- Game Boy
- Nintendo Entertainment System
- Super Nintendo Entertainment System
- Game Boy Advance

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"



Picture from wikipedia

http://en.wikipedia.org/wiki/File:Gameboy.jpg

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.
- 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

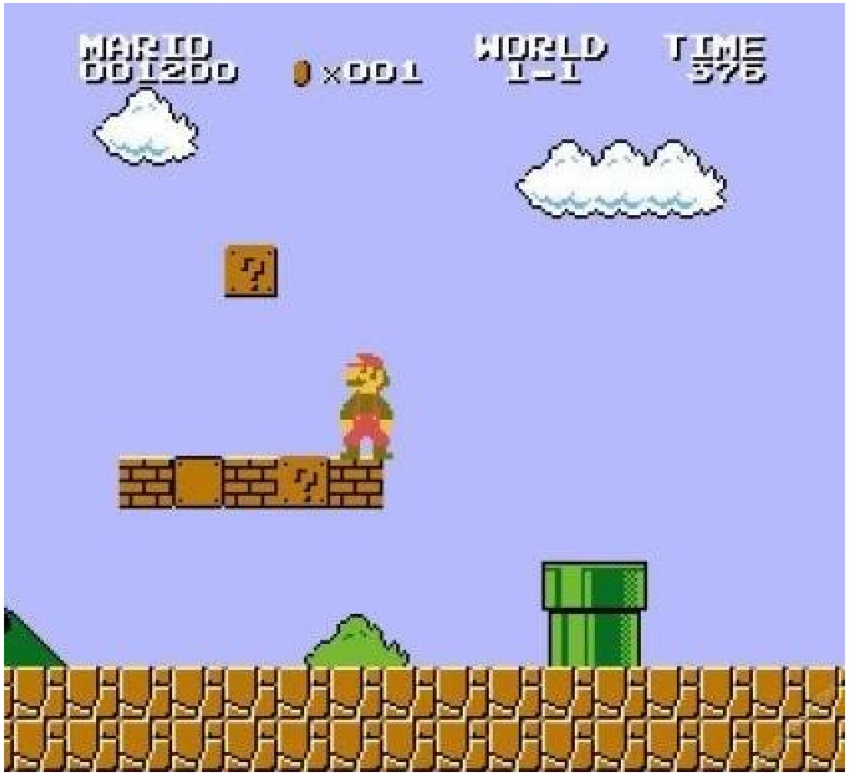
- CPU
  - Ricoh 2A03 8-bit processor (MOS Technology 6502 core)
- Controller input
  - 2 controller port
- Best-selling game
  - Super Mario Bros.
- Predecessor
  - Color TV Game
- Successor
  - Super Nintendo Entertainment System

Picture from Wikipedia

http://en.wikipedia.org/wiki/File:Famicom.jpg

Nintendo Entertainment System

- Ways to Save ROM size
  - The bushes in SMB are actually green-colored clouds
  - The bush is exactly the same as the leftmost cloud, other than color of course.

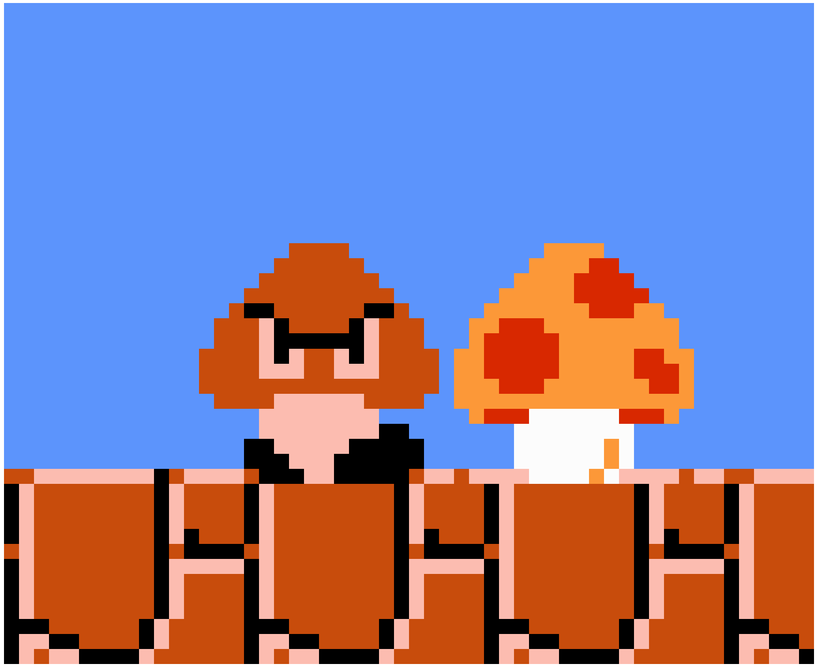


This is a screenshot of a non-free copyrighted video game or computer game, and thecopyright for it is most likely held by the company or person that developed the game. It is believed that the use of a limited number of web-resolution screenshots

Adapted from
http://www.destructoid.com/blogs/Jordan+Devore/the-bushes-in-smb-are-actually-green-colored-clouds-and-other-crazy-facts-67302.phtml

Nintendo Entertainment System

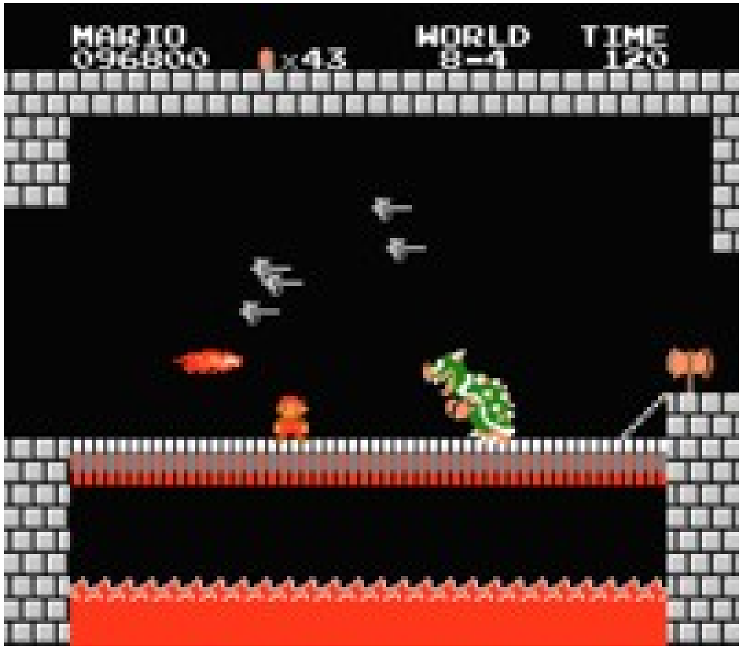
- Ways to Save ROM size
  - Another striking resemblance.

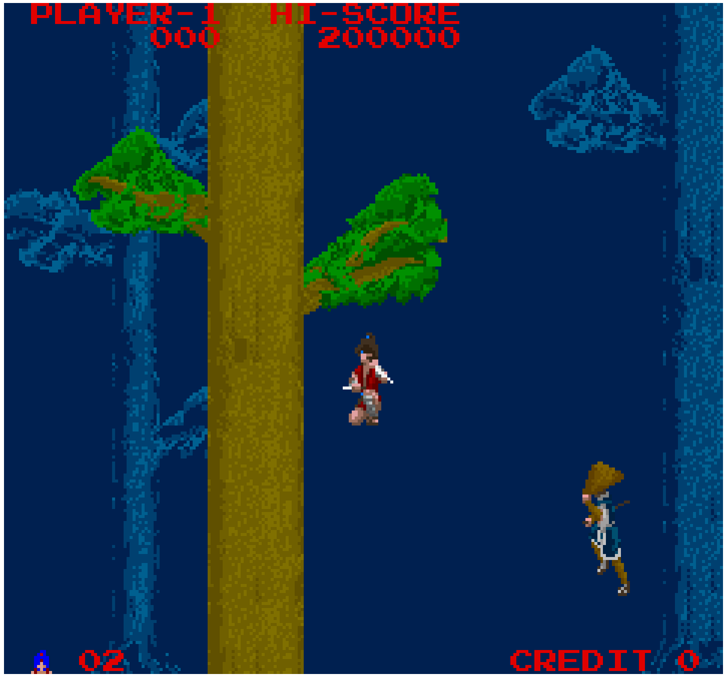


Adapted from
http://www.destructoid.com/blogs/Jordan+Devore/the-bushes-in-smb-are-actually-green-colored-clouds-and-other-crazy-facts-67302.phtml

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 MARIO ALL-STARS
POWER
EJECT
RESET
Nintendo
SUPER NINTENDO ENTERTAINMENT SYSTEM
SUPER NINTENDO ENTERTAINMENT SYSTEM
SELECT
START
X
A
Y
B

Picture from Wikipedia

http://en.wikipedia.org/wiki/File:SNES_800.jpg

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

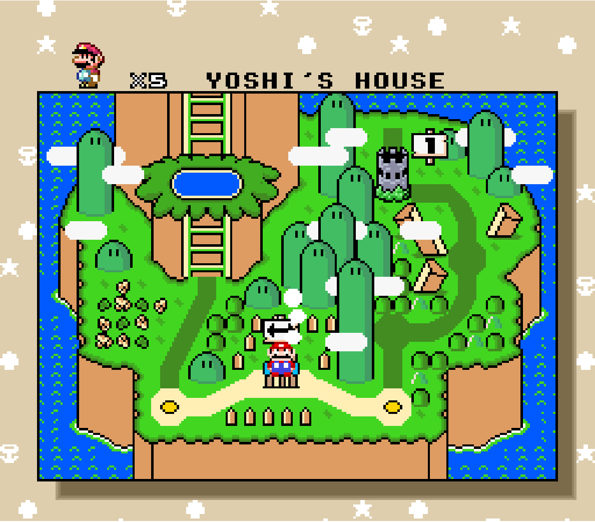
- Additional Features
    - DMA and HDMA
    - Timed IRQ
    - Parallel I/O processing
    - Hardware multiplication and division

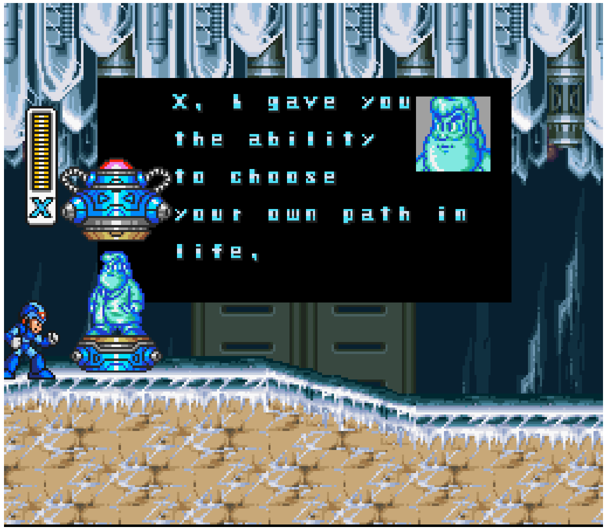
Super Nintendo Entertainment System

- 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

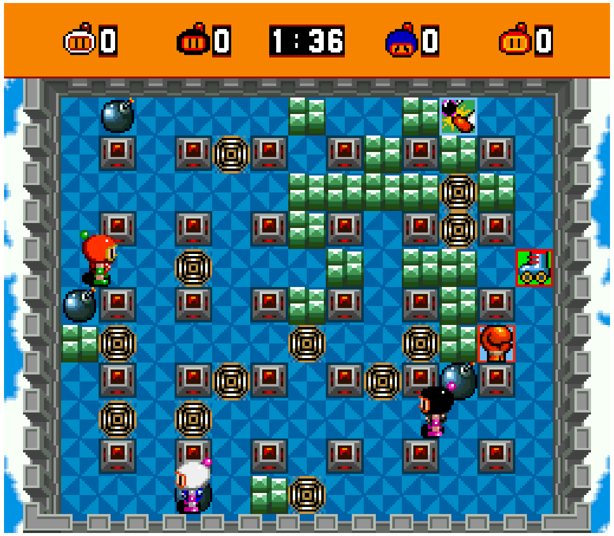
Super Nintendo Entertainment System

- Game
  - Super Mario World
  - Mega Man X
  - Super Bomber Man 5





X, I gave you
the ability
to choose
your own path in
life.



This is a screenshot of a non-free copyrighted video game or computer game, and thecopyright for it is most likely held by the company or person that developed the game. It is believed that the use of a limited number of web-resolution screenshots

Porting how-to

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

Getting started

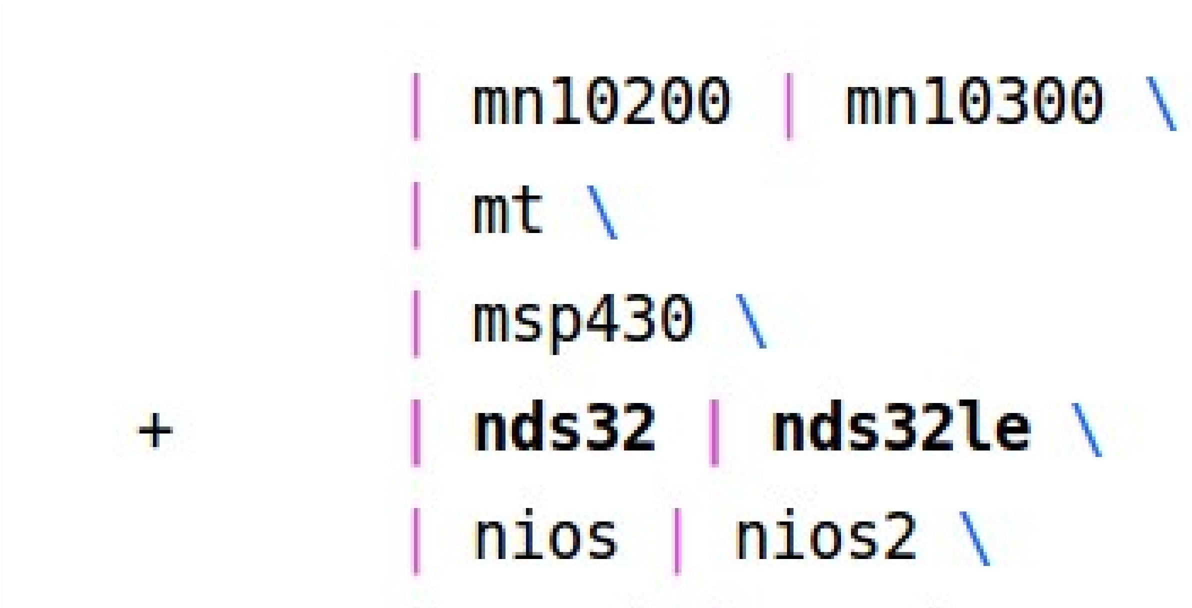
- Environment
  - Ubuntu 10.04
  - BSP 2.1
  - Andes tool chain v1.4
  - A working target

Environment settings for Andes tool chain

- Set the location of your tool chain
  - source bashrc.nds32le-linux-V0
  - export ANDESIGHT_ROOT=/home/path/toolchains/nds32-elf-n1213-43u1h
  - export PATH=$ANDESIGHT_ROOT/bin:$PATH

Modify config.sub

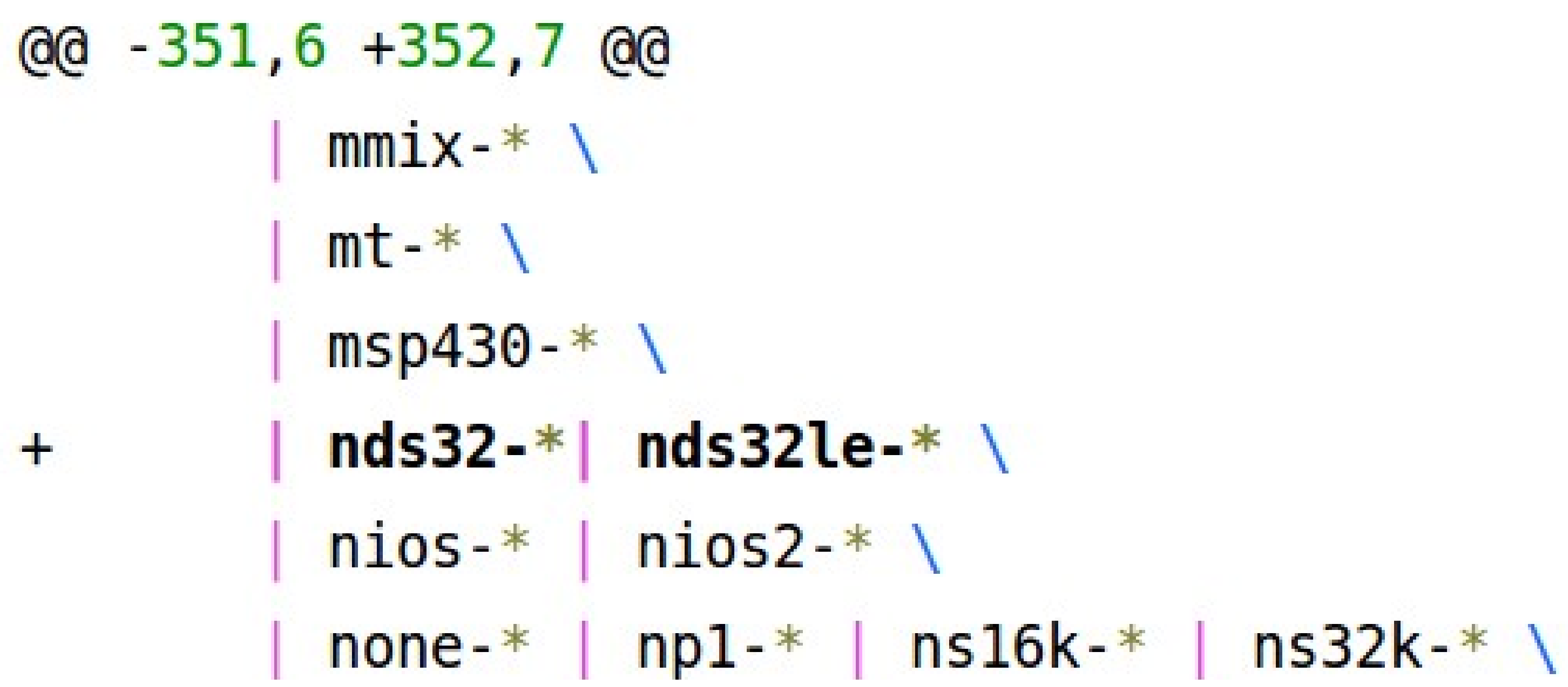
- Find the line below "Some are omitted here ..."
- I have already sent patch to config-patches@gnu.org



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

Modify config.sub

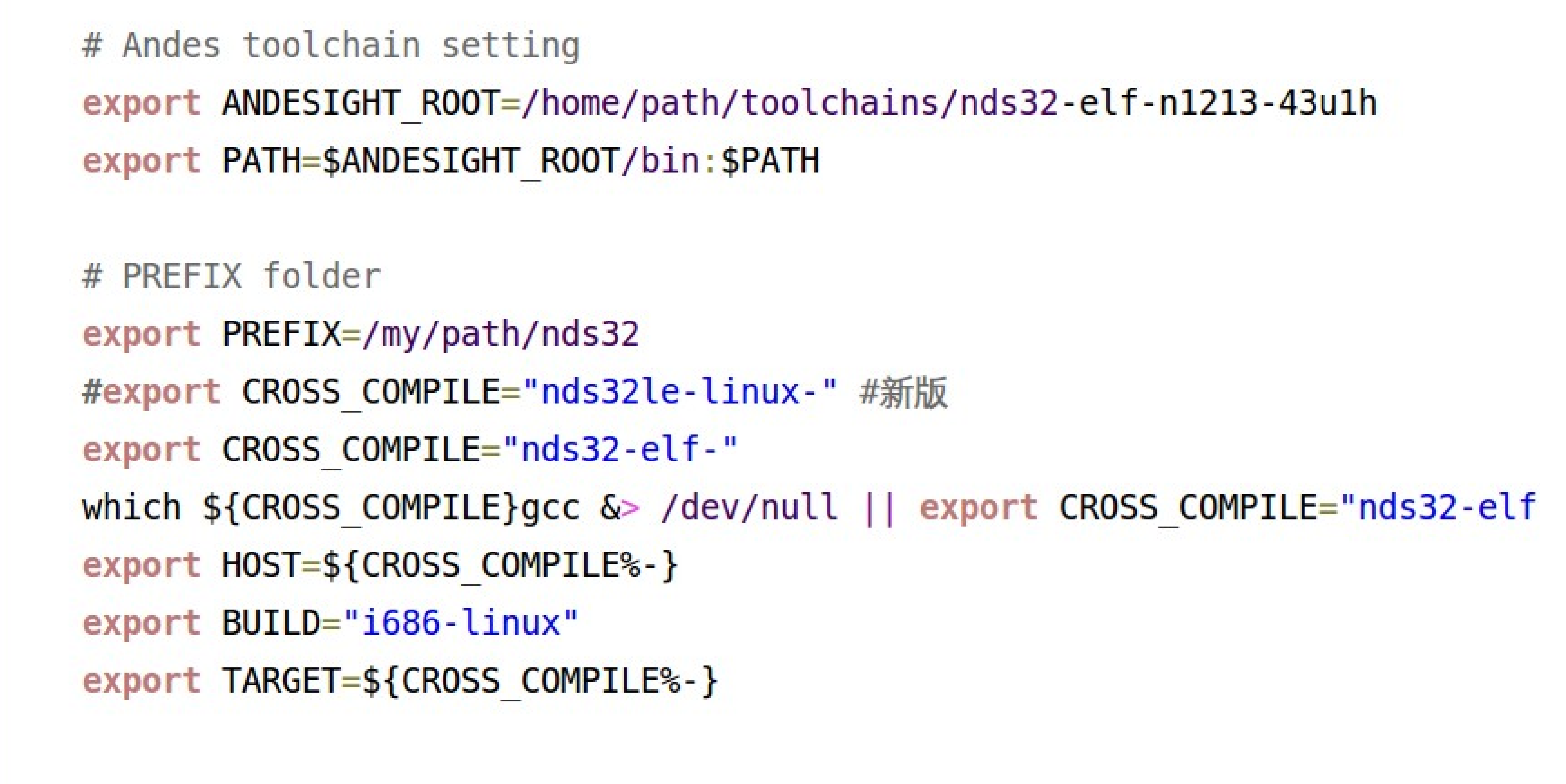
- Find the line below "Recognize the basic CPU types with company name."



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

Configure

- Using build scripts

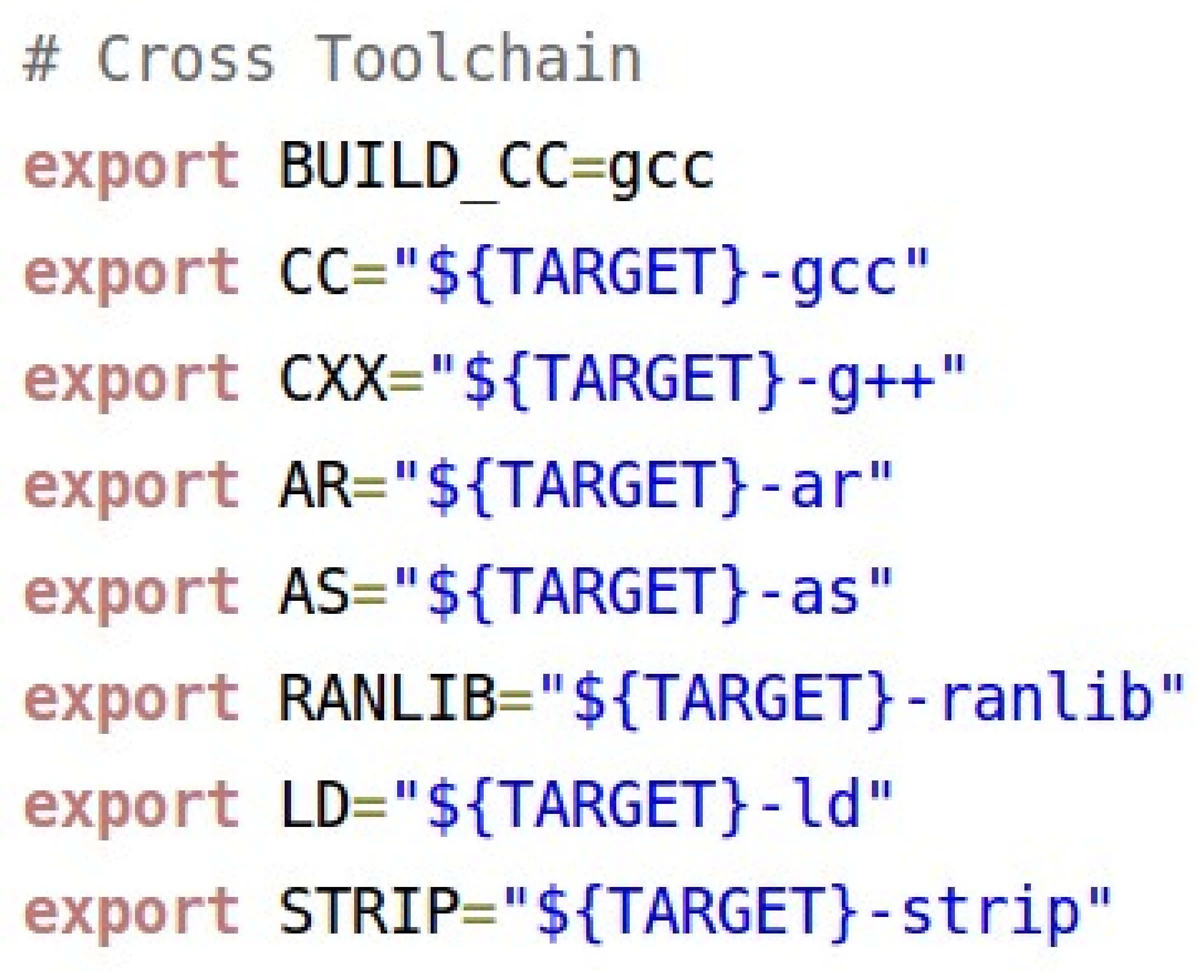


```
# Andes toolchain setting
export ANDESIGHT_ROOT=/home/path/toolchains/nds32-elf-n1213-43u1h
export PATH=$ANDESIGHT_ROOT/bin:$PATH

# 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%-}
```

Conscreenshot of a non-free copyrighted video gamefigure

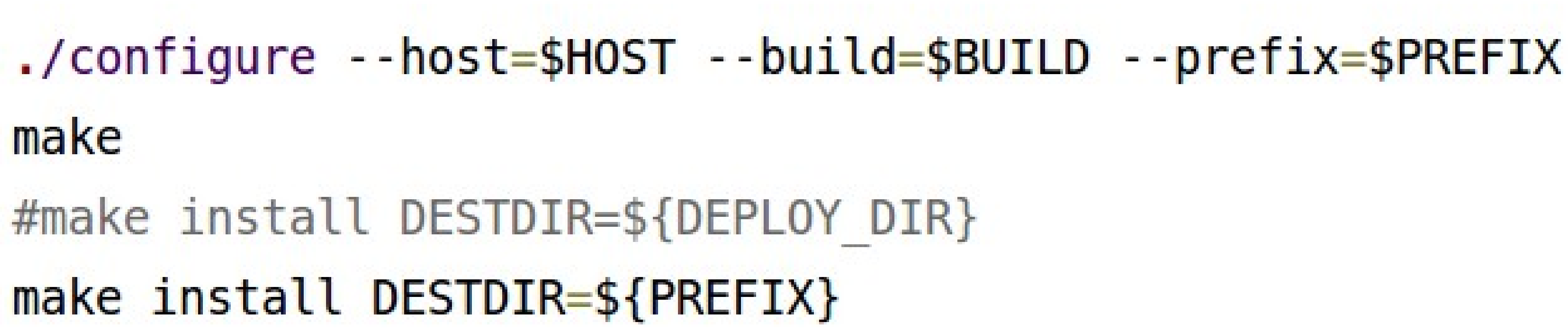
- Assign Andes toolchains



```
# 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"
```

Make and Install

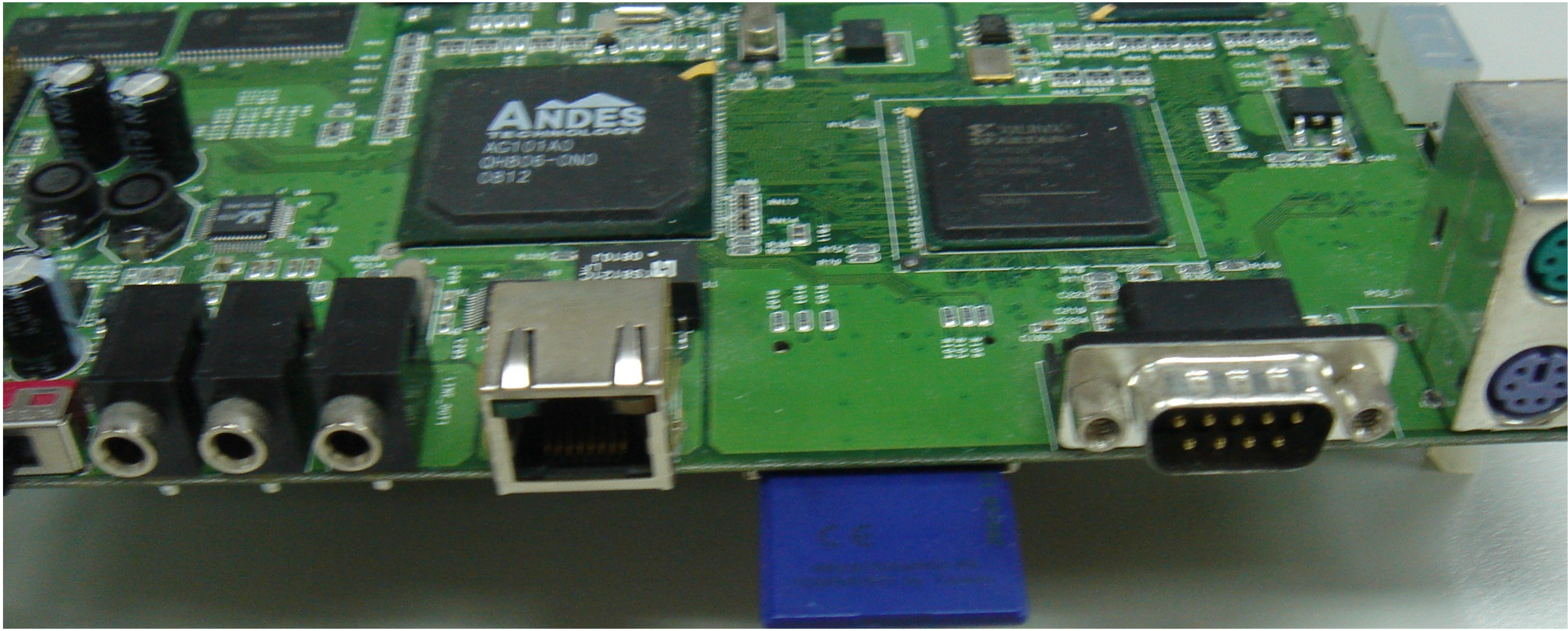
- The compile time error can find in this step



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

Deploy

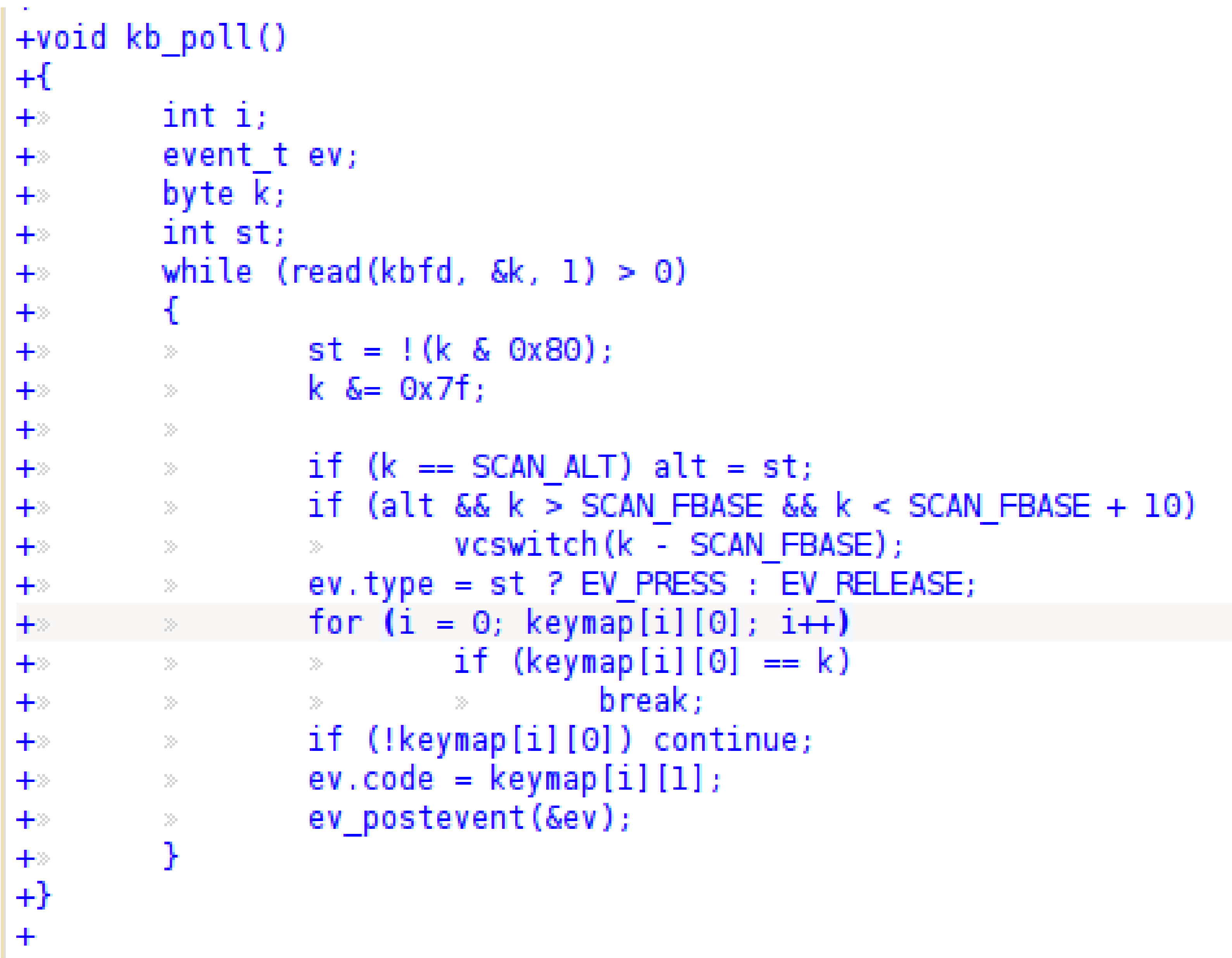
- Copy the folder of your building path to SD card
- Set environment variable of library



Add GPIO (Joystick) thread

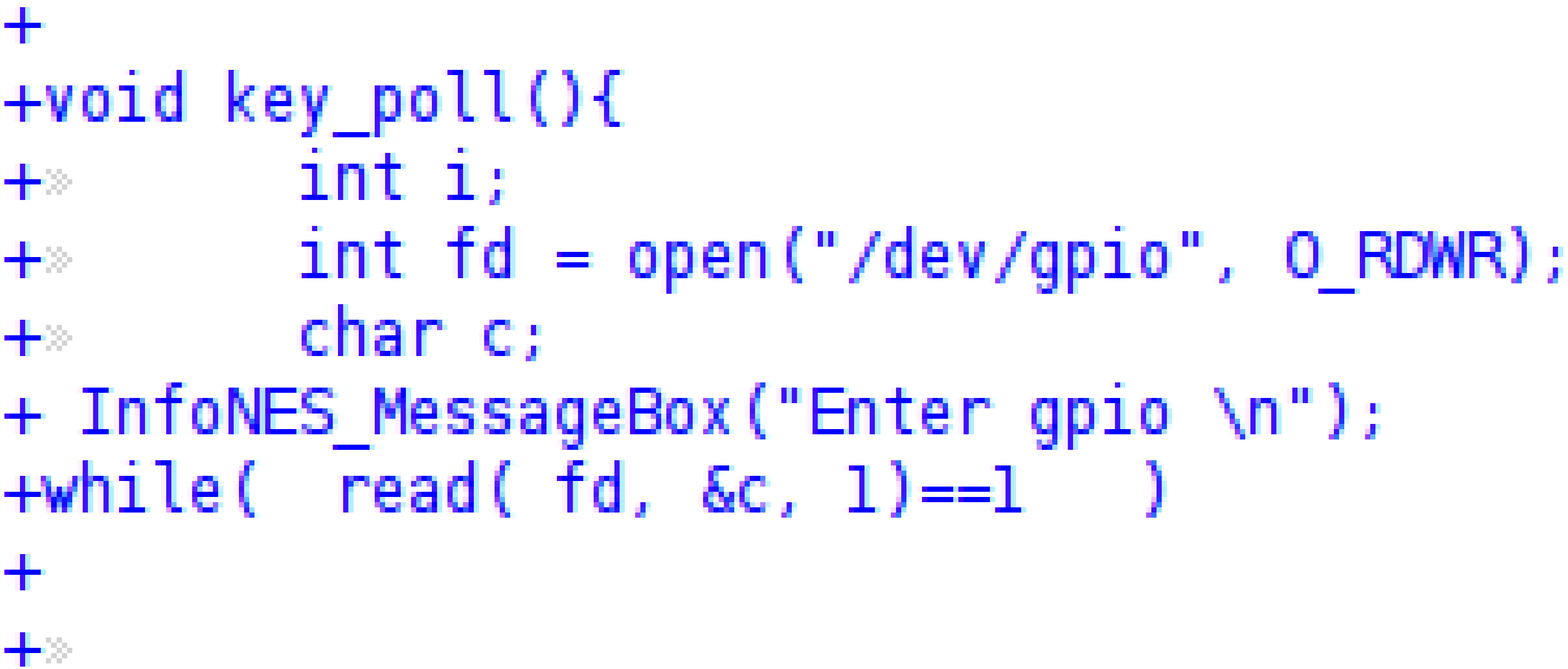
- Discard SDL Joystick
- kb_poll()
- /dev/gpio
- pthread

kb_poll()



```
+void kb_poll()
+{
+»      int i;
+»      event_t ev;
+»      byte k;
+»      int st;
+»      while (read(kbfd, &k, 1) > 0)
+»      {
+»      »       st = !(k & 0x80);
+»      »       k &= 0x7f;
+»      »
+»      »       if (k == SCAN_ALT) alt = st;
+»      »       if (alt && k > SCAN_FBASE && k < SCAN_FBASE + 10)
+»      »       »       vcswitch(k - SCAN_FBASE);
+»      »       ev.type = st ? EV_PRESS : EV_RELEASE;
+»      »       for (i = 0; keymap[i][0]; i++)
+»      »       »       if (keymap[i][0] == k)
+»      »       »       »       break;
+»      »       if (!keymap[i][0]) continue;
+»      »       ev.code = keymap[i][1];
+»      »       ev_postevent(&ev);
+»      }
+}
+
```

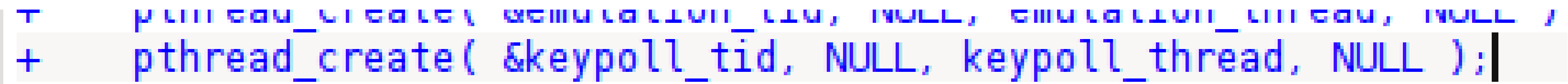
Open /dev/gpio



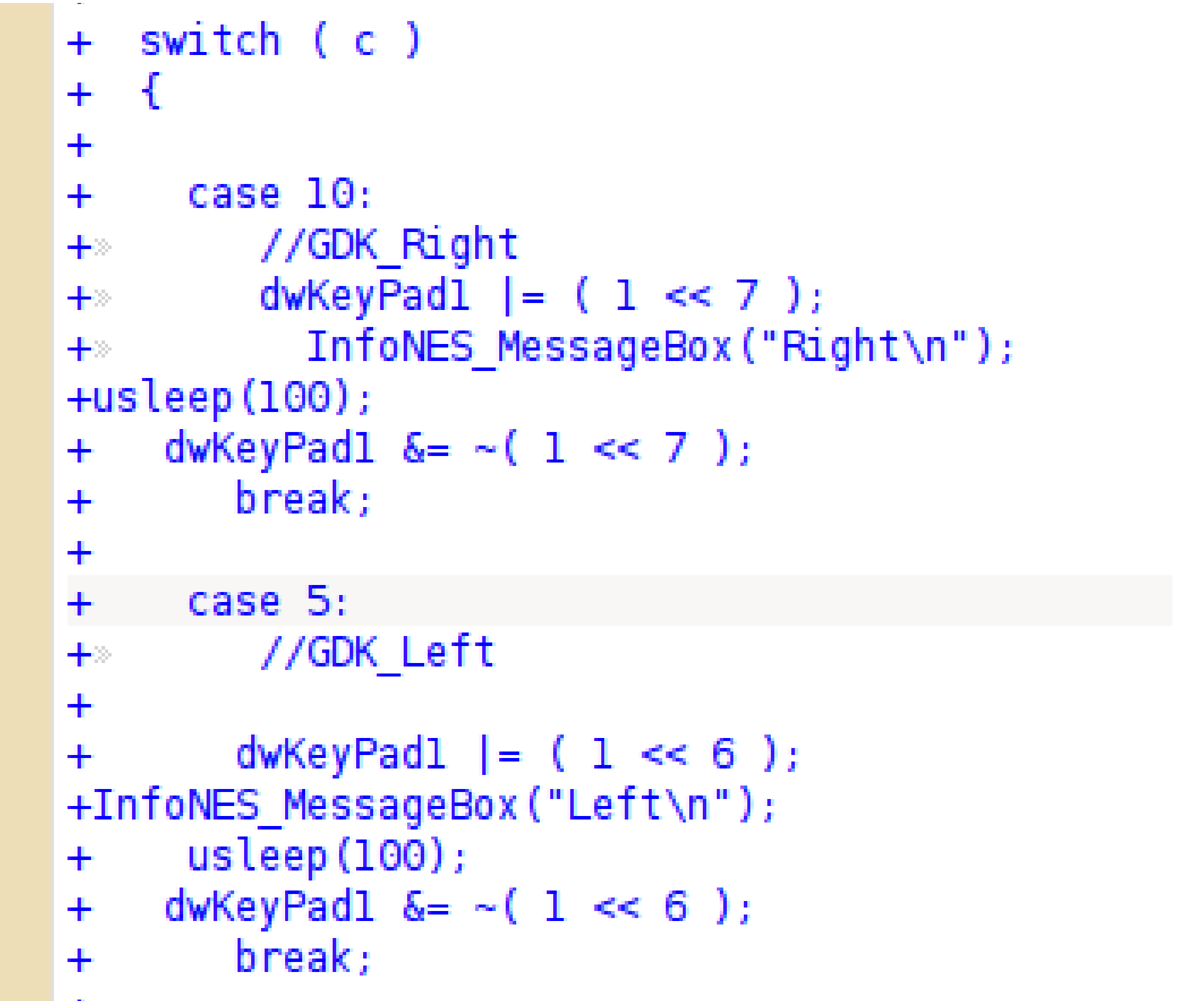
```
+
+void key_poll(){
+»      int i;
+»      int fd = open("/dev/gpio", O_RDWR);
+»      char c;
+ InfoNES_MessageBox("Enter gpio \n");
+while(  read( fd, &c, 1)==1   )
+
+»
```

Pthread create thread

- pthread_create



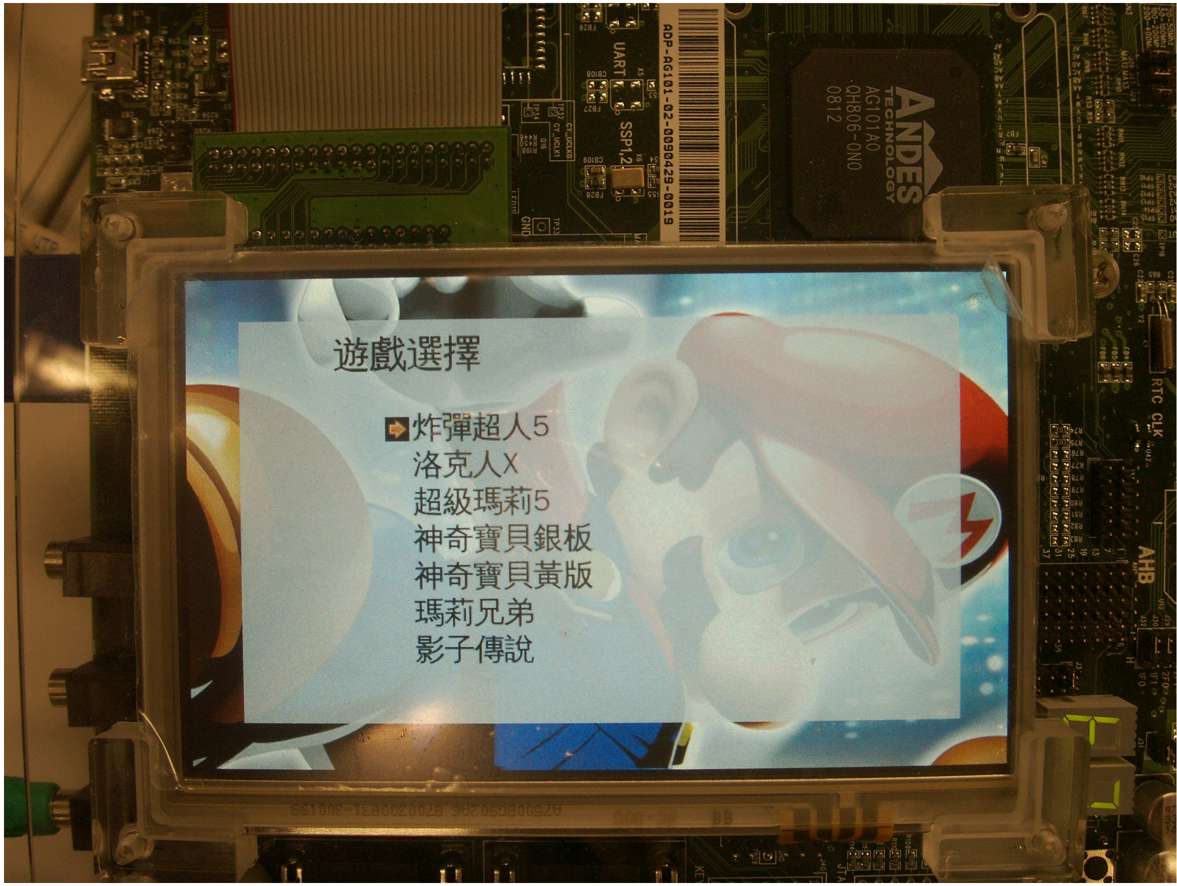
```
+     pthread_create( &emulation_tid, NULL, emulation_thread, NULL );
+     pthread_create( &keypoll_tid, NULL, keypoll_thread, NULL );
+
```



```
+   switch ( c )
+   {
+
+     case 10:
+        //GDK_Right
+        dwKeyPad1 |= ( 1 << 7 );
+          InfoNES_MessageBox("Right\n");
+usleep(100);
+    dwKeyPad1 &= ~( 1 << 7 );
+       break;
+
+     case 5:
+        //GDK_Left
+
+       dwKeyPad1 |= ( 1 << 6 );
+InfoNES_MessageBox("Left\n");
+     usleep(100);
+    dwKeyPad1 &= ~( 1 << 6 );
+       break;
```

Game Menu

- Custom shell scripts
- GIMP to edit image
- SDL read bitmap



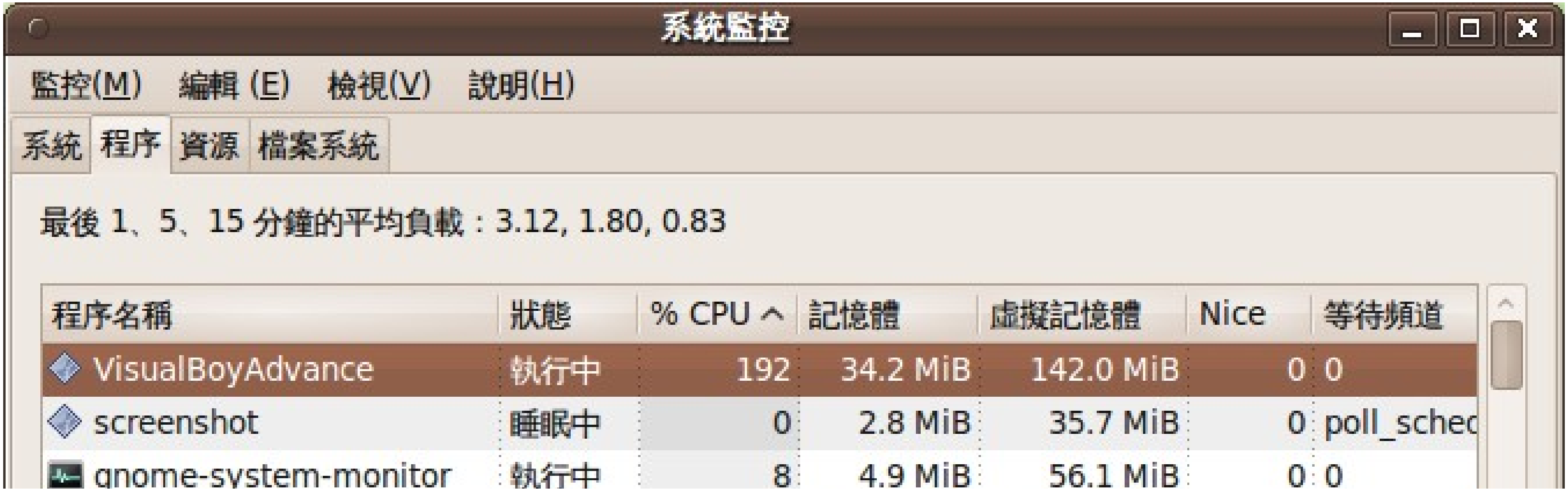
Floating point in sound generation code

- void BIOS_MidiKey2Freq()
- log("MidiKey2Freq: WaveData=%08x mk=%08x fp=%08x\n",reg[0].I, reg[1].I,reg[2].I);

```
int freq = CPUReadMemory(reg[0].I+4);
double tmp;
tmp = ((double)(180 - reg[1].I)) - ((double)reg[2].I / 256.f);
tmp = pow((double)2.f, tmp / 12.f);
reg[0].I = (int)((double)freq / tmp);
```

Resource usage and profiling

- VisualBoyAdvance on Intel(R) Core(TM)2 Quad CPU   Q6600 @ 2.40GHz
- 192/100 => 1.92 Core



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_schec
gnome-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

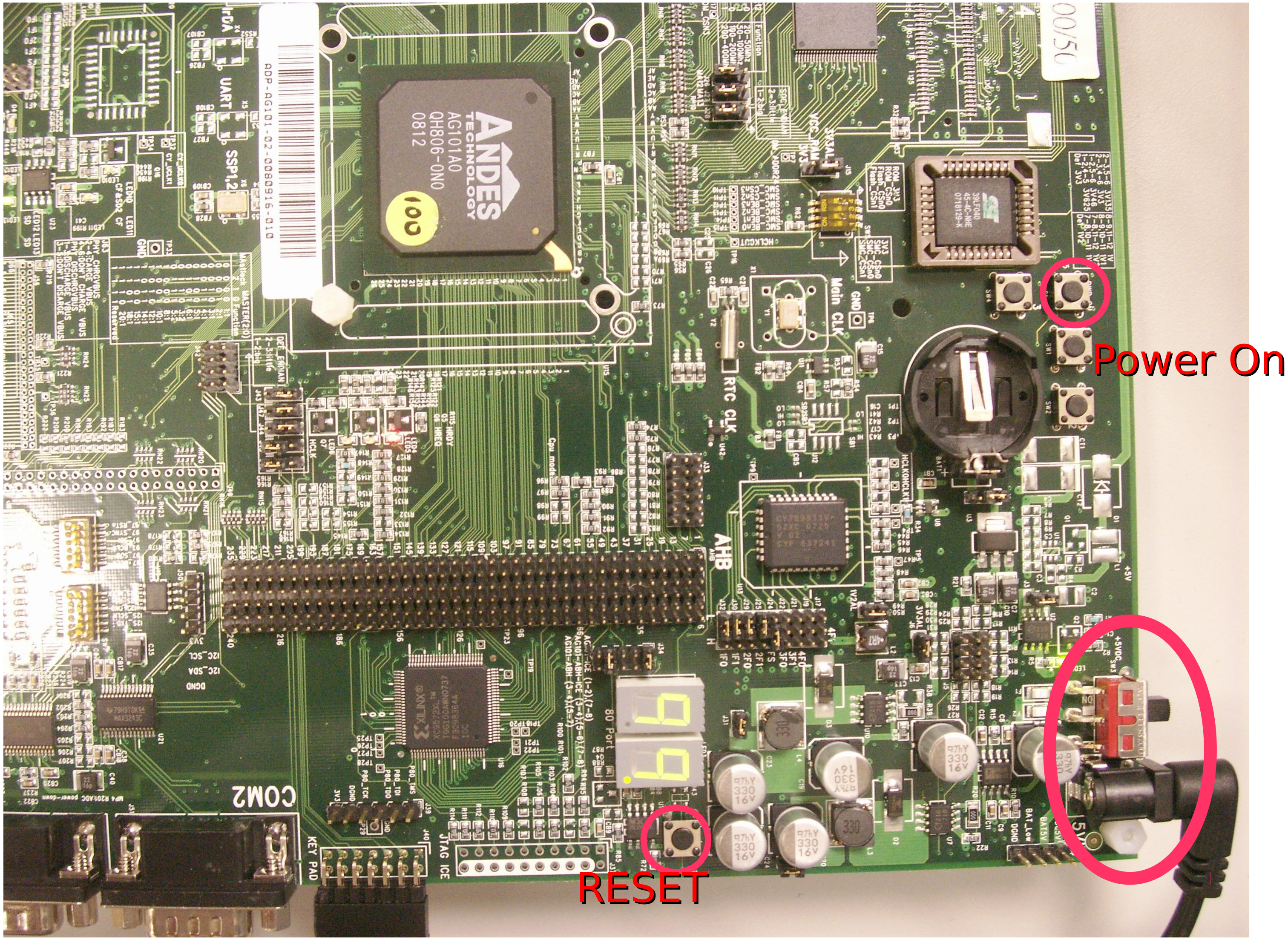
The ways of enhancement performance

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

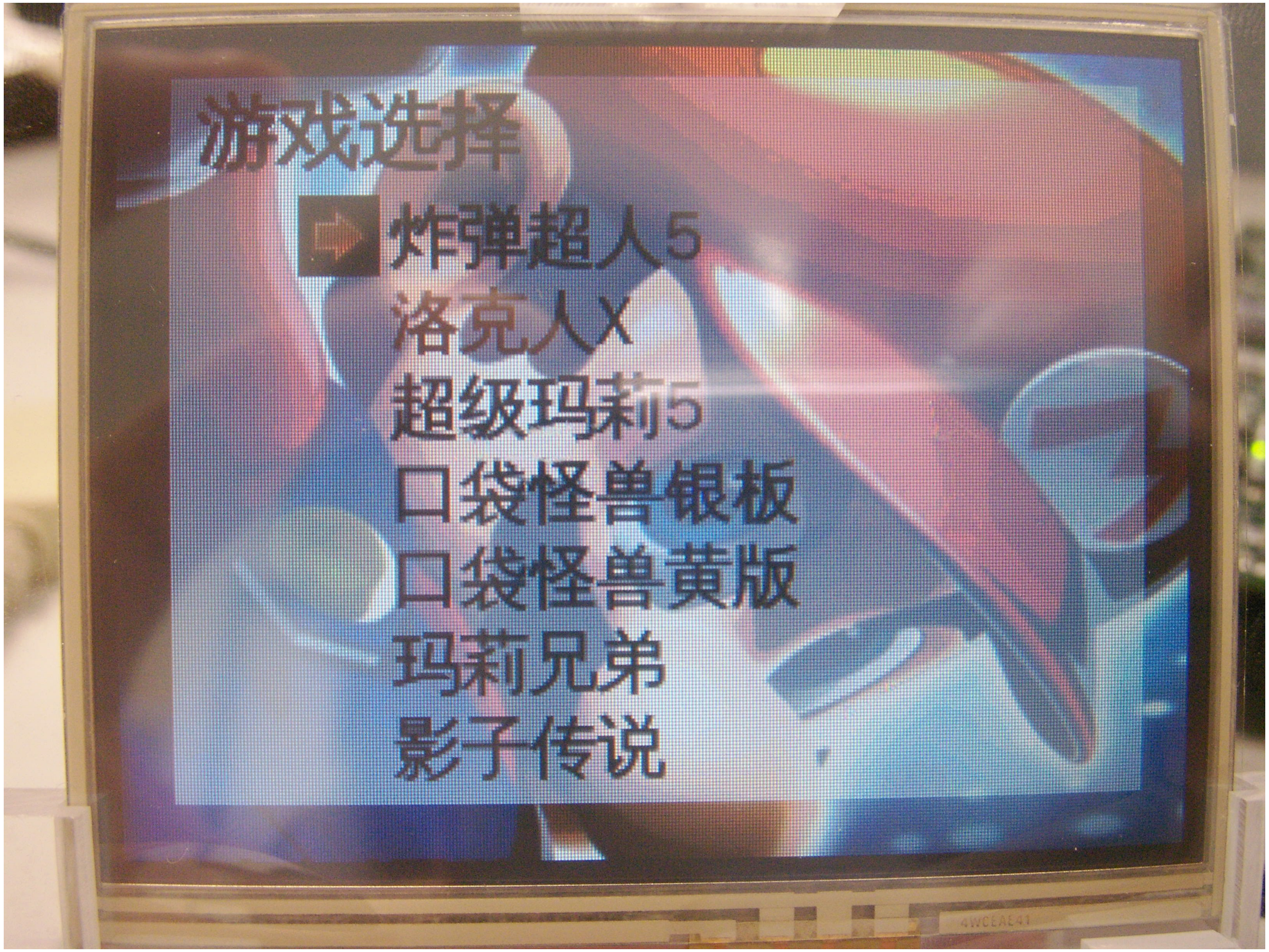
Have fun

- Power on
- Get into Game Menu
- Select game
- Play

Power on

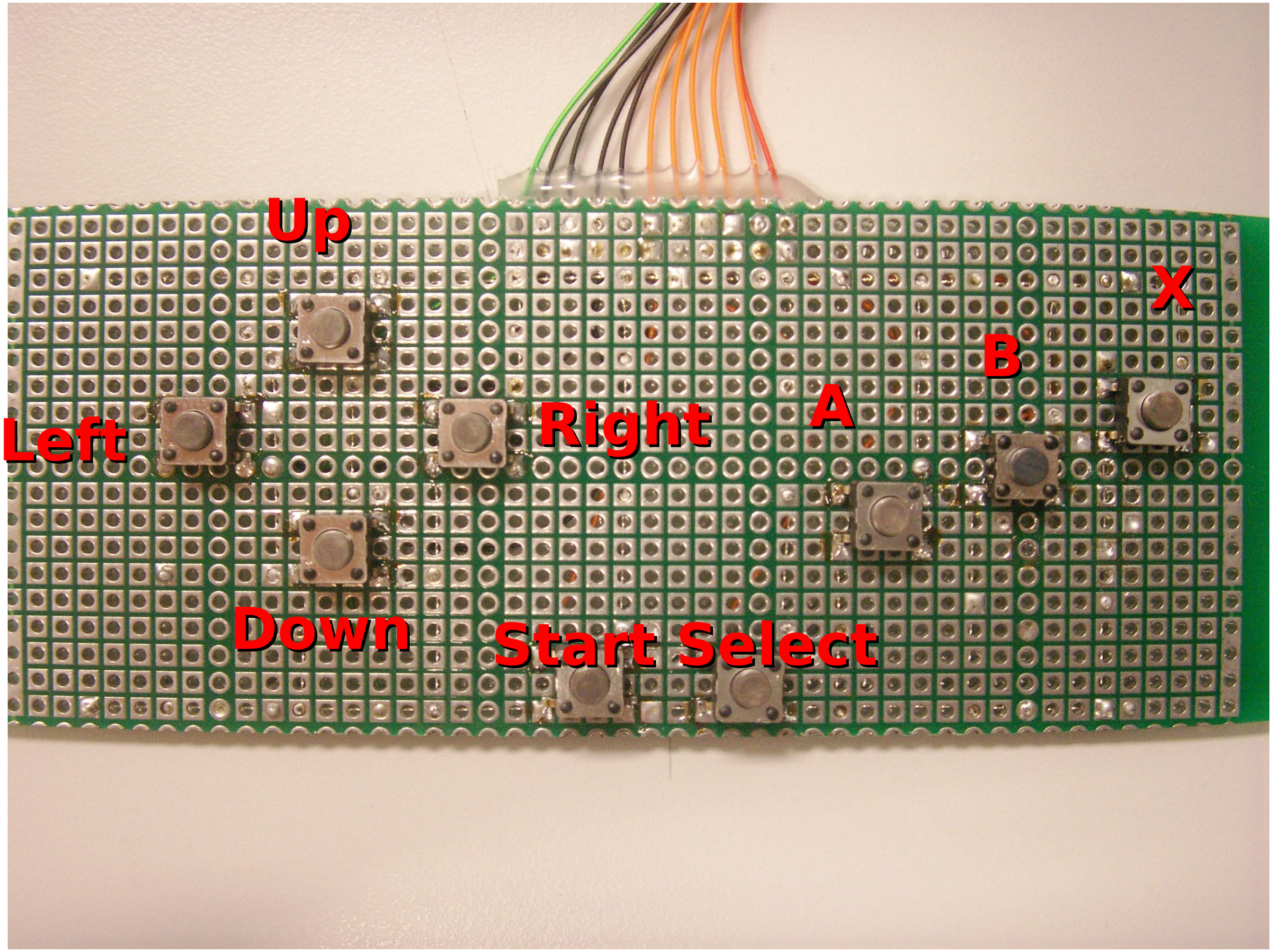

Power On
RESET
ANDES TECHNOLOGY
AG101A0
QH806-0N0
0812
UART
SSP1,2
Main CLK
RTC CLK
AHB
80 Port
COM2
KEY PAD
JTAG ICE

Game Menu



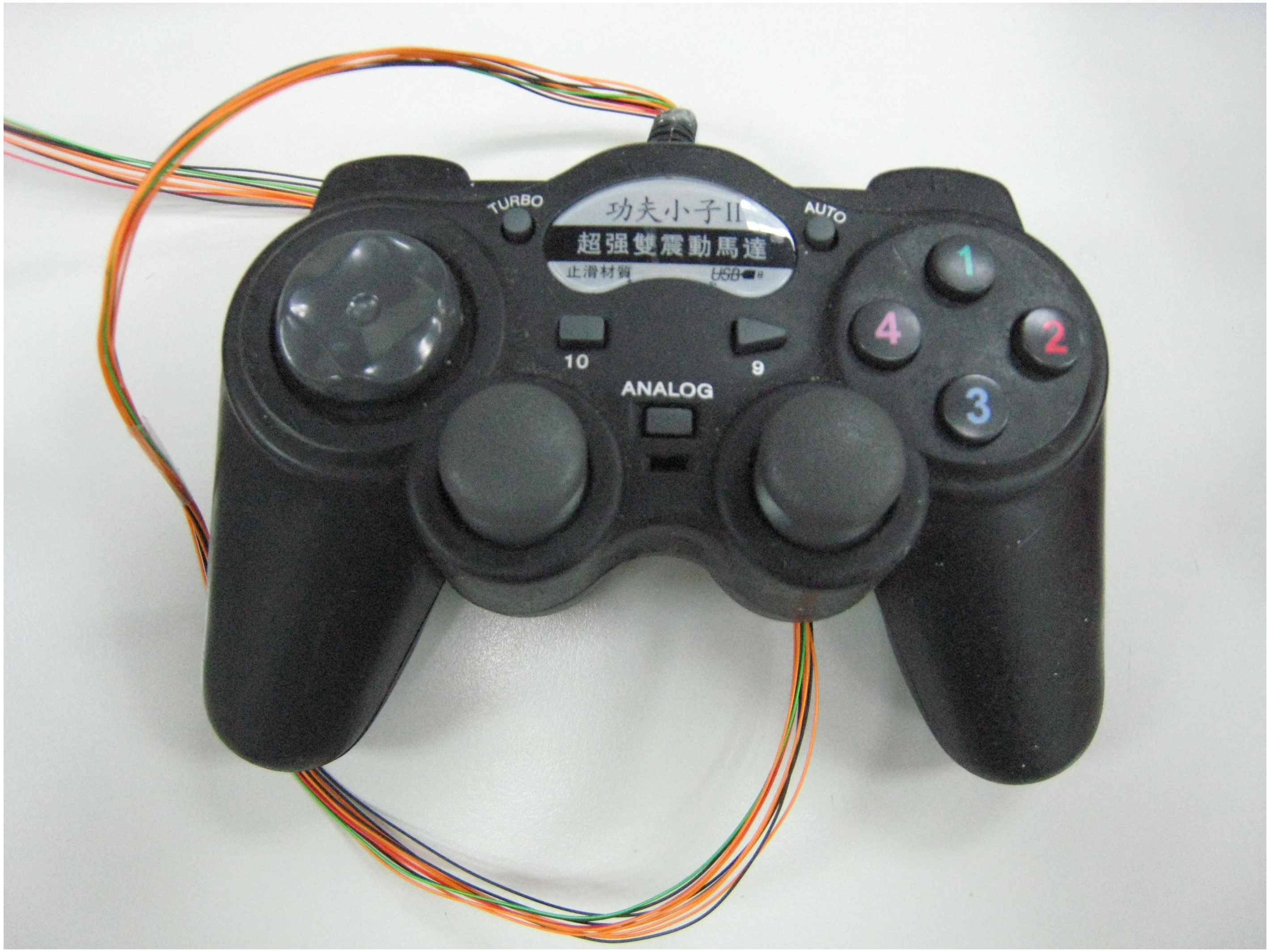
Joy Stick(1/3)

- Engineer version



Joy Stick(2/3)

- New version



Joy Stick(3/3)

- Press A button to select in game menu
- Press START button to start these game
- X button is only available in Super Nintendo game

Conclusion

- Have fun!

Q&A

SUPER BOMBERMAN 5
NORMAL GAME
BATTLE GAME
OPTION
©1997 HUDSON SOFT
CF
COM2
COM4
MIC IN
LINE IN
LINE OUT
AC97
XILINX
GAME

About this slide

- Except screenshots of a non-free copyrighted video game
- Using Creative Commons licenses
- CC-BY