Get Linux demo running

Atmel SAMA5D4x

TRAINING MANUAL

Table of Contents

Prerequisites

- Hardware Prerequisites
  - Atmel® SAMA5D4Xplained Ultra Evaluation Kit
  - SD Card
  - A USB serial TTL adapter
  - A USB cable with micro-A
- Software Prerequisites
  - SAM-BA 3.1.2
  - A terminal application (e.g. HyperTerminal on Microsoft Windows)
  - AT91 USB CDC driver
  - USB TTL serial Connector driver

Introduction

This Hands-On training shows the basic use of the SAMA5D4x using a SAMA5D4Xplained Ultra Evaluation Kit (SAMA5D4-XULT) and Linux demo running on the board, booting from NAND flash and SD card.

The goal of this Hands-On Training is to:

- Become familiar with the SAMA5D4x device.
- How to flashand boot up Linux demo on SAMA5D4-XULT.

Icon Key Identifiers

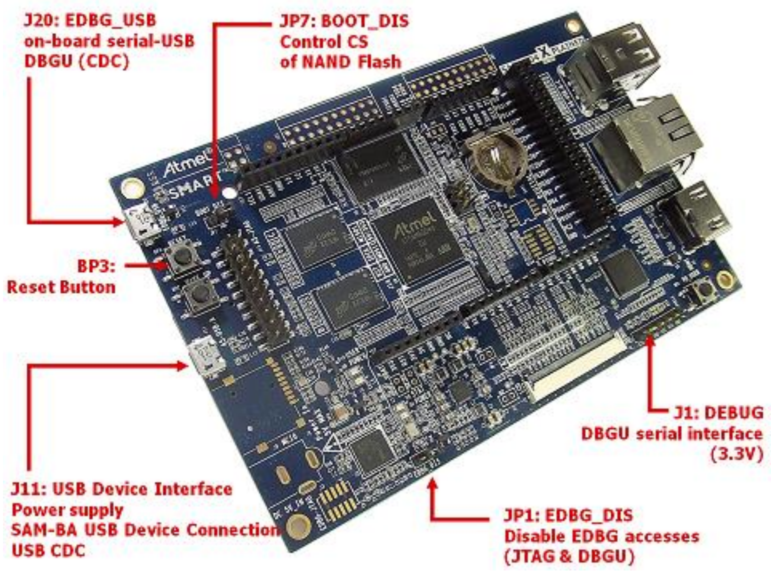
ℹ	**INFO**	Delivers contextual information about a specific topic
✎	**TIPS**	Highlights useful tips and techniques
📝	**TO DO**	Highlights objectives to be completed
✓	**RESULT**	Highlights the expected result of an assignment step
!	**WARNING**	Indicates important information
➔	**EXECUTE**	Highlights actions to be executed out of the target when necessary

1. Hardware Prerequisites

1.1 Atmel SAMA5D4 Evaluation Kit (SAMA5D4-XULT)

As the evaluation kit is a fast-prototyping and low-cost evaluation platform, the purpose of it is familiar with Linux demo of Atmel SAMA5D4 ARM Cortex®-A5-based microprocessor (MPU).

Figure 1-1. SAMA5D4-XULT Board Overview

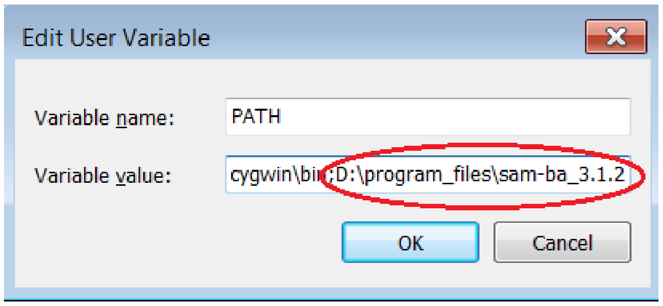


2. PC Software Prerequisites

2.1 Install SAM-BA 3.1.2

TO DO **Install sam-ba 3.1.2 by following the below steps.**

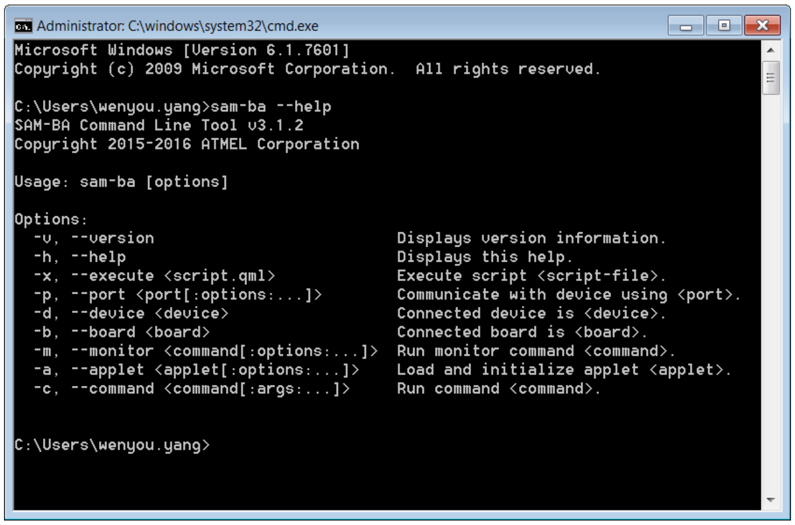
- Copy the sam-ba_3.1.2-win32.zipfile to a directory what you like, and unzip it.
- Add the path of sam-ba.exeto environment variable PATH.



Variable name: PATH

Variable value: cygwin\bin;D:\program_files\sam-ba_3.1.2

- Check if sam-ba_3.1.2 isinstalled properly, please enter sam-ba - - help command as below.



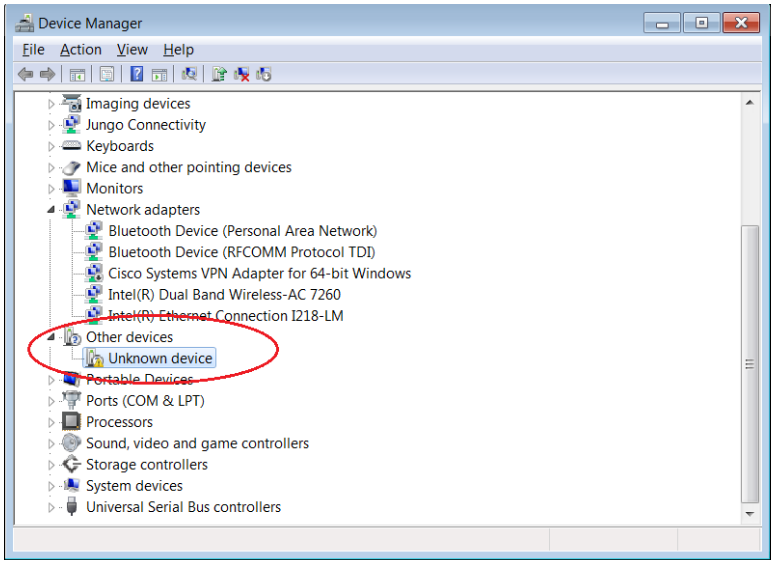
```
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation.  All rights reserved.

C:\Users\wenyou.yang>sam-ba --help
SAM-BA Command Line Tool v3.1.2
Copyright 2015-2016 ATMEL Corporation

Usage: sam-ba [options]

Options:
  -v, --version                         Displays version information.
  -h, --help                            Displays this help.
  -x, --execute <script.qml>            Execute script <script-file>.
  -p, --port <port[:options:...]>       Communicate with device using <port>.
  -d, --device <device>                 Connected device is <device>.
  -b, --board <board>                   Connected board is <board>.
  -m, --monitor <command[:options:...]> Run monitor command <command>.
  -a, --applet <applet[:options:...]>   Load and initialize applet <applet>.
  -c, --command <command[:args:...]>    Run command <command>.


C:\Users\wenyou.yang>
```

2.2 Install AT91 USB CDC driver

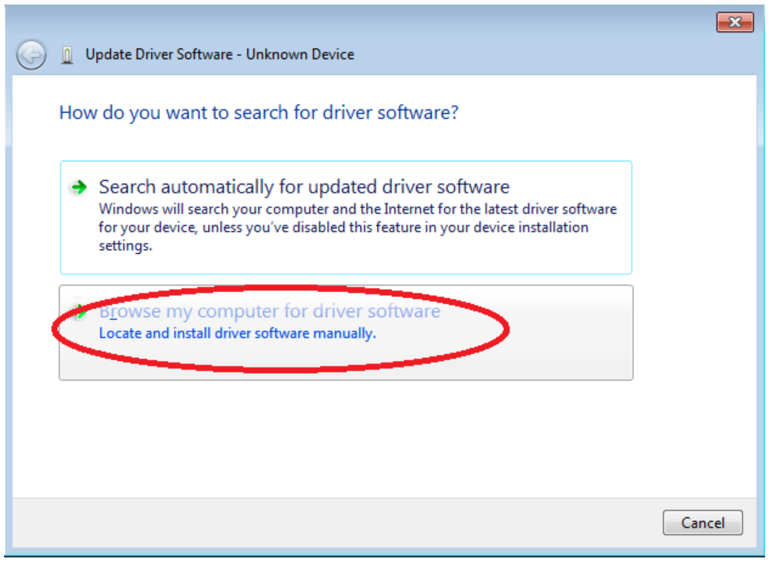
TO DO **Install AT91 USB CDC driver by following the below steps.**

WARNING This driver is necessary when sam-ba3.1.2 communicates with the board over the USB cable.

- Connect the USB to the board, and reset the board, the computershould detect the device.


Device Manager
File
Action
View
Help
Imaging devices
Jungo Connectivity
Keyboards
Mice and other pointing devices
Monitors
Network adapters
Bluetooth Device (Personal Area Network)
Bluetooth Device (RFCOMM Protocol TDI)
Cisco Systems VPN Adapter for 64-bit Windows
Intel(R) Dual Band Wireless-AC 7260
Intel(R) Ethernet Connection I218-LM
Other devices
Unknown device
Portable Devices
Ports (COM & LPT)
Processors
Sound, video and game controllers
Storage controllers
System devices
Universal Serial Bus controllers

- Install the driver which is under the driver directory of sam-ba3.1.2.

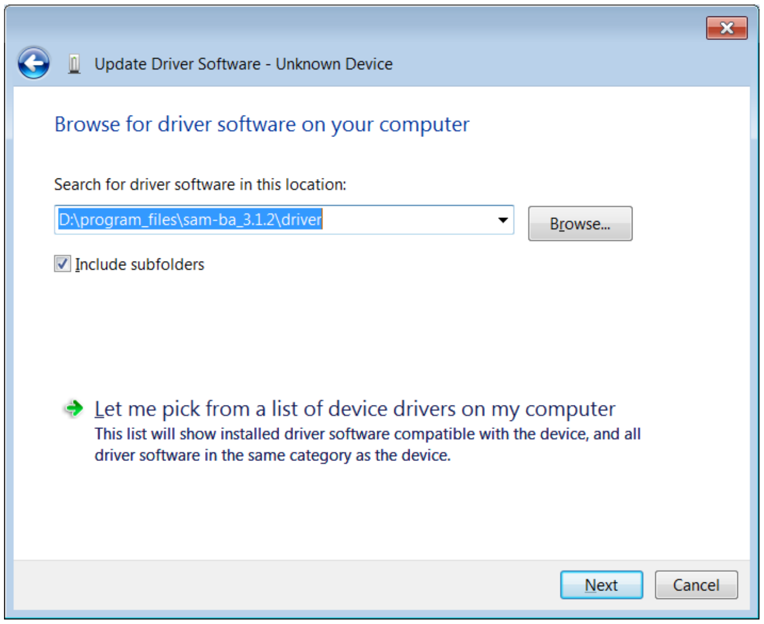


Update Driver Software - Unknown Device

How do you want to search for driver software?

Search automatically for updated driver software
Windows will search your computer and the Internet for the latest driver software for your device, unless you've disabled this feature in your device installation settings.

Browse my computer for driver software
Locate and install driver software manually.



Update Driver Software - Unknown Device

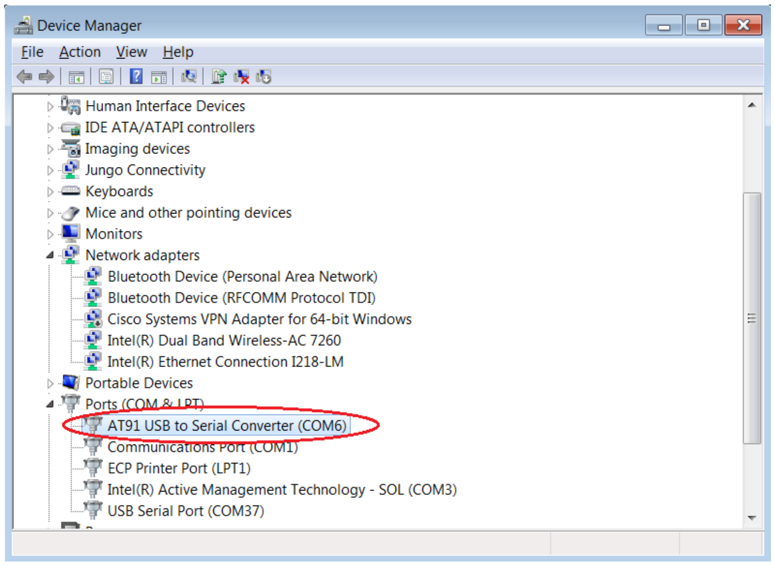
Browse for driver software on your computer

Search for driver software in this location:
D:\program_files\sam-ba_3.1.2\driver

☑ Include subfolders

Let me pick from a list of device drivers on my computer
This list will show installed driver software compatible with the device, and all driver software in the same category as the device.

- AT91 USB to Serial Converter should appear in Device Manager when the driver installed successfully.

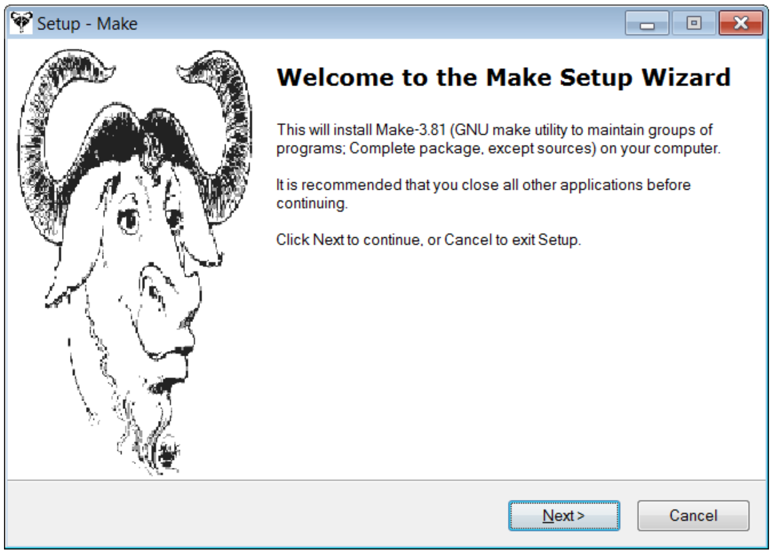


Human Interface Devices
IDE ATA/ATAPI controllers
Imaging devices
Jungo Connectivity
Keyboards
Mice and other pointing devices
Monitors
Network adapters
Bluetooth Device (Personal Area Network)
Bluetooth Device (RFCOMM Protocol TDI)
Cisco Systems VPN Adapter for 64-bit Windows
Intel(R) Dual Band Wireless-AC 7260
Intel(R) Ethernet Connection I218-LM
Portable Devices
Ports (COM & LPT)
AT91 USB to Serial Converter (COM6)
Communications Port (COM1)
ECP Printer Port (LPT1)
Intel(R) Active Management Technology - SOL (COM3)
USB Serial Port (COM37)

2.3 Install GnuWin32 make

TO DO Install GnuWin32 make by following the below steps.

- Double click make-3.81.exe to install make tool.



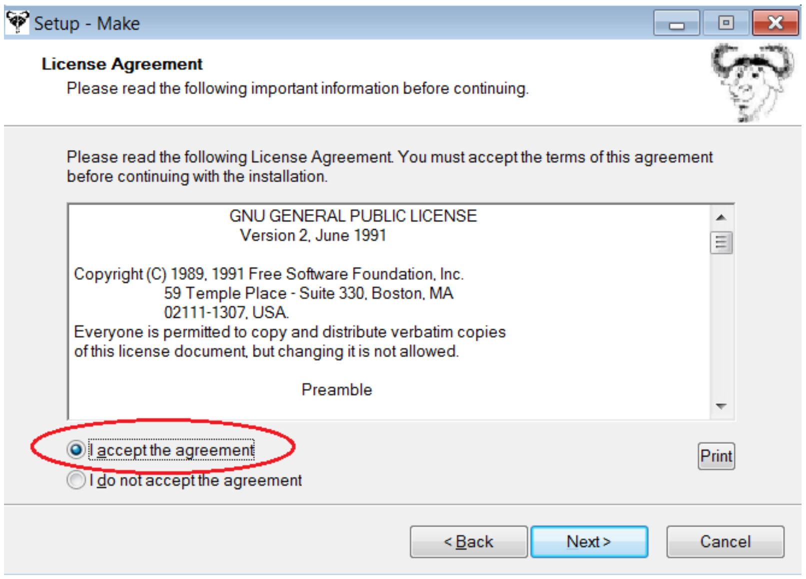
Welcome to the Make Setup Wizard

This will install Make-3.81 (GNU make utility to maintain groups of programs; Complete package, except sources) on your computer.

It is recommended that you close all other applications before continuing.

Click Next to continue, or Cancel to exit Setup.

- Accept the agreement, click “Next” to continue.


Setup - Make

License Agreement

Please read the following important information before continuing.

Please read the following License Agreement. You must accept the terms of this agreement before continuing with the installation.

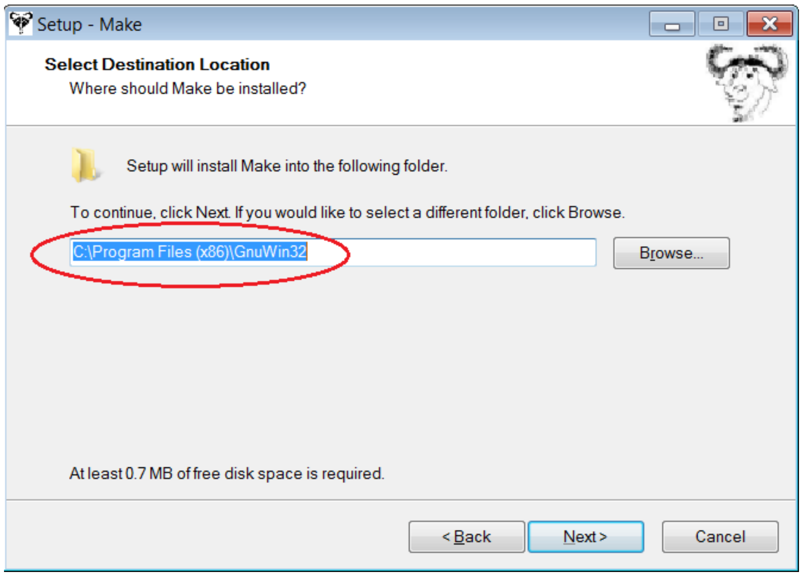
GNU GENERAL PUBLIC LICENSE
Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
59 Temple Place - Suite 330, Boston, MA
02111-1307, USA.
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

[✓ I accept the agreement]
I do not accept the agreement

- Choose the installation folder, and click "Next” to continue.


Setup - Make

Select Destination Location

Where should Make be installed?

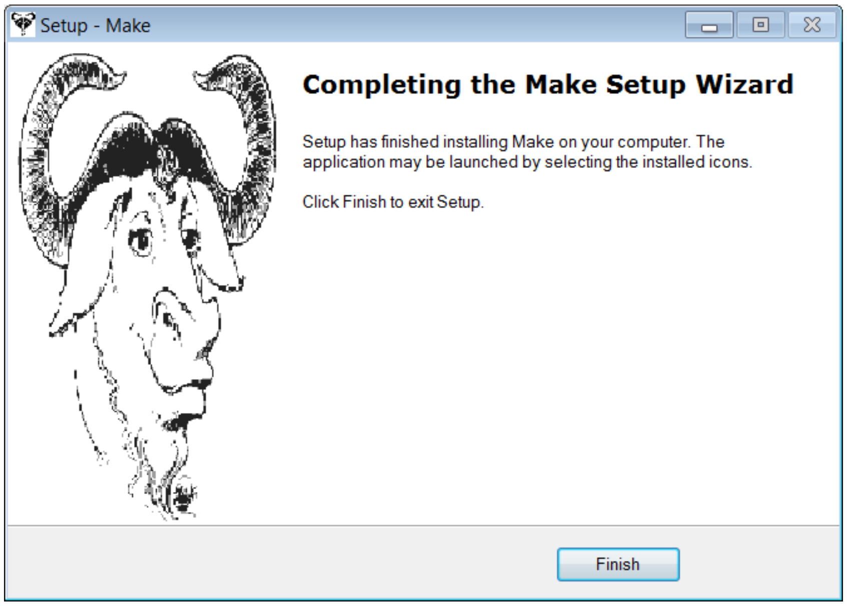
Setup will install Make into the following folder.

To continue, click Next. If you would like to select a different folder, click Browse.

C:\Program Files (x86)\GnuWin32

At least 0.7 MB of free disk space is required.

- Continue “Next” to “Finish”.

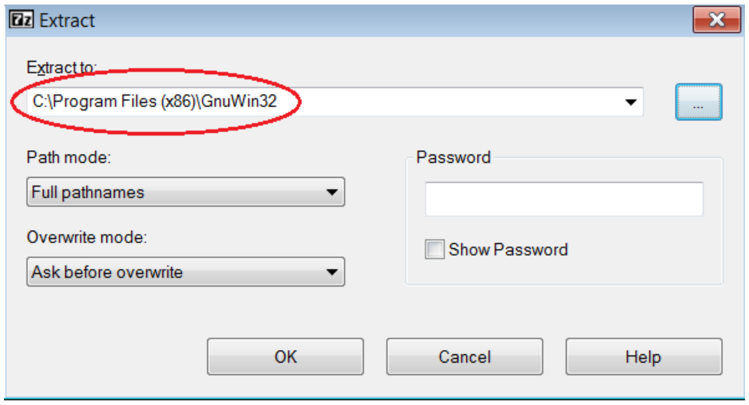

Setup - Make

Completing the Make Setup Wizard

Setup has finished installing Make on your computer. The application may be launched by selecting the installed icons.

Click Finish to exit Setup.

- Extract make-3.81-dep.zip to the make installation folder.


Extract

Extract to: C:\Program Files (x86)\GnuWin32

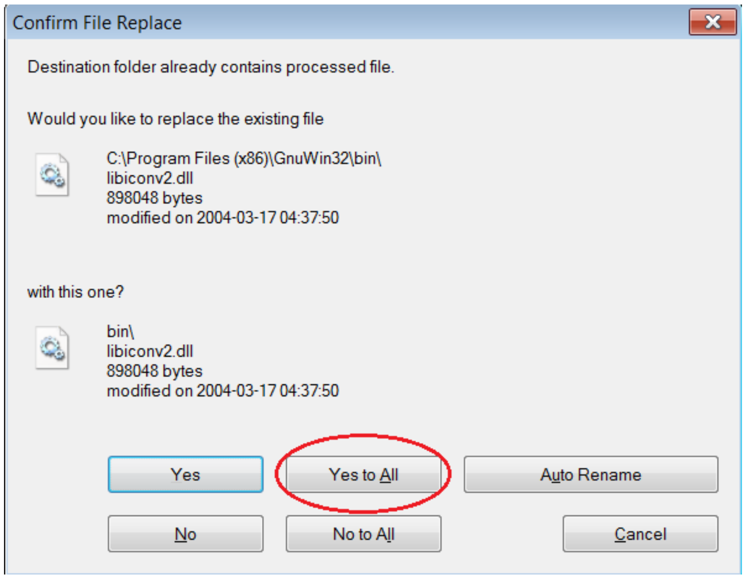
Path mode: Full pathnames

Overwrite mode: Ask before overwrite

Password

Show Password

- Choose “yes to All” to replace the corresponding files.



Destination folder already contains processed file.

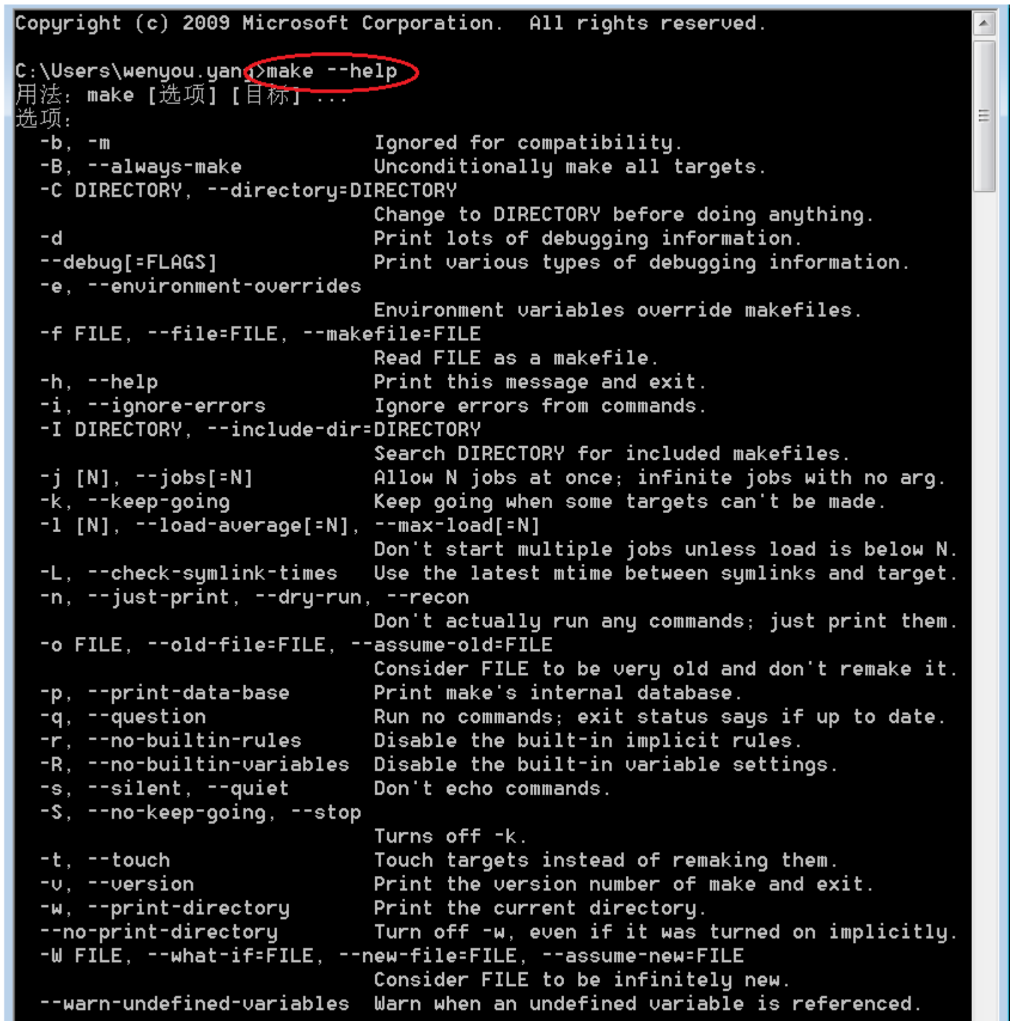
Would you like to replace the existing file

C:\Program Files (x86)\GnuWin32\bin\
libiconv2.dll
898048 bytes
modified on 2004-03-17 04:37:50

with this one?

bin\
libiconv2.dll
898048 bytes
modified on 2004-03-17 04:37:50

- Check if make isinstalled properly, please enter make - - helpcommand as below.



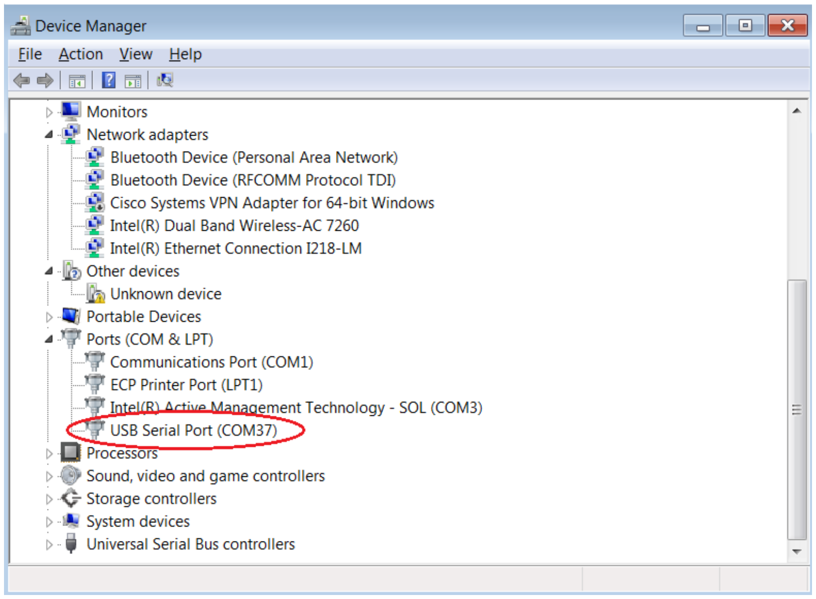
```
Copyright (c) 2009 Microsoft Corporation.  All rights reserved.

C:\Users\wenyou.yang>make --help
用法: make [选项] [目标] ...
选项:
  -b, -m                      Ignored for compatibility.
  -B, --always-make           Unconditionally make all targets.
  -C DIRECTORY, --directory=DIRECTORY
                              Change to DIRECTORY before doing anything.
  -d                          Print lots of debugging information.
  --debug[=FLAGS]             Print various types of debugging information.
  -e, --environment-overrides
                              Environment variables override makefiles.
  -f FILE, --file=FILE, --makefile=FILE
                              Read FILE as a makefile.
  -h, --help                  Print this message and exit.
  -i, --ignore-errors         Ignore errors from commands.
  -I DIRECTORY, --include-dir=DIRECTORY
                              Search DIRECTORY for included makefiles.
  -j [N], --jobs[=N]          Allow N jobs at once; infinite jobs with no arg.
  -k, --keep-going            Keep going when some targets can't be made.
  -l [N], --load-average[=N], --max-load[=N]
                              Don't start multiple jobs unless load is below N.
  -L, --check-symlink-times   Use the latest mtime between symlinks and target.
  -n, --just-print, --dry-run, --recon
                              Don't actually run any commands; just print them.
  -o FILE, --old-file=FILE, --assume-old=FILE
                              Consider FILE to be very old and don't remake it.
  -p, --print-data-base       Print make's internal database.
  -q, --question              Run no commands; exit status says if up to date.
  -r, --no-builtin-rules      Disable the built-in implicit rules.
  -R, --no-builtin-variables  Disable the built-in variable settings.
  -s, --silent, --quiet       Don't echo commands.
  -S, --no-keep-going, --stop
                              Turns off -k.
  -t, --touch                 Touch targets instead of remaking them.
  -v, --version               Print the version number of make and exit.
  -w, --print-directory       Print the current directory.
  --no-print-directory        Turn off -w, even if it was turned on implicitly.
  -W FILE, --what-if=FILE, --new-file=FILE, --assume-new=FILE
                              Consider FILE to be infinitely new.
  --warn-undefined-variables  Warn when an undefined variable is referenced.
```

2.4 Install a terminal application

You can install your familiar terminal application, such as putty, Tera Term, Realterm and HyperTerminal.

2.5 Install your USB serial TTL adapter driver

WARNING After the driver installed, please check USB Serial Port should appear in Device Manager.



```
▷ Monitors
◢ Network adapters
    Bluetooth Device (Personal Area Network)
    Bluetooth Device (RFCOMM Protocol TDI)
    Cisco Systems VPN Adapter for 64-bit Windows
    Intel(R) Dual Band Wireless-AC 7260
    Intel(R) Ethernet Connection I218-LM
◢ Other devices
    Unknown device
▷ Portable Devices
◢ Ports (COM & LPT)
    Communications Port (COM1)
    ECP Printer Port (LPT1)
    Intel(R) Active Management Technology - SOL (COM3)
    USB Serial Port (COM37)
▷ Processors
▷ Sound, video and game controllers
▷ Storage controllers
▷ System devices
▷ Universal Serial Bus controllers
```

3. Hands-On

3.1 Prepare to flash the demo

3.1.1 Connect the USB to the board

- Close the JP7 (BOOT_DIS) to prevent booting from NAND Flash by disabling Flash Chip Selects.
- Connect a USB micro-A cable to the board (J11 5V-USB-A) and power the board.
- Check whether the board is found in your computer as a USB device. AT91 USB to Serial Converter should appear in Device Manager if everything is ready.

TIPS SAMA5D4-XULT can also be powered through the EDBG interface or the USB-A interface.

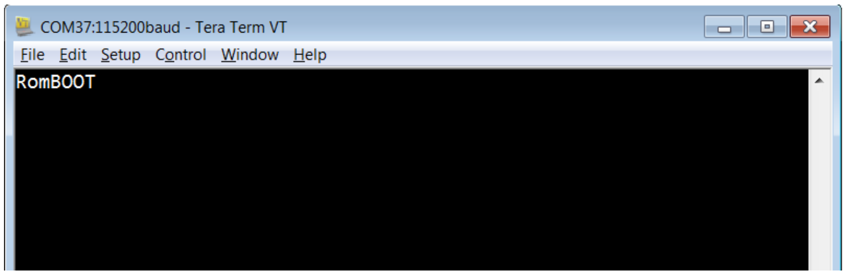
TIPS Port number for serial port can be checked in Device Manager.

3.1.2 Connect the board'sDBGU to the computer with a USB serial TTL adapter

- The Serial port setup parameter is 115200 8-N-1.

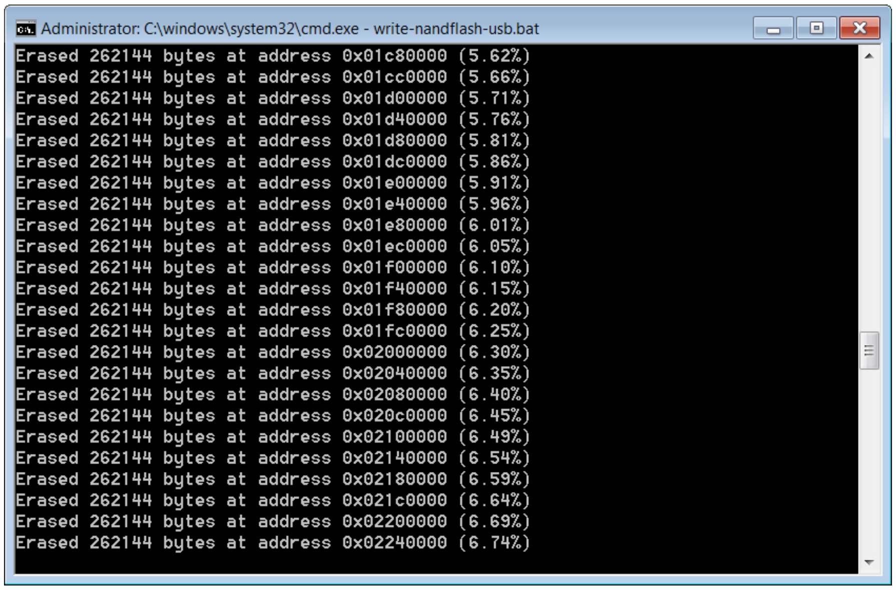
3.2 Run script to flash the demo

- Close the JP7 (BOOT_DIS) to prevent booting from NAND Flash by disabling Flash Chip Selects.
- Reset the board by press the RESET button.
- Please make sure **"RomBOOT"** appears in the Terminal and AT91 USB to Serial Converter should appear in Device Manager.


COM37:115200baud - Tera Term VT
File Edit Setup Control Window Help

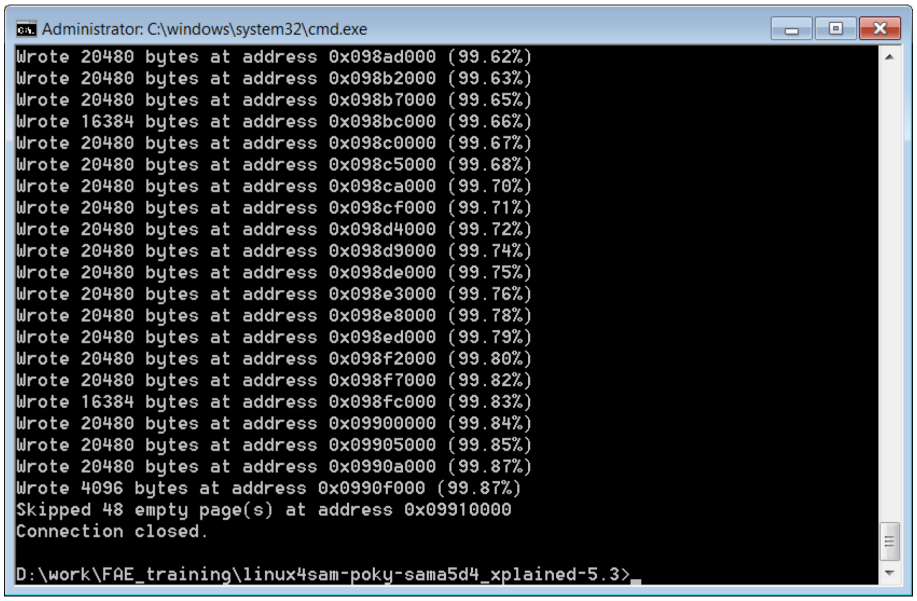
```
RomBOOT
```

- Open the JP7 (BOOT_DIS) to enable NAND Flash by enabling Flash Chip Selects.
- Copy and extract the demo package, linux4sam-poky-sama5d4_xplained-5.3.zip.
- Launch the write-nandflash-usb.bat script to flash the demo. This script will run sam-ba_3.1.2 with proper parameters.
- The following message should appear in the cmd.exe window.


Administrator: C:\windows\system32\cmd.exe - write-nandflash-usb.bat

```
Erased 262144 bytes at address 0x01c80000 (5.62%)
Erased 262144 bytes at address 0x01cc0000 (5.66%)
Erased 262144 bytes at address 0x01d00000 (5.71%)
Erased 262144 bytes at address 0x01d40000 (5.76%)
Erased 262144 bytes at address 0x01d80000 (5.81%)
Erased 262144 bytes at address 0x01dc0000 (5.86%)
Erased 262144 bytes at address 0x01e00000 (5.91%)
Erased 262144 bytes at address 0x01e40000 (5.96%)
Erased 262144 bytes at address 0x01e80000 (6.01%)
Erased 262144 bytes at address 0x01ec0000 (6.05%)
Erased 262144 bytes at address 0x01f00000 (6.10%)
Erased 262144 bytes at address 0x01f40000 (6.15%)
Erased 262144 bytes at address 0x01f80000 (6.20%)
Erased 262144 bytes at address 0x01fc0000 (6.25%)
Erased 262144 bytes at address 0x02000000 (6.30%)
Erased 262144 bytes at address 0x02040000 (6.35%)
Erased 262144 bytes at address 0x02080000 (6.40%)
Erased 262144 bytes at address 0x020c0000 (6.45%)
Erased 262144 bytes at address 0x02100000 (6.49%)
Erased 262144 bytes at address 0x02140000 (6.54%)
Erased 262144 bytes at address 0x02180000 (6.59%)
Erased 262144 bytes at address 0x021c0000 (6.64%)
Erased 262144 bytes at address 0x02200000 (6.69%)
Erased 262144 bytes at address 0x02240000 (6.74%)
```

- When the demo flash is done, the similar message should appear like this.



```
Wrote 20480 bytes at address 0x098ad000 (99.62%)
Wrote 20480 bytes at address 0x098b2000 (99.63%)
Wrote 20480 bytes at address 0x098b7000 (99.65%)
Wrote 16384 bytes at address 0x098bc000 (99.66%)
Wrote 20480 bytes at address 0x098c0000 (99.67%)
Wrote 20480 bytes at address 0x098c5000 (99.68%)
Wrote 20480 bytes at address 0x098ca000 (99.70%)
Wrote 20480 bytes at address 0x098cf000 (99.71%)
Wrote 20480 bytes at address 0x098d4000 (99.72%)
Wrote 20480 bytes at address 0x098d9000 (99.74%)
Wrote 20480 bytes at address 0x098de000 (99.75%)
Wrote 20480 bytes at address 0x098e3000 (99.76%)
Wrote 20480 bytes at address 0x098e8000 (99.78%)
Wrote 20480 bytes at address 0x098ed000 (99.79%)
Wrote 20480 bytes at address 0x098f2000 (99.80%)
Wrote 20480 bytes at address 0x098f7000 (99.82%)
Wrote 16384 bytes at address 0x098fc000 (99.83%)
Wrote 20480 bytes at address 0x09900000 (99.84%)
Wrote 20480 bytes at address 0x09905000 (99.85%)
Wrote 20480 bytes at address 0x0990a000 (99.87%)
Wrote 4096 bytes at address 0x0990f000 (99.87%)
Skipped 48 empty page(s) at address 0x09910000
Connection closed.

D:\work\FAE_training\linux4sam-poky-sama5d4_xplained-5.3>
```

- Power cycle the board.
- Look the system booting through the serial linein Terminal.

3.3 Play with the demo

Now you should have the Linux demo running on your board! You can access the Linux console through the serial line.

Use the `root` login account without password.

This hands-on document is built as a series of questions to help you discover the possibilities of the pre-installed embedded Linux environment.

INFO Commands that can be run on the target or host machines are written like the example below. The # as the first character only represents the command prompt that can be of different form: ">", "$" or "#" for instance. So, when copying the command line, simply remove this first character:

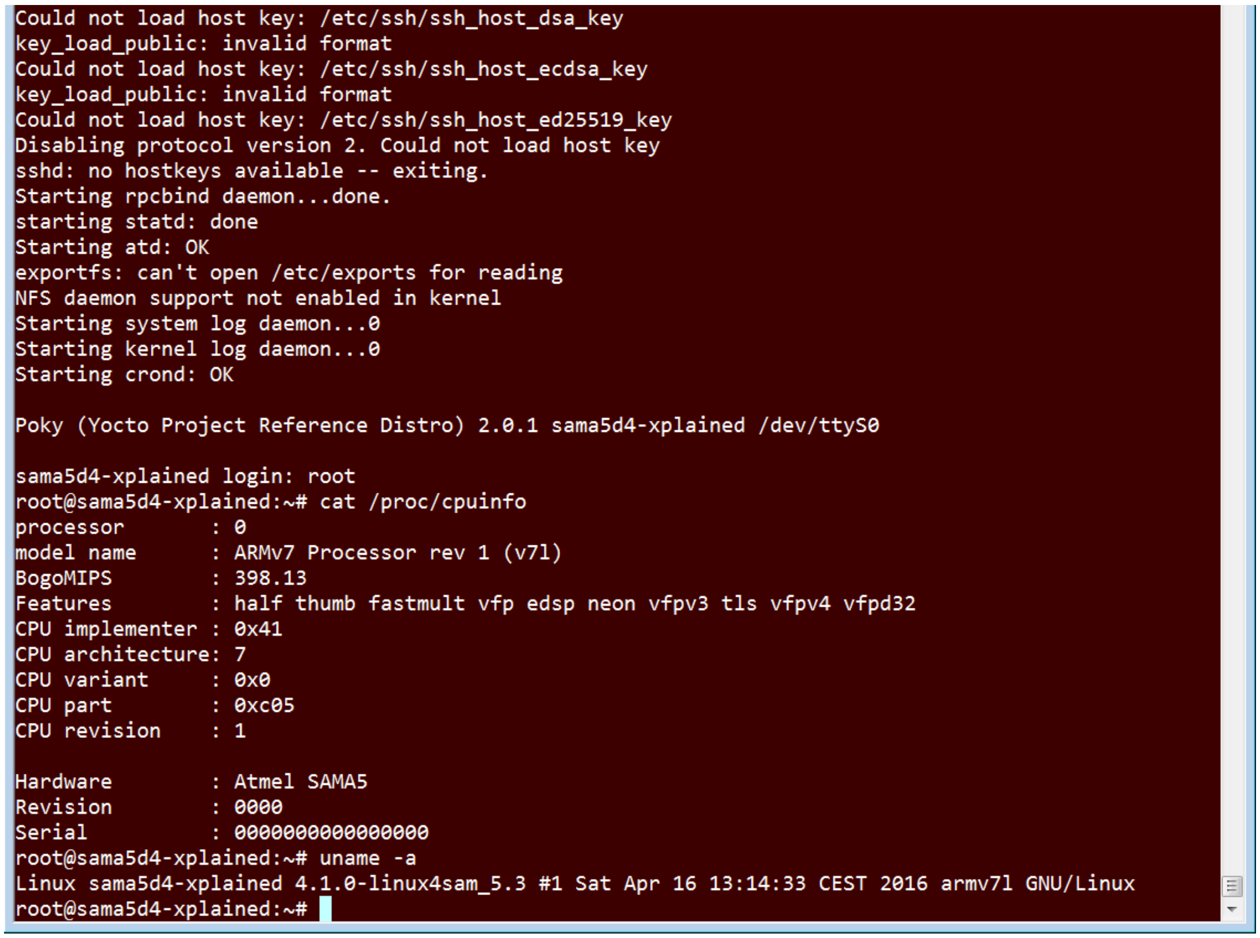
```
# command -a -b -c -d
```

TIPS The useful applications and commands that will allow you to answer these questions are described just below the question series. Usually, you can also obtain help on a specific command by simply typing it with the –h argument.

You can have a look at the boot log by using the `dmesg` command:

```
# dmesg
```

or have an overview in the image below :



```
Could not load host key: /etc/ssh/ssh_host_dsa_key
key_load_public: invalid format
Could not load host key: /etc/ssh/ssh_host_ecdsa_key
key_load_public: invalid format
Could not load host key: /etc/ssh/ssh_host_ed25519_key
Disabling protocol version 2. Could not load host key
sshd: no hostkeys available -- exiting.
Starting rpcbind daemon...done.
starting statd: done
Starting atd: OK
exportfs: can't open /etc/exports for reading
NFS daemon support not enabled in kernel
Starting system log daemon...0
Starting kernel log daemon...0
Starting crond: OK

Poky (Yocto Project Reference Distro) 2.0.1 sama5d4-xplained /dev/ttyS0

sama5d4-xplained login: root
root@sama5d4-xplained:~# cat /proc/cpuinfo
processor       : 0
model name      : ARMv7 Processor rev 1 (v7l)
BogoMIPS        : 398.13
Features        : half thumb fastmult vfp edsp neon vfpv3 tls vfpv4 vfpd32
CPU implementer : 0x41
CPU architecture: 7
CPU variant     : 0x0
CPU part        : 0xc05
CPU revision    : 1

Hardware        : Atmel SAMA5
Revision        : 0000
Serial          : 0000000000000000
root@sama5d4-xplained:~# uname -a
Linux sama5d4-xplained 4.1.0-linux4sam_5.3 #1 Sat Apr 16 13:14:33 CEST 2016 armv7l GNU/Linux
root@sama5d4-xplained:~#
```

TO DO **Answer to every question with the help of the commands described in the "*Tips*" section. Of course, the trainers will help you each time you need an advice: do not hesitate to call them.**

3.3.1 Linux system diagnostic and debug information

Once you have access to the system, it's interesting to know what's running on it and what its status is. The usual applications will help you answer the questions below:

Q1: What is Cortex-A5 revision of SAMA5D4? What is the FPU flavor used for SAMA5D4? Does the SAMA5D4 embed the NEON Advanced SIMD (Single Instruction Multiple Data) Media Processing Engine?

Q2: What is the amount of memory installed on SAMA5D4 Xplained board?

Q3: What is the amount of mass storage that is installed on SAMA5D4 Xplained board?

Q4: What is the type of filesystem used on the pre-flashed demo?

Q5: At which frequency the PIO (PIOA for instance) hardware sub-system runs (we talk about PIO IP internal frequency)?

Q6: What are the virtual and hardware interrupt numbers of the eth0 interface?

Q7: How much kernel modules are already loaded?

Q8: How much serial interfaces are available on this running system?

Q9: Which DMA channel is used by SHA hardware crypto-engine?

Q10: List the software packages installed on this Yocto Project Linux system

Q11: is the package "python-smbus" already installed on the system?

TIPS

Command to learn the ARM core characteristics:

```
# cat /proc/cpuinfo
```

`ps (ps ax)` - applications running on the system

`/proc` and `/sys, /sys/kernel/debug` - filesystems exposing information from the running kernel
this command for example prints a summary of all the clocks configured by the Linux kernel:

```
# cat /sys/kernel/debug/clk/clk_summary
```

`free`, `/proc/meminfo` - information about the memory of the system

`df, df -h` – information about the “disk” space used and available on the system

Serial interfaces can be monitored through several system information files and some usfull commands like `setserial` and `stty`. One file lists serial ports that are handled by the atmel_serial driver:

```
# cat /proc/tty/driver/atmel_serial
```

For knowing information about the **interrupts**, two files have to be checked: one that lists Linux specific "virtually numbered" IRQ lines (first column) :

```
# cat /proc/interrupts
```

The other file gives the correspondence between Linux "virtual" IRQ number and the real hardware interrupt number that you can find in the product datasheet (**Peripheral Identifiers** chapter):

```
cat /sys/kernel/debug/irq_domain_mapping
```

`lsmod` - information about kernel modules `dmesg` - useful command for exploring the system boot log and all kernel messages. It can give excellent information about the hardware resources probed of the SoC.

`mount` - without options, this command gives a status of filesystems already mounted on the running system

`opkg` - is the package management tool used in this Yocto Project system.
The `opkg list_installed` command, for example, prints the list of all installed .ipk packages on the system. For example, you can list all python packages with:

```
# opkg list_installed | grep python
```

3.3.2 Information about the Device Tree

The Device Tree is the database responsible of describing the hardware present on the platform so that the Linux kernel can find all the SoC and board peripherals. The Device Tree binary blob (compiled form of a Device Tree) is passed to the Linux kernel during the boot process by the bootloader (U-Boot is responsible for this on SAMA5D4 Xplained).

Once the system runs, it is possible to inspect the Device Tree structure by accessing the `/proc` filesystem. The whole tree is presented as directories which represent nodes and files which represent properties.

The `compatible` property of each node tells the Linux kernel which driver to load.

Q1: What is the compatible string for the ARM core that powers the SAMA5D4?

Q2: Is the LCD controller enabled on your platform?

Q3: To which compatible string the Ethernet driver matches on SAMA5D4?

Q4: Can you find the hardware IRQ number for the AES peripheral in the Device Tree description under `/proc/device-tree`?

TIPS

The topmost compatible string is used to match the entire platform:

```
# hexdump -C /proc/device-tree/compatible
```

The LCD controller can be found on a node present in the `/proc/device-tree/ahb/apb/` path.

Find if a node is enabled or disabled by checking its `status` property.

The match between what the Linux driver is able to handle and the declaration in the Device Tree is done through the **compatible string**:

```
cat /proc/device-tree/ahb/apb/ethernet\@f8020000/compatible && echo
```

The hardware interrupt number in listed in the `interrupts` property which contains 2 32 bits values: the first one is the interrupt number, the second one tells that this interrupt property is stored on a 4 bytes length variable. You'll have to use `hexdump` command to read this file.

3.3.3 Using interfaces / access to the hardware

As a fast prototyping platform, the SAMA5D4 Xplained is designed to interact with several kinds of hardware interfaces. Many tools are available to ease the access to these interfaces.

TO DO You will discover them through a list of questions that will cover the on-board hardware.

Q1: What appends when you connect a USB key to one of the USB host ports (if you have one)?

Q2: What is the proper command to gain access to the USB key content (if you have one)?

Q3: What controls the `d8` and `d10` LEDS, which *trigger* actually?

Q4: Switch off then Light up again LED `d8`. Change its trigger value so that it blinks regularly, like `d10`.

Q5: Check that the `USER` Push Button triggers an "Input" Linux event.

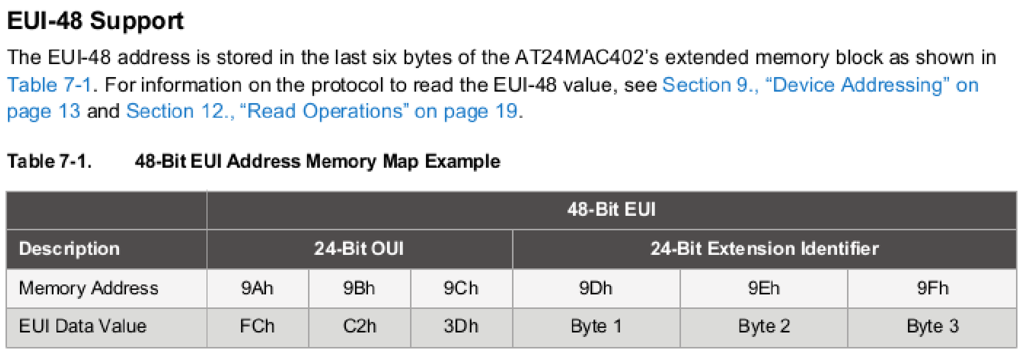
Q6: Set the date and store it in the integrated RTC clock. Reboot the system and see that the date is preserved (as long as the power is maintained, optional coin cell or VBAT provided).

Q7: How much devices are connected to the i2c bus #0? What (is) are (its) their i2c address(es)?

Q8: Run the following command from addresses 0x9a to 0x9f:

```
# i2cget -y 0 0x5c 0x9a
```

following the addressing scheme below:



EUI-48 Support

The EUI-48 address is stored in the last six bytes of the AT24MAC402's extended memory block as shown in Table 7-1. For information on the protocol to read the EUI-48 value, see Section 9., "Device Addressing" on page 13 and Section 12., "Read Operations" on page 19.

Table 7-1. 48-Bit EUI Address Memory Map Example

	48-Bit EUI					
Description	**24-Bit OUI**			**24-Bit Extension Identifier**		
Memory Address	9Ah	9Bh	9Ch	9Dh	9Eh	9Fh
EUI Data Value	FCh	C2h	3Dh	Byte 1	Byte 2	Byte 3

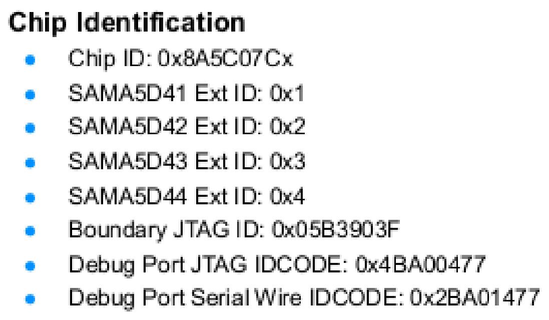
Documentation from Atmel-8807-SEEPROM-AT24MAC402-602-Datasheet.pdf.

Then compare the value observed with the HWaddr field of the `ifconfig` command:

```
# ifconfig eth0
```

What is stored in this i2c memory named AT24MAC402 present on the board?

Q9: The SAMA5D4 Chip ID Register is located at physical address 0xFC069040 and Chip ID Extension Register at 0xFC069044. With information given below, can you figure out exactly which SoC runs on the board?


Chip Identification

- Chip ID: 0x8A5C07Cx
- SAMA5D41 Ext ID: 0x1
- SAMA5D42 Ext ID: 0x2
- SAMA5D43 Ext ID: 0x3
- SAMA5D44 Ext ID: 0x4
- Boundary JTAG ID: 0x05B3903F
- Debug Port JTAG IDCODE: 0x4BA00477
- Debug Port Serial Wire IDCODE: 0x2BA01477

Documentation from the SAMA5D4 datasheet.

TIPS

`dmesg` or `cat /var/log/syslog` - allow to monitor system activity

`lsusb (lsusb -vvv)` - check USB devices connected to the system

`mount` (with arguments) - allow to attach mass-storage devices to the root filesystem and "mount" new filesystems:

```
# mount /dev/sda /mnt
```

or

```
# mount /dev/sda1 /mn
```

`/sys/class/leds/` - is the sysfs interface for controlling the Linux LED subsystem (Documentation/leds/leds-class.txt).

`hexdump` and `/dev/input/event0` are useful for knowing that the Linux kernel "input" sub-system is working correctly. Check also the handy following command:

```
# evtest
```

`date, hwclock` are the tools used for setting the date and storing it to the SAMA5D4 RTC:

```
root@sama5d4-xplained-sd:~# date -s "20160610 16:11"
Fri Jun 10 16:11:00 UTC 2016
root@sama5d4-xplained-sd:~# hwclock --systohc
root@sama5d4-xplained-sd:~# date && hwclock
Fri Jun 10 16:11:25 UTC 2016
Fri Jun 10 16:11:25 2016  0.000000 seconds
```

`i2cdetect` and all `i2c-tools` like `i2cget` and `i2cset` - probe and use i2c devices from user space

`devmem2` command can read or write any physical address in a Linux system:

```
# devmem2 0xfc069040
# devmem2 0xfc069044
```

By running the following commands, you can confirm that you've just found the right SAMA5D4 variant of the family:

```
# dmesg | grep Detected
[    0.090000] AT91: Detected SoC family: sama5d4
[    0.090000] AT91: Detected SoC: sama5d44, revision 0
```

3.3.4 Development tools

On such a fast prototyping board, it is important to have all tools at hand to getting started quickly. Development tools are a pretty important topic in embedded world and particularly in embedded Linux. We can classify these tools in several categories:

- the classical cross-compilation toolchain.
- the native C language compiler. This compilation process shall be performed on the board itself or on an equivalent platform;
- the use of scripting languages like shell scripting or the more elaborated python scripting language.

TO DO **During this hands-on, you will experiment with the last of these possibilities. It's the more practical way of developing useful code on the SAMA5D4 Xplained board as the code is directly written and instantly tested on-board itself.**

Q1: Create a bash script that reads and prints brightness of the `d10` LED every 0.5 second

Q2: print "Hello world" with python

Q3: write a little python script which prints "Hello world"

Q3: run the following python script. Which modules does it need? What does it do?

```
#!/usr/bin/env python

import smbus

bus = smbus.SMBus(0)    # 0 = /dev/i2c-0 (port I2C0), 1 = /dev/i2c-1 (port I2C1)

DEVICE_ADDRESS = 0x5c
DEVICE_REG_EUI = 0x9a

addr = []

for i in range (DEVICE_REG_EUI, DEVICE_REG_EUI + 6):
     var = bus.read_byte_data(DEVICE_ADDRESS, i)
     addr.append(var)
     print("%02x" % var)

print("\nNow print Ethernet address:")
print ":".join(["%02x" % i for i in addr])
```

TIPS

`bash` - "`while true; do <something>; done`" is a infinite loop in bash

`bash` - you can read a file with the `cat` command:

```
# cat /sys/class/leds/d10/brightness
```

`bash` - `sleep` is a simple command to wait for a moment

`vi` - the ubiquitous text editor

You can run `python` on the command line and directly jump into the interpreter: an easy way to test little python commands.

a python script begins with `#!/usr/bin/env python`

`chmod` (`chmod +x <my_file>`) - a command that modifies a file access rights (can add the "executable" permission to a file)

Atmel | Enabling Unlimited Possibilities®

Atmel Corporation
1600 Technology Drive
San Jose, CA 95110
USA
Tel: (+1)(408) 441-0311
Fax: (+1)(408) 487-2600
www.atmel.com

Atmel Asia Limited
Unit 01-5 & 16, 19F
BEA Tower, Millennium City 5
418 Kwun Tong Road
Kwun Tong, Kowloon
HONG KONG
Tel: (+852) 2245-6100
Fax: (+852) 2722-1369

Atmel Munich GmbH
Business Campus
Parkring 4
D-85748 Garching b. Munich
GERMANY
Tel: (+49) 89-31970-0
Fax: (+49) 89-3194621

Atmel Japan G.K.
16F Shin-Osaki Kangyo Bldg.
1-6-4 Osaki, Shinagawa-ku
Tokyo 141-0032
JAPAN
Tel: (+81)(3) 6417-0300
Fax: (+81)(3) 6417-0370