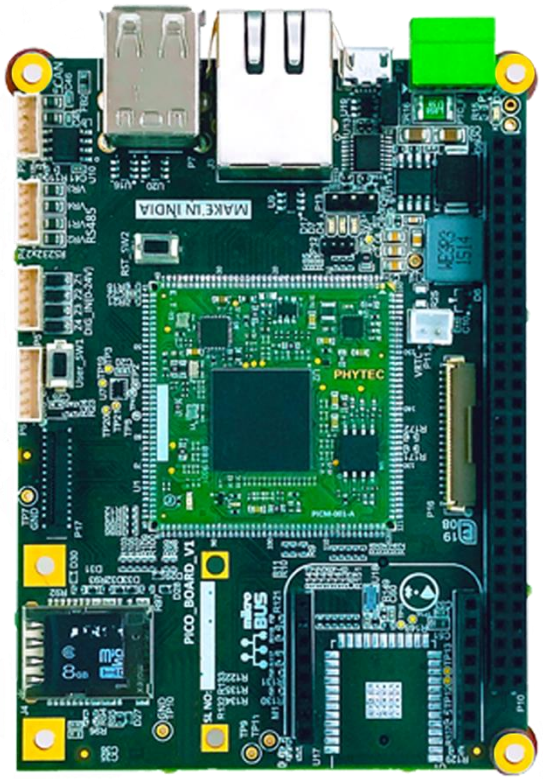
Embedded Linux Porting (C8)

Linux Kernel

Agenda

- Linux Kernel ARCH
- Kernel Dir Structure
- Kernel Layers H/W dependent (BSP) and independent
- Kernel Build System (KConfig)

RuggedBOARD



- A5D2x @500MHz
- Cortex - A5
- 64MB Ram
- 32MB Flash
- 2 x RS232
- 1x RS485
- 1 x CAN
- 1 x Ethernet
- TFT & Cap Touch
- 1 x MicroSD Slot
- 2 x USB
- DC & USB Power
- Expansion Header
- mikroBUS Conn.
- mPCIe conn.
- Micro Sim Slot

open source hardware

open source initiative

Industrial Grade Hardware for IIoT

https://Community.ruggedboard.com

Kernel Source

Browse Source: https://github.com/rugged-board/linux-rba5d2x.git

Download U-Boot for RuggedBOARD

$ wget https://github.com/rugged-board/linux-rba5d2x/archive/linux-rba5d2x.zip

Or

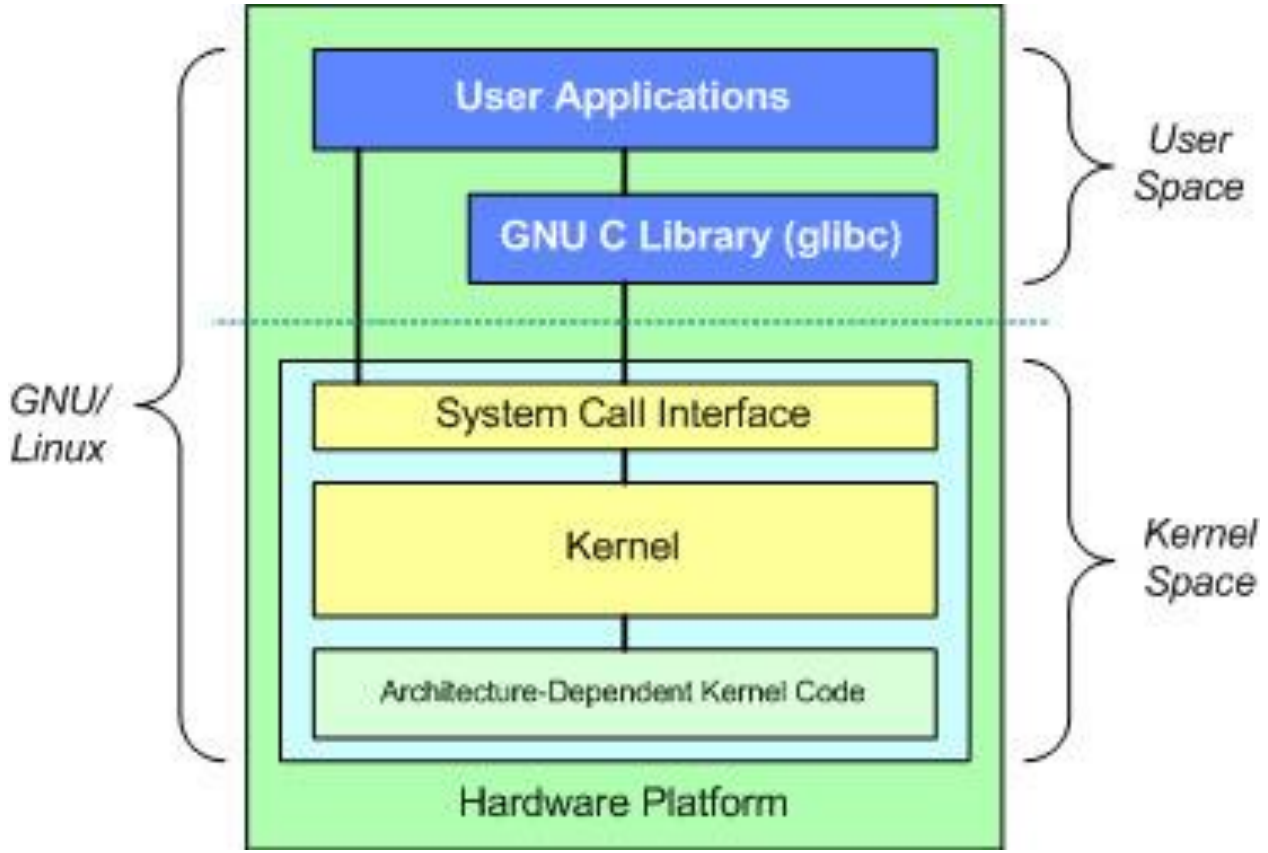
$ git clone https://github.com/rugged-board/linux-rba5d2x.git

Kernel ARCH

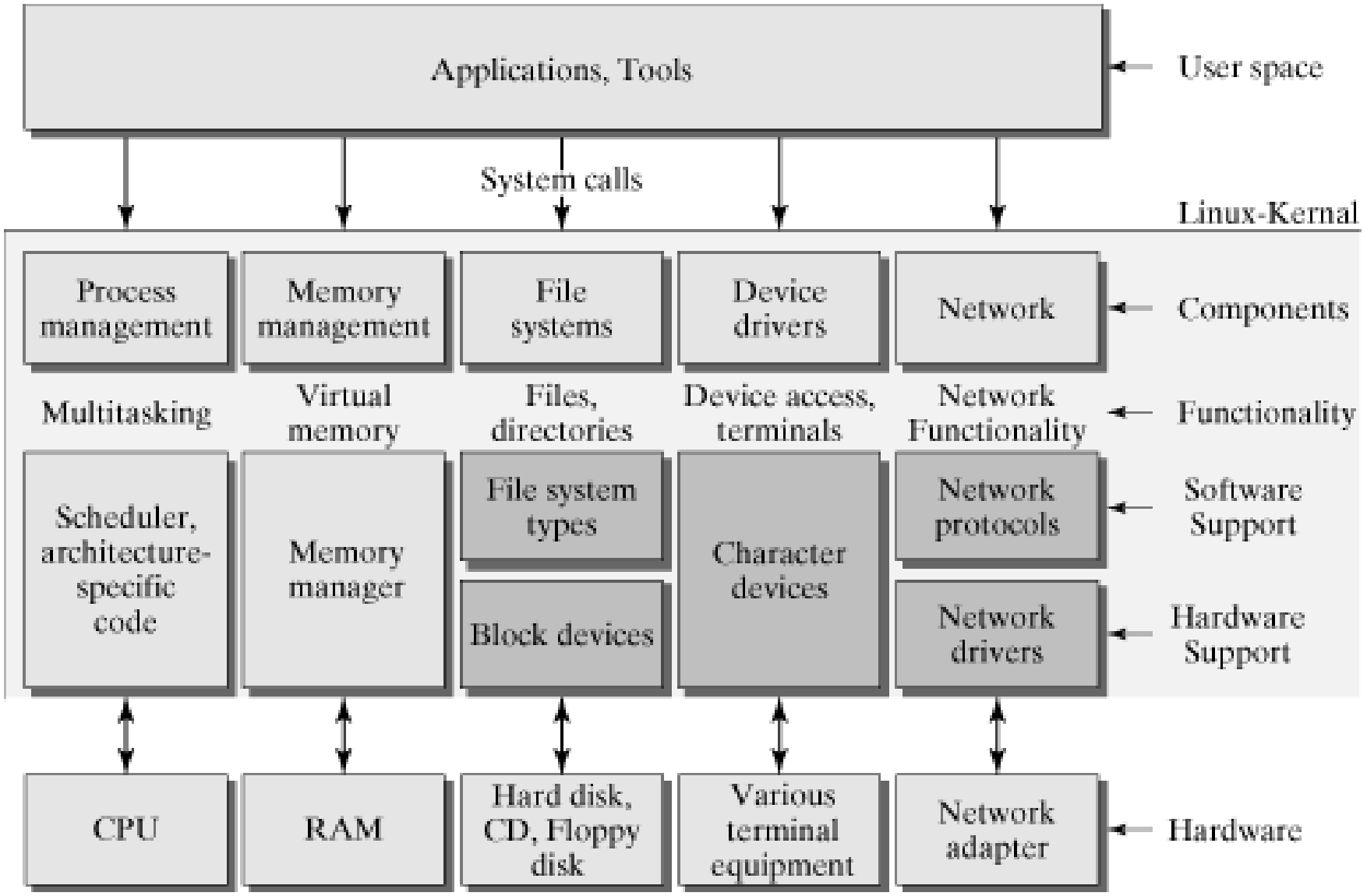
OS Components

- Scheduler
- Process Mgmt
- Memory Mgmt
- Device Mgmt

Linux Kernel Layout



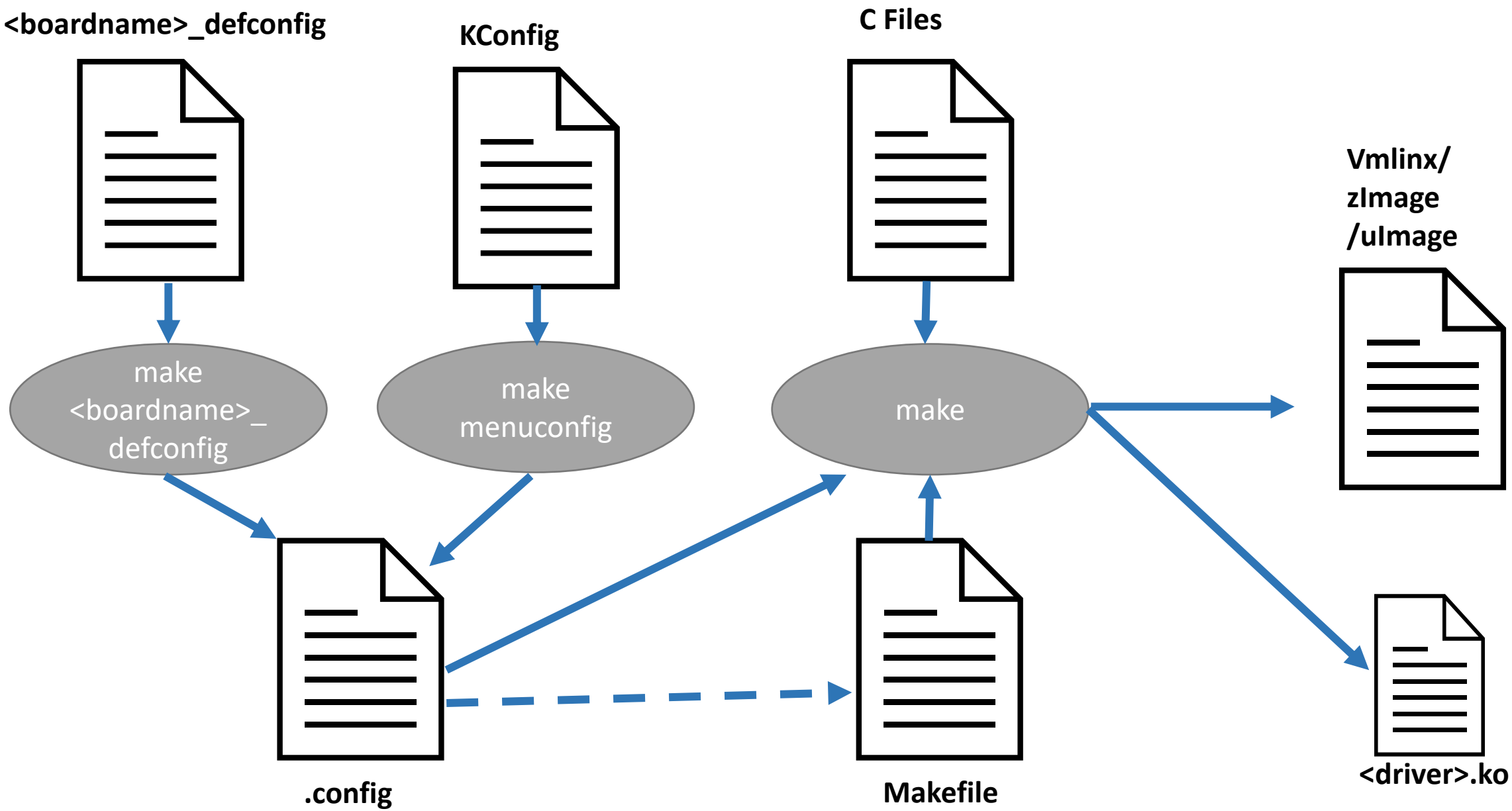
Kernel ARCH


Applications, Tools
User space
System calls
Linux-Kernal
Process management
Memory management
File systems
Device drivers
Network
Components
Multitasking
Virtual memory
Files, directories
Device access, terminals
Network Functionality
Functionality
Scheduler, architecture-specific code
Memory manager
File system types
Block devices
Character devices
Network protocols
Network drivers
Software Support
Hardware Support
CPU
RAM
Hard disk, CD, Floppy disk
Various terminal equipment
Network adapter
Hardware

Kernel Dir Structure

linux/arch/arm/kernel	Arch & Core specific code
linux/arch/arm/plat-<soc>	SOC specific code
Linux/arch/arm/mach-<soc>	Board directory contains board files
linux/arch/arm/boot/dts	device tree directory consists of device tree files for SOC, SOM, SBC *sama5d2.dsi , rb_a5d2x.dtsi, rugged_board_a5d2x.dts*
linux/arch/arm/boot	Linux Kernel Boot-strap
linux/arch/arm/configs	Contains board default configuration file used to configure kernel for a specific board. <boardname_defconfig> for ruggedboard we have rb_a5d2x_defconfig
linux/drivers	Contains bus drivers & device drivers (gpio, serial, i2c, spi, mmc, usb, net) Device Driver: rtc/ds1307.c, misc/i2c_eeprom.c …
linux/kernel	Linux Kernel core logic H/W Independent
linux/init	Kernel init code main.c (Start point of Kernel C code)
linux/fs	File System Supported by linux kernel

Kernel Kconfig



Compiling Linxu Kernel

Developer Wiki Page link here …

#1 Download the source
#2 Set the toolchain
#3 Clean the source only for the first time
$make distclean

#4 Configure the kernel source for ruggedboard-a5d2x
$make rb_a5d2x_defconfig

#5 Do additional configuration if required using menuconfig
$make menuconfig

#6 Compile the Kernel code
$make

Kernel Experiments

1. Delete RootFS Image from u-boot and boot the board (make note of the error log)
2. Delete Kernel Image from u-boot and boot the board (make note of the error log)
3. Do Kernel banner modifications, compile & test the new kernel Image
4. Add sled driver under driver/misc folder and test the kernel

Open Discussions

Connect-us