

Debugging Embedded Linux Systems: Linux/Kernel Overview

Debugging Embedded Linux Training Series [Part 1]

Agenda

- Linux Kernel Debugging Training Series overview
- Linux/kernel overview
- For more information

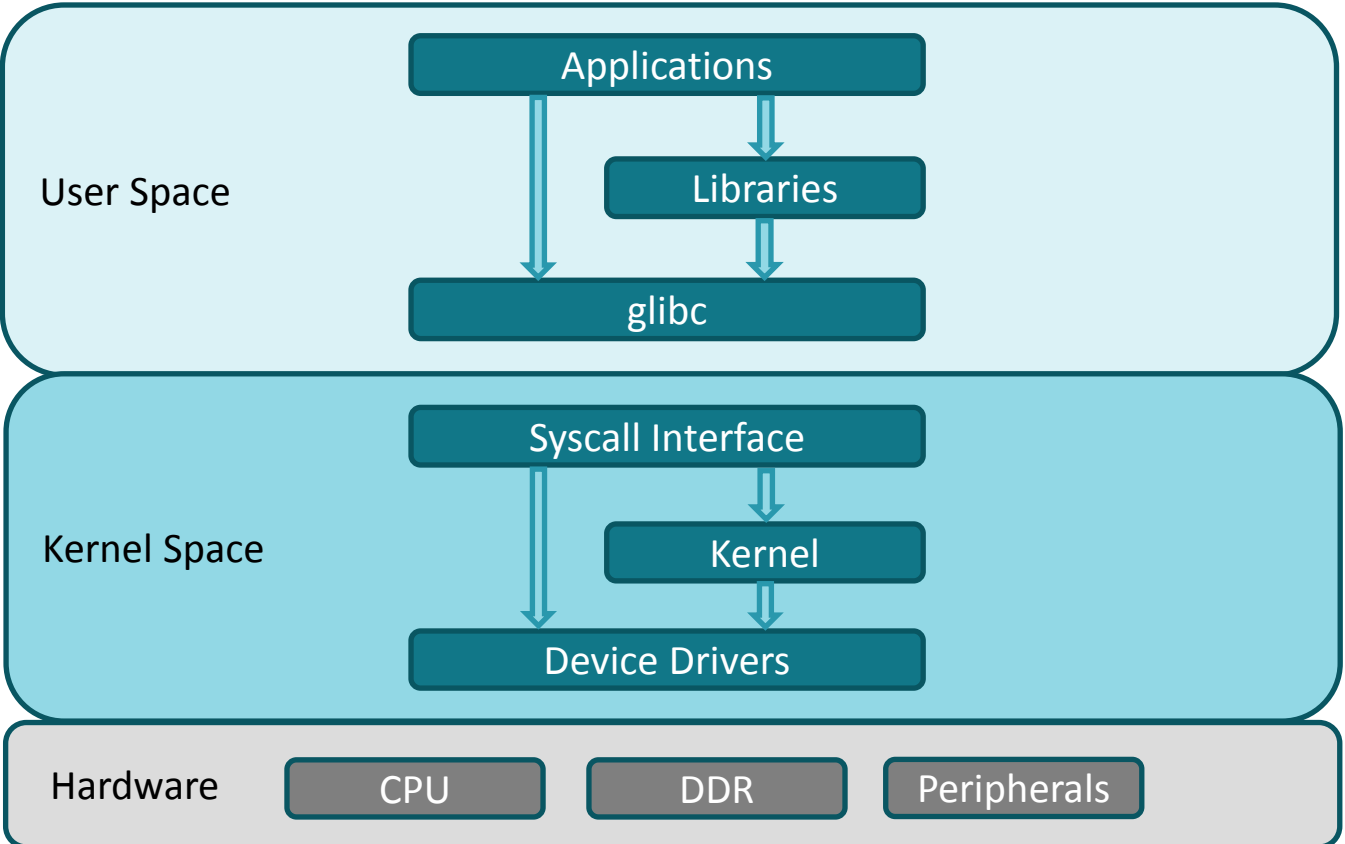
Debugging embedded Linux training summary

- **Purpose:** Linux is well-adopted within embedded systems. But debugging Linux system issues can be overwhelming. This training series teaches the techniques of debugging kernel issues that may be encountered in embedded Linux systems.
- **Goal:** Learn how to capture proper kernel logs for debugging issues.
- **Scope:**
 - Explain Linux Kernel logging system and logging API.
 - Illustrate how to locate a particular device driver.
 - Demonstrate how to read kernel Oops logs.

Debugging Embedded Linux Training Series

- **Part 1: Linux/Kernel Overview**
- Part 2: Kernel Logging System Overview
- Part 3: printk and Variations
- Part 4: Dynamic Debug
- Part 5: Locate Device Driver Source Code
- Part 6: Understand Kernel Oops Logs

Linux overview



Kernel config

- Kernel is configurable
- .config
- make menuconfig

make menuconfig

```
.config - Linux/arm 4.4.41 Kernel Configuration

Linux/arm 4.4.41 Kernel Configuration
Arrow keys navigate the menu. <Enter> selects submenus ---> (or empty
submenus ----). Highlighted letters are hotkeys. Pressing <Y> includes, <N>
excludes, <M> modularizes features. Press <Esc><Esc> to exit, <?> for Help,
</> for Search. Legend: [*] built-in [ ] excluded <M> module < > module
+-----+
| -- Patch physical to virtual translations at runtime |
| General setup ---> |
| [*] Enable loadable module support ---> |
| [*] Enable the block layer ---> |
| System Type ---> |
| Bus support ---> |
| Kernel Features ---> |
| Boot options ---> |
| CPU Power Management ---> |
| Floating point emulation ---> |
| Userspace binary formats ---> |
| Power management options ---> |
| [*] Networking support ---> |
| Device Drivers ---> |
| Firmware Drivers ---> |
| File systems ---> |
| Kernel hacking ---> |
| Security options ---> |
| -- Cryptographic API ----> |
| Library routines ---> |
| [ ] Virtualization ----> |
+-----+
| <Select> < Exit > < Help > < Save > < Load > |
+-----+
```

.config example

```
#  
# Automatically generated file; DO NOT EDIT.  
# Linux/arm 4.4.41 Kernel Configuration  
#
```

```
CONFIG_ARM=y
```

```
CONFIG_INIT_ENV_ARG_LIMIT=32
```

```
CONFIG_CROSS_COMPILE=""
```

```
# CONFIG_COMPILE_TEST is not set
```

```
CONFIG_LOCALVERSION=""
```

```
# CONFIG_IRQ_DOMAIN_DEBUG is not set
```

```
CONFIG_LOG_CPU_MAX_BUF_SHIFT=12
```

```
CONFIG_XFRM=y
```

```
CONFIG_XFRM_ALGO=m
```

```
CONFIG_XFRM_USER=m
```

```
# CONFIG_XFRM_SUB_POLICY is not set
```


For more information

- Processor SDK Training Series:
<http://training.ti.com/processor-sdk-training-series>
- Debugging Embedded Linux Training Series:
<http://training.ti.com/debug-embedded-linux-training-series>
- Processor SDK Linux Getting Started Guide:
http://processors.wiki.ti.com/index.php/Processor_SDK_Linux_Getting_Started_Guide
- Download Processor SDK Linux for Embedded Processors:
<http://www.ti.com/processorsdk>
- For questions about this training, refer to the E2E Embedded Linux Community Forum: <http://e2e.ti.com/support/embedded/linux/f/354>



©Copyright 2017 Texas Instruments Incorporated. All rights reserved.

This material is provided strictly “as-is,” for informational purposes only, and without any warranty.
Use of this material is subject to TI’s **Terms of Use**, viewable at TI.com