Lecture 27: Smartphone Architecture and Mobile OS (Android)

Mobile Device

DI05011011: Mobile Communication

Agenda for Today

- 1. Introduction to Smartphone Architecture
- 2. System on Chip (SoC) Concept
- 3. Core Components of SoC
- 4. Modems and Connectivity in SoC
- 5. Smartphone Sensors Overview
- 6. Common Sensors (Motion & Position)
- 7. Environmental Sensors
- 8. Sensor Fusion Example
- 9. Introduction to Mobile OS
- 10. What is Android?
- 11. Android Architecture: Linux Kernel
- 12. Android Architecture: Middleware & Runtime
- 13. Android Architecture: Application Framework
- 14. Summary
- 15. Preview

Smartphone Architecture Overview

- A smartphone is fundamentally a highly integrated, battery-powered, miniaturized computer.
- **Key Modules:**
- 1. **Processing & Control:** System on Chip (SoC)
- 2. **Memory:** RAM (Volatile) and Storage (NAND Flash, UFS)
- 3. **Power Management:** PMIC (Power Management IC) and Battery
- 4. **I/O and Sensors:** Display, touch, cameras, audio, and environmental sensors.
- 5. **Connectivity:** Cellular modem, Wi-Fi, Bluetooth, NFC, GPS.

System on Chip (SoC) Concept

- An **SoC** integrates almost all critical components of a computer or electronic system into a single silicon chip.
- In a smartphone, it replaces the traditional CPU, GPU, chipset, and modem found on separate chips in a PC.
- **Why SoC?**
- - Space saving (critical for mobile devices).
- - Lower power consumption.
- - Faster communication between integrated components.
- Examples: Qualcomm Snapdragon, Apple A-Series, Samsung Exynos, MediaTek Dimensity.

Core Components of an SoC

- **CPU (Central Processing Unit):** General-purpose processing, runs the OS and apps (usually ARM architecture).
- **GPU (Graphics Processing Unit):** Handles UI rendering, gaming, and video playback.
- **NPU (Neural Processing Unit):** Specialized for AI/Machine Learning tasks (e.g., face recognition, computational photography).
- **ISP (Image Signal Processor):** Processes raw data from the camera sensors into JPEG/HEIC images.
- **DSP (Digital Signal Processor):** Handles audio processing and sensor data efficiently.

Modems and Connectivity in SoC

- Modern SoCs integrate the cellular modem directly into the silicon to save power.
- **Baseband Processor/Modem:** Handles all radio functions, modulation, demodulation, and encoding for 4G/5G.
- **Wi-Fi/Bluetooth Subsystem:** Manages local wireless connections.
- **Location Subsystem:** Integrates GPS, GLONASS, Galileo for positioning.
- Integrating the modem requires complex shielding to prevent interference with the CPU/GPU.

Smartphone Sensors Overview

- Sensors provide the smartphone with awareness of its physical environment.
- They are micro-electromechanical systems (MEMS), incredibly small mechanical devices built on silicon chips.
- They operate via the I2C or SPI interface to communicate with the SoC's sensor hub or DSP.
- Without sensors, a smartphone cannot adapt to user context (orientation, brightness, movement).

Common Sensors (Motion & Position)

- **Accelerometer:** Measures linear acceleration (including gravity) across 3 axes (X, Y, Z). Used for step counting and basic tilt detection.
- **Gyroscope:** Measures angular rotational velocity. Used for precise motion tracking, VR, and camera stabilization (OIS).
- **Magnetometer (Compass):** Measures magnetic fields to determine North. Used for maps and navigation.

Environmental & User Sensors

- **Proximity Sensor:** Emits infrared light to detect objects near the screen. Used to turn off the display during calls to prevent accidental touches.
- **Ambient Light Sensor (ALS):** Measures environmental light levels. Used to auto-adjust screen brightness.
- **Barometer:** Measures atmospheric pressure. Helps determine altitude for fitness tracking and faster GPS locks.
- **Biometric Sensors:** Fingerprint scanners (optical/ultrasonic) and 3D facial recognition (IR dot projectors).

Sensor Fusion Example: Auto-Rotate

- How does a phone know to switch from Portrait to Landscape?
- 1. **Accelerometer** detects a change in the direction of gravity across its X/Y axes.
- 2. **Gyroscope** confirms a rotational movement occurred.
- 3. **Sensor Hub (DSP)** combines this data ('Sensor Fusion') to confidently conclude the phone was rotated.
- 4. DSP interrupts the **CPU**, which tells the **OS** to redraw the UI in landscape mode via the **GPU**.

Introduction to Mobile Operating Systems

- A Mobile OS is software that manages mobile hardware resources and provides common services for mobile applications.
- It acts as a bridge between the user/apps and the complex SoC hardware.
- **Key constraints for Mobile OS:**
- - Strict power management (Battery limits).
- - Limited RAM compared to PCs.
- - Touch-based user interfaces.
- - Security and app sandboxing.

What is Android?

- Android is an open-source mobile operating system developed by Google (via the Open Handset Alliance).
- It is based on a modified version of the Linux kernel.
- It uses a software stack architecture to isolate different levels of functionality.
- Applications are primarily written in Java or Kotlin.
- Dominates the global smartphone market share due to its open nature.

Android Architecture: Kernel & HAL

- **1. Linux Kernel (The Foundation):**
- - Handles core system services: memory management, process management, network stack, and drivers.
- - Provides power management (wakelocks).
- **2. Hardware Abstraction Layer (HAL):**
- - Defines standard interfaces for hardware vendors to implement.
- - Allows the higher-level OS to interact with hardware (camera, bluetooth, audio) without knowing the low-level driver specifics.

Android Architecture: Runtime & Libraries

- **3. Android Runtime (ART):**
- - The engine that runs Android applications.
- - Compiles Java/Kotlin bytecode into native machine code (Ahead-Of-Time compilation).
- - Includes Core Libraries providing Java API interfaces.
- **4. Native C/C++ Libraries:**
- - Core libraries used by the OS for heavy lifting.
- - Includes WebKit/Blink (browser engine), OpenGL/Vulkan (graphics), and SQLite (database).

Android Architecture: Framework & Apps

- **5. Application Framework:**
- - The API block that developers use to write apps.
- - Includes Activity Manager (lifecycle), Window Manager (UI), Location Manager, and Notification Manager.
- **6. System and User Apps:**
- - The top layer that users interact with.
- - Includes pre-installed apps (Dialer, SMS, Settings) and third-party apps downloaded from the Play Store.

Summary

- An SoC integrates CPU, GPU, NPU, ISP, and Modems into a single power-efficient chip.
- MEMS sensors (Accelerometer, Gyroscope, Proximity) provide environmental context.
- Sensor Fusion combines data for accurate tracking.
- Android is a Linux-based OS designed for power efficiency and touch interfaces.
- Its architecture relies on a stack: Kernel -¿ HAL -¿ Runtime/Libraries -¿ Framework -¿ Apps.

Next Lecture Preview

- In the next lecture, we will discuss a critical safety aspect of mobile communication:
- **1. Radiation Hazards:** Understanding electromagnetic radiation.
- **2. Ionizing vs. Non-Ionizing Radiation.**
- **3. Specific Absorption Rate (SAR):** Definition and basics.