Phd. lecturer Liviu Octavian Mafteiu-Scai

West University of Timisoara, FMI, CS Department

Course I

0. About this course

1. A Very Short History
2. Types of mobile devices
3. Hardware structure
4. Android Operating System.

The goal of this course:

Development students’ abilities/skills to:

understand,

design

and

develop

mobile apps

for Android with Android Studio

7 courses and 14 labs

->

too little time to become an expert

but

the basic concepts presented
and your activity in labs
could be a good start in this field

Topics :

- Hardware structure.
- Android Operating System.
- Programming languages and SDKs
- Types of mobile apps
- Mobile Apps Development – UI and UX
- Databases
- Augmented reality
- Animation and Game development
- Mobile WEB apps and Cloud
- M-learning
- Mobile Apps Testing
- Security in mobile devices and mobile apps

In labs:

small mobile apps in Android Studio.

In our labs' computers, Android Studio is installed, but could be problems with data transfer ((home-computer ⇔ lab-computer) due to different AS versions

=>

If you can, use your own PC

How to pass the exam:

work during the labs for small tasks/apps->points

work at home -> your homework project

40 % your work in labs

30% your homework project

30% written test (session)

=> you can pass the discipline during the semester

Let’s start

Mobile Device (MD)

a portable electronic equipment (must have a battery) that can be connected to the internet.

nowadays,

MD (from the user's point of view)

- **portable hand-held device;**
- **has a display screen;**
- **has an input module (touch and/or a keyboard);**
- **may provide users with telephony capabilities, Internet navigation and running some (many) apps**

(especially games!!!!!!!!!!!!)

Why a little history ?

who doesn’t know history,

doesn’t understand the present

and

cannot anticipate the future !

past

present

the near future

A little history: from telegraph to smartphone

Alessandro Volta, 1800, voltaic pile, first *electrical battery* that could continuously provide an electrical current to a circuit.

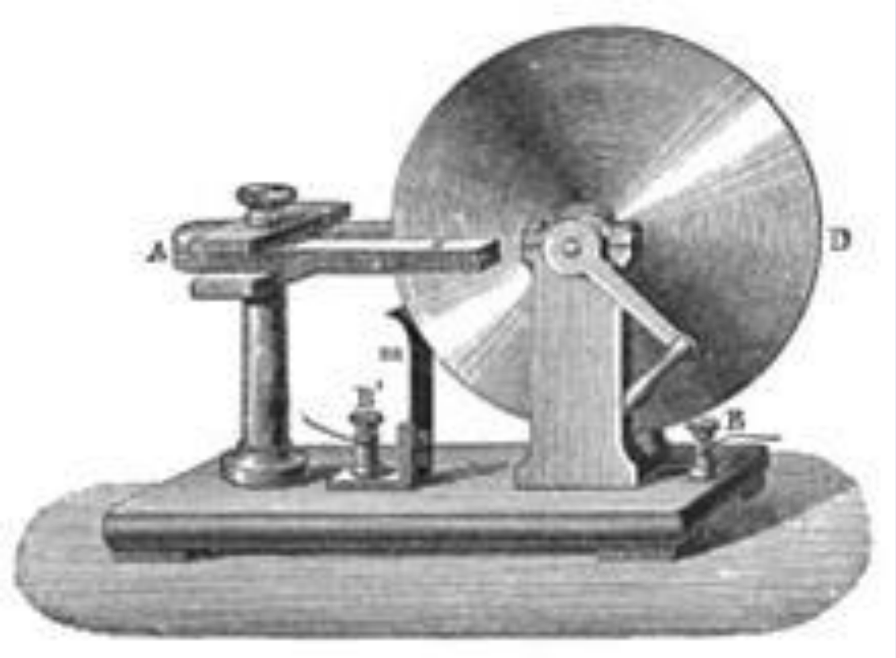

Alessandro Volta
(1745-1827)

The Volta's voltaic pile

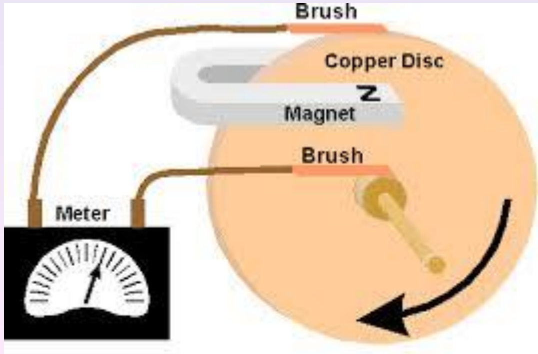
A little history: from telegraph to smartphone

Michael Faraday, 1831, *electric dynamo* which can generate electric power


Michael Faraday (1791-1867)



Electric-dinamo: first electric generator



Symbolic illustration of the Faraday Dynamo

Communications

one engine

of

human evolution

A little history: from telegraph to smartphone

William Fothergill Cooke and Charles Wheatstone, in 1837, patented the first commercial electrical **telegraph**

The first comercial telegraph

A Morse key

A little history: from telegraph to smartphone

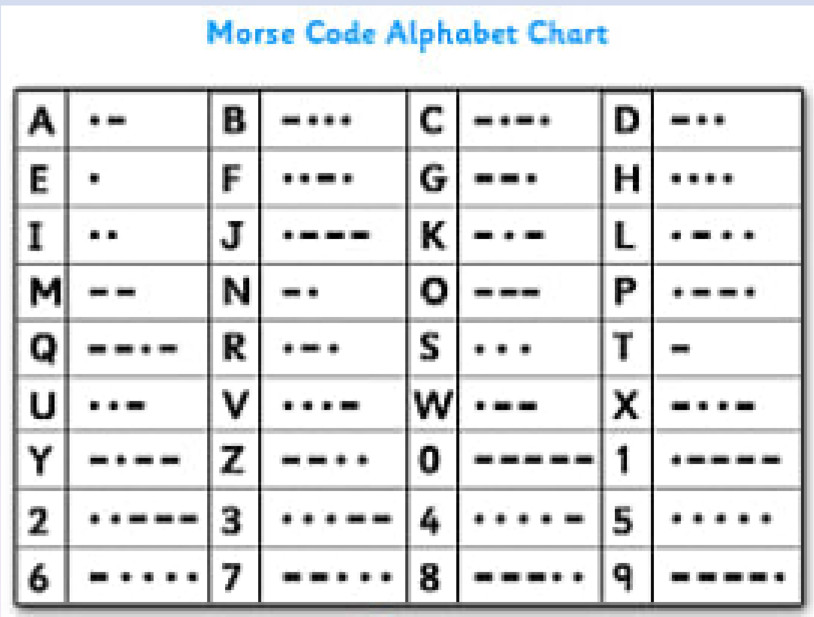
?

Morse code

?

A little history: from telegraph to smartphone

Samuel **Morse** developed a **code** (bearing his name) (1840s) that assigned a set of dots and dashes to each letter of the English alphabet



Morse Code Alphabet Chart

A	.-	B	-...	C	-.-.	D	-..
E	.	F	..-.	G	--.	H	....
I	..	J	.---	K	-.-	L	.-..
M	--	N	-.	O	---	P	.--.
Q	--.-	R	.-.	S	...	T	-
U	..-	V	...-	W	.--	X	-..-
Y	-.--	Z	--..	0	-----	1	.----
2	..---	3	...--	4	....-	5	.....
6	-....	7	--...	8	---..	9	----.

Hello World ⬄ **.... . .-.. .-.. --- .-- --- .-. .-.. -..**

The code is still used. Along the time, it saved many human lifes. You can learn it.

A little history: from telegraph to smartphone

1876: Alexander Graham Bell -> the first telephone

1877: *Bell Telephone Company*

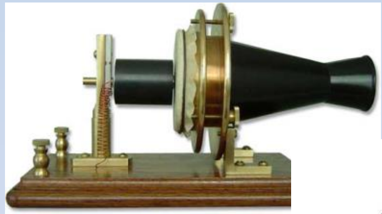
=>

1886: more than 150000 people had phones (local calls in US)

1915: the first intercontinental phone call

The Bell Telephone Memorial

A little history: from telegraph to smartphone






some old phones

A little history: from telegraph to smartphone

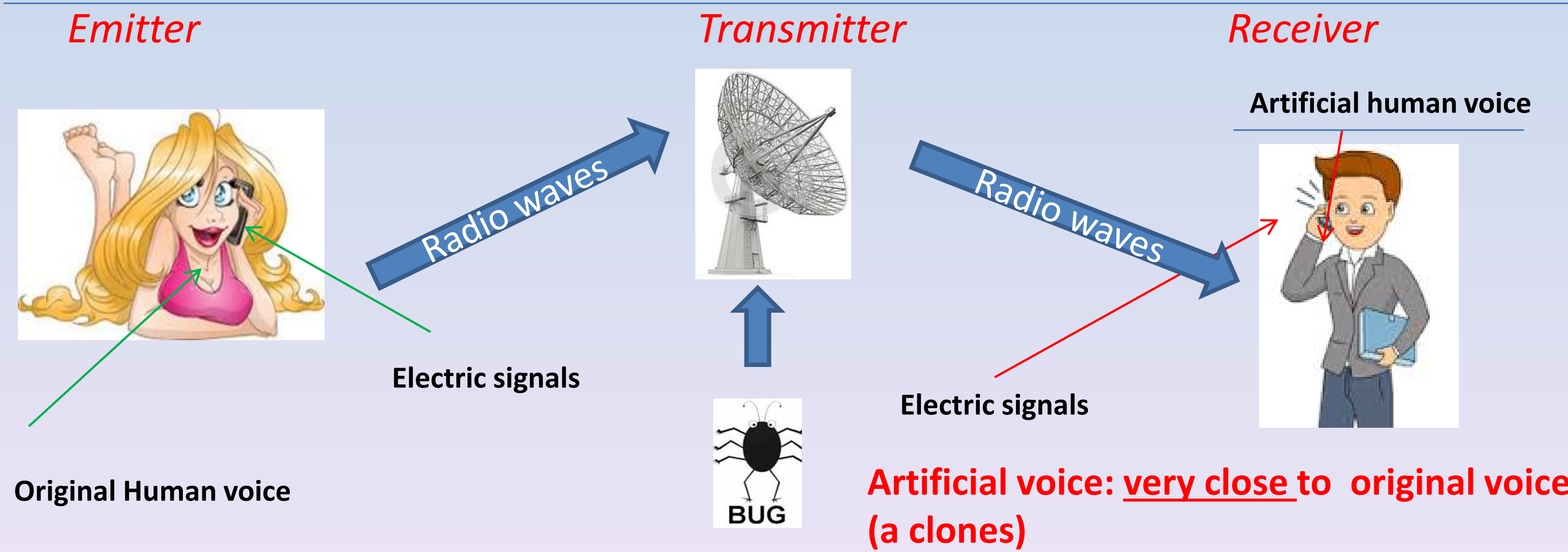
?

telephone

?

A little history: from telegraph to smartphone

Definition: A (tele)phone is a device that converts the human voice, into electric signals suitable for transmission via cables or other transmission media over long distances.



Note: in every phone there is a double converter: sound waves <-> electrical signals

The beginning of mobile telephony

- 1926, Germany, phone connections in trains: link between Hamburg and Berlin. *This service was only to 1st class travelers (very expensive price)*

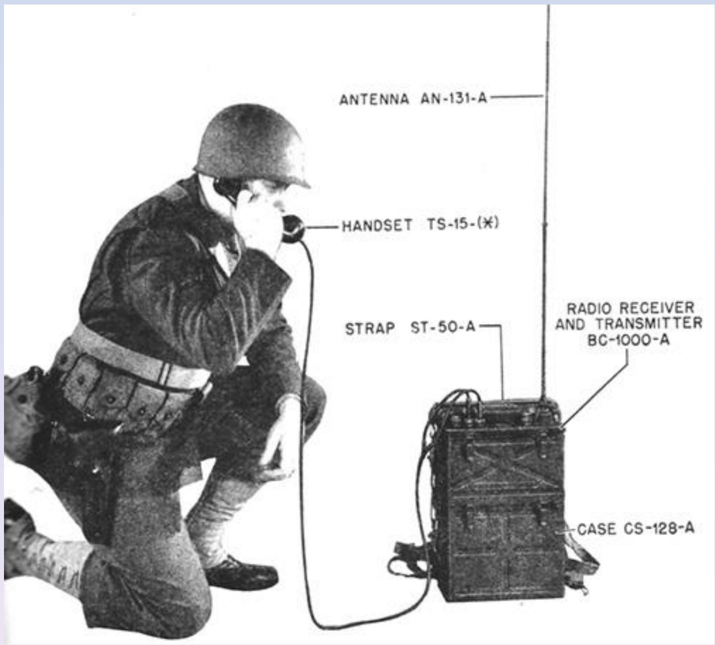
train's railroad worked as antenna

about 300 Km

- **1948: the earliest fictional descriptions of a mobile phone: a science fiction novel *Space Cadet* by Robert Heinlein**.

The beginnings of mobile telephony

In the Second World War, US army used wireless/radio telephony connections



SCR 320 or “walkIE & IR-talkIE & IR”
- made by **Motorola** in 1940
- about 17Kg
- 4,8 Km range

SCR 536 (called “handy-talkIE & IR” at first)
-made by Motorola in 1942
-about 5kg
-a variant for aircraft

The beginnings of mobile telephony

- after Second World War, wireless telephones for automobiles (1946)

- 36 kg
- Only 3 channels available in urban area

? “only 3 channell” ?

A little history: from telegraph to smartphone

The beginnings of mobile telephony

1961-USSR (Union of Soviet Socialist Republics, 1922–1991)

ALTAI

- 70 g weight
- could fit on a palm.
- about 30 km radius
- with batteries

A little history: from telegraph to smartphone

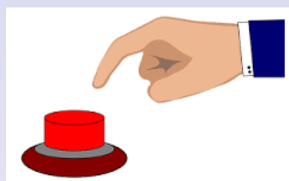
A classification of mobile phones

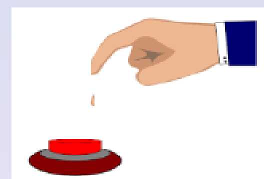
ZERO generation (0G)

1949: MTS (Mobile Telephone Service), services provided by AT & T Company.

Now, those services seem primitive because:

- the user had to press a button on the handset to talk and then release the button to listen.





- only three channels available

A little history: from telegraph to smartphone

ZERO generation (0G)

1964: MTS was replaced by Improved Mobile Telephone Service (IMTS).

- had more channels.
- in general, these “mobile phones” were as briefcase and were usually mounted in cars or trucks

A little history: from telegraph to smartphone

First generation (1G)

The first **semi-smartphone**: ***Nokia Mobira Senator***

- was released in 1982
- **international calls** can be made.
- about 10 kg

prohibited for girls

permitted only for boys with strong muscles

Second generation (2G)

Motorola International 3200

Finally, it is **handheld** !

Now we call it "a brick"

1989: the first ***pocket phone*,** **Motorola MicroTAC**

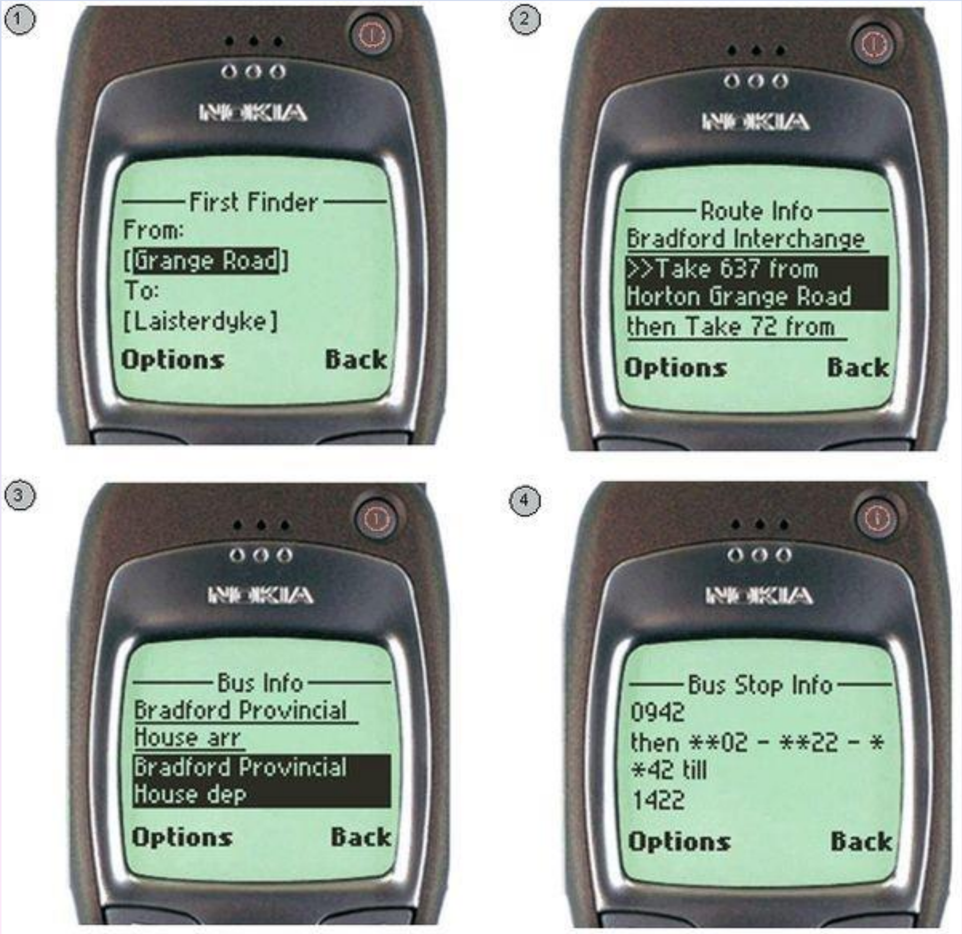
A little history: from telegraph to smartphone

1998: The first cell phone **without external antenna, Nokia 8810**

A little history: from telegraph to smartphone

1999: the first mobile phone which incorporate WAP (Wireless Application Protocol) a rudimentary **web access** (only text mode) but a revolutionary step for the mobile **Internet**.

Nokia 7110



A little history: from telegraph to smartphone

1999: The first mobile phone that incorporate an integrated browser (GeoSentric) was **Benefon Esc**.

*A primitive **GPS** and Infrared port*.

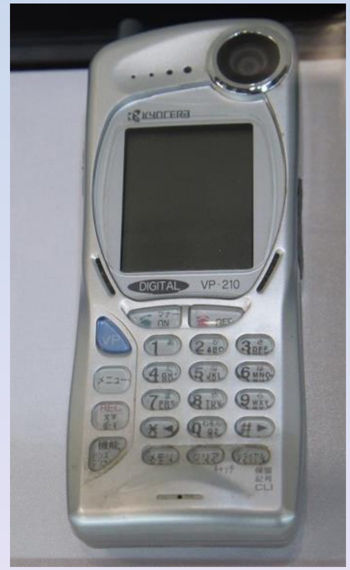
Even if the maps were presented only in shades of gray (see photo), it allowed users to upload maps to track the position and movement.

A little history: from telegraph to smartphone

1999: The first mobile phone with a **video camera** was:

Kyocera Visual Phone VP-210

!!!! Only 0.1 megapixels



	main camera	front camera
Sony Xperia Z3 Compact	20.7Mp	2.2Mp

Now, we have 5G

What is 5G?

- started at the end of 2018

- high downloading and uploading speed of the data
    - 10 - 30Gbps
    - 4G: about 100-300Mbps

? Mb / second ?????????

- access to multiple devices at the same time

- Broadband-like mobile service: high-definition streaming video without any buffering
- Autonomous vehicles
- Helps AI development

A little history: from telegraph to smartphone

What is 6G ?

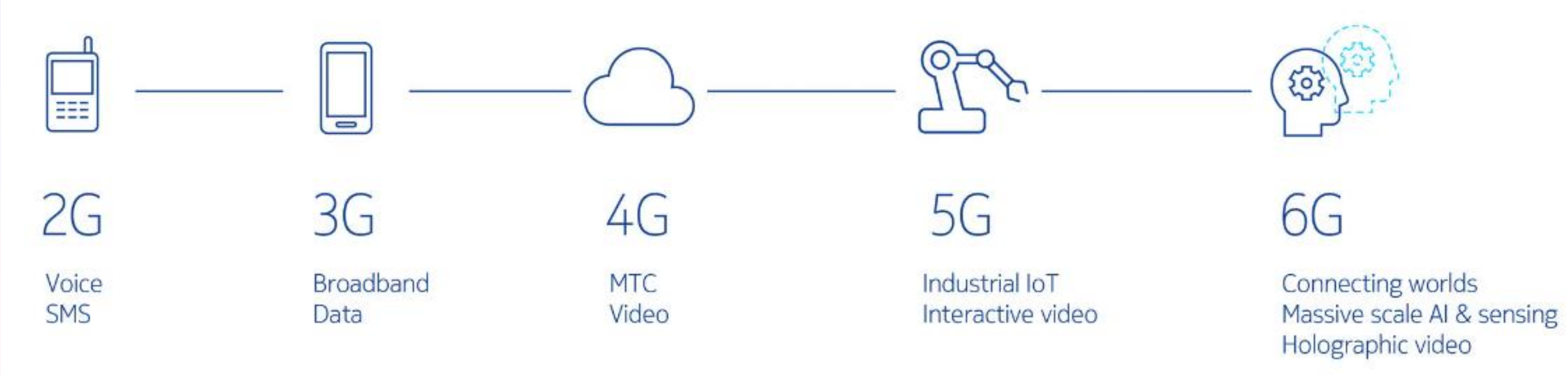
- *It is not yet a functioning technology (may be in 2030)*

- able to use higher frequencies than 5G

Millimeter waves: 30 to 300 GH and **Terahertz** radiation 300 to 3000 GHz (about 2 to 30 GHz used in 5G and Wi-Fi)

- much lower latency, i.e. about 1,000 times faster;
- about 1 terabyte per second for wireless data;
- support mobile edge computing;
- support machine-to-machine communication in IoT.

- *China has launched a 6G test satellite equipped with a terahertz system*
- *Cloud computing AND* ***edge-computing***



Smartphones - Non-standard definitions:

- a phone that can run a full operating system;

- a device that **combines a cell phone with a minicomputer** that can be held in hand (hand-held computer) and that provides access to Internet, email, data storage capacity, etc;

- [Collins English Dictionary 2012] “a mobile phone with a computer facility that may be able to interact with other computer systems, send emails and access the web”

A simple definition:

- a computer that fits in your palm

- you can use it for almost anything

- you can't live without it

?

What does the name “Steve Job" means for you?

?

Which is the most important thing he did for us?

Curiosity

a ton of smartphones contains about 300 g of gold

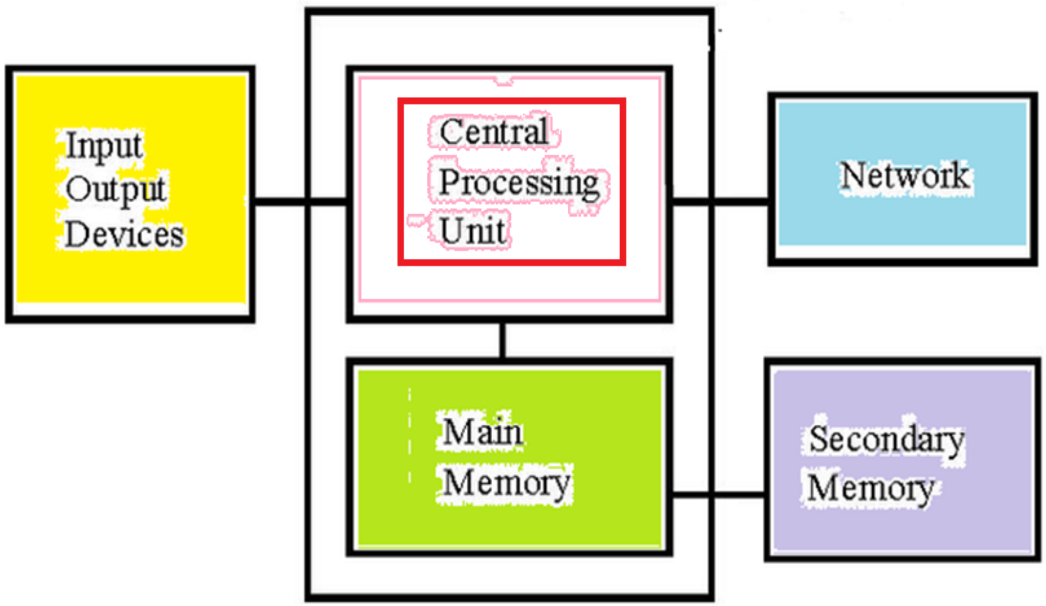
Question:

How many phones are needed to collect 1 kilo of gold?

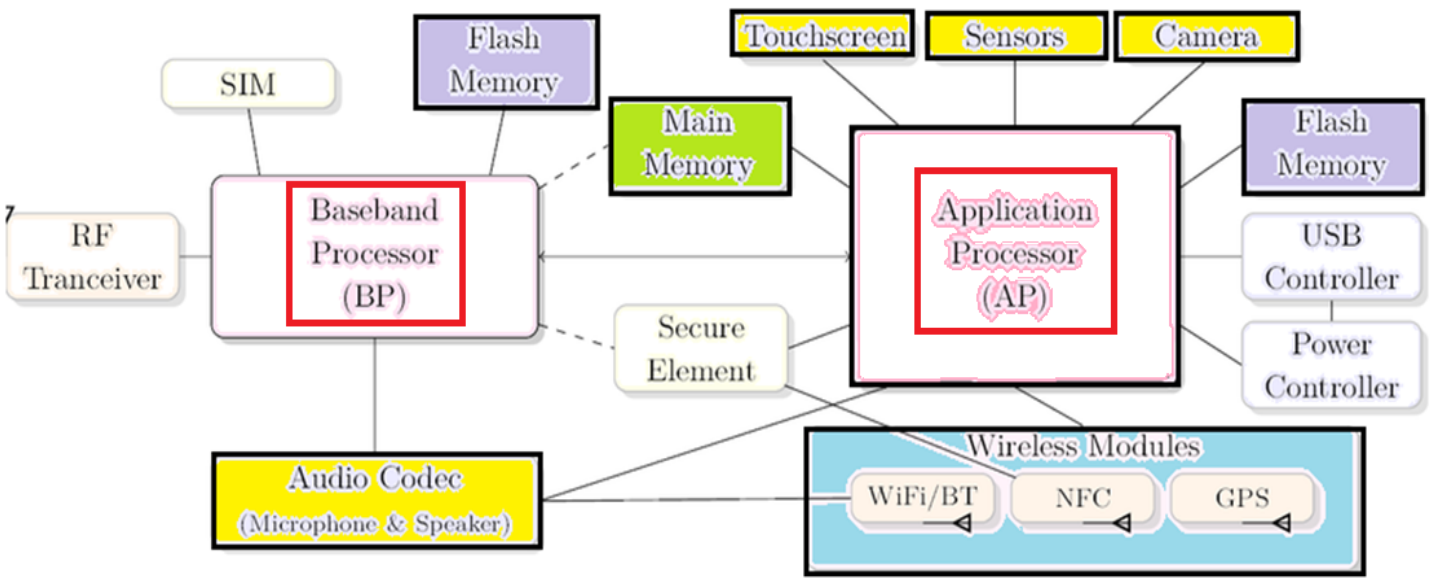
Smartphone

hardware structure

Computer hardware



Smartphone hardware



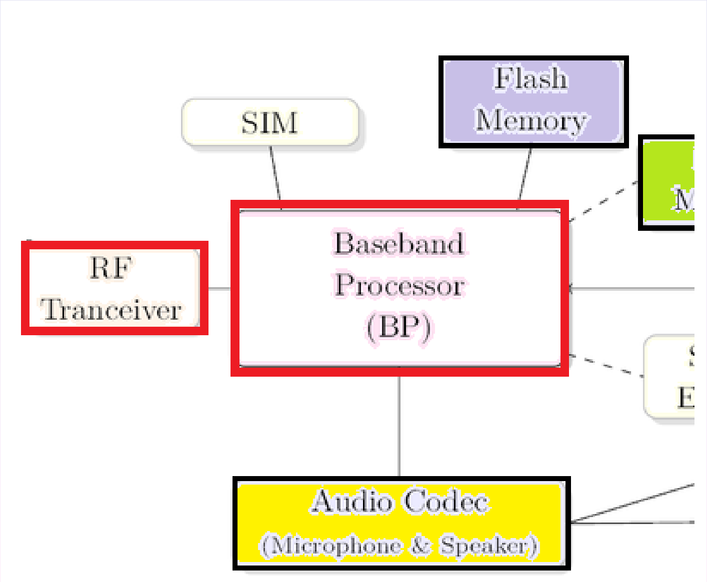
General hardware structure of a smartphone

In both architectures there are:

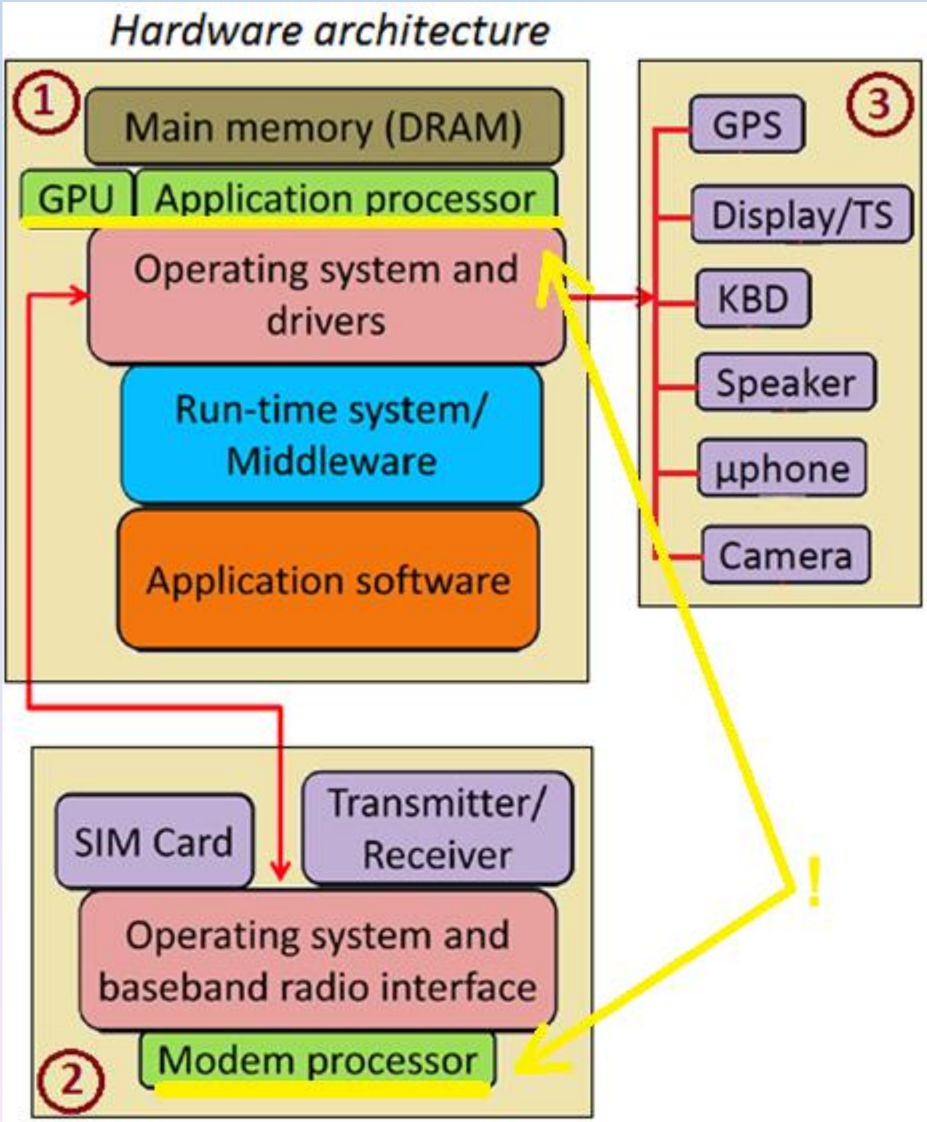
- main/user'/applications processor
- internal memory
- external memory
- input/output systems
- BUSes for communication

Smartphone architecture has an additional processor:

- Baseband processor (modem procesor)



General hardware structure of a smartphone



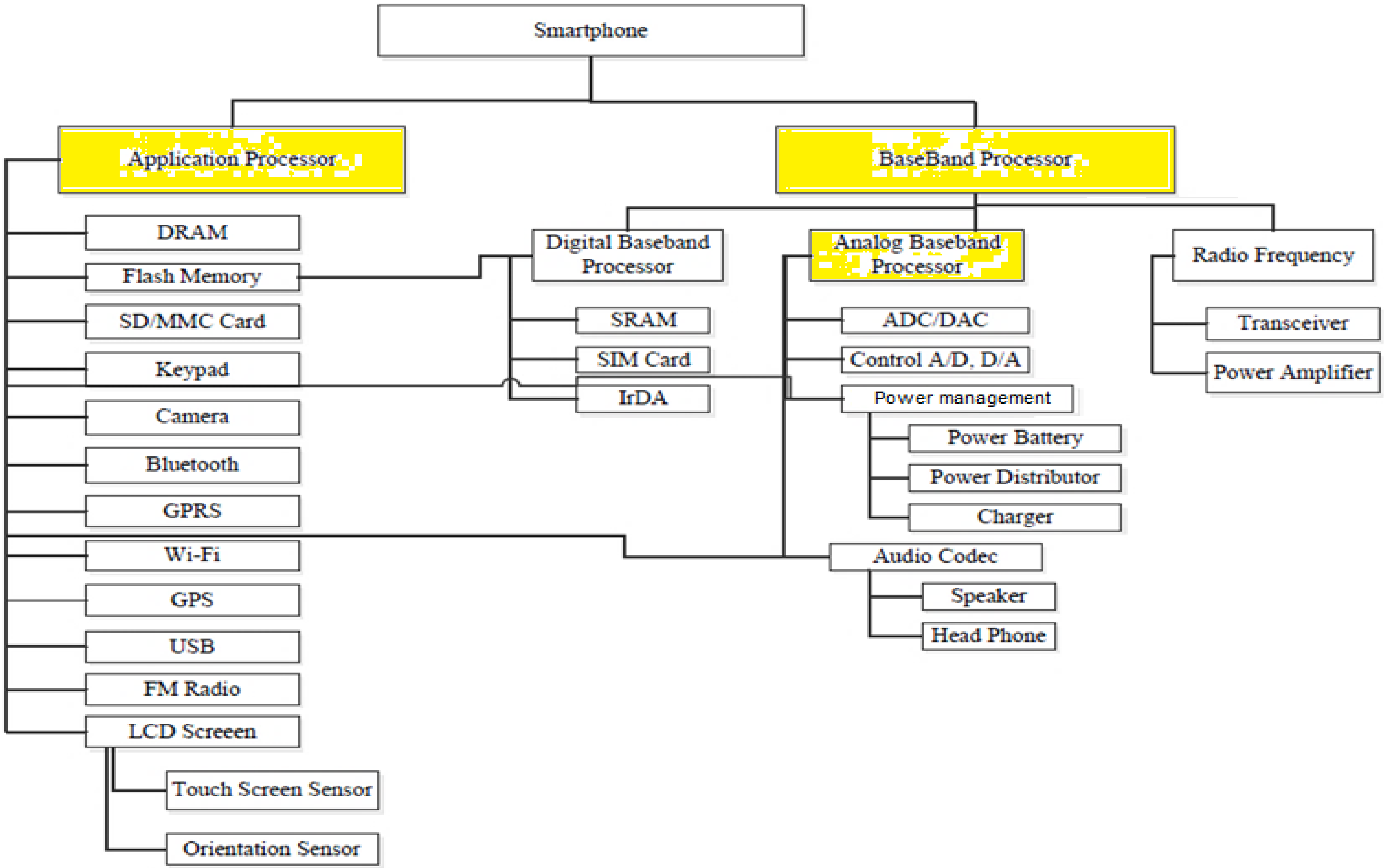
Primary components:

1. **application processor:** executes the end-user's application software with assistance from operating system (OS);

2. **baseband processor** with its own operating system components responding to the baseband radio activities (transmission and reception of audio, video, and other data contents)

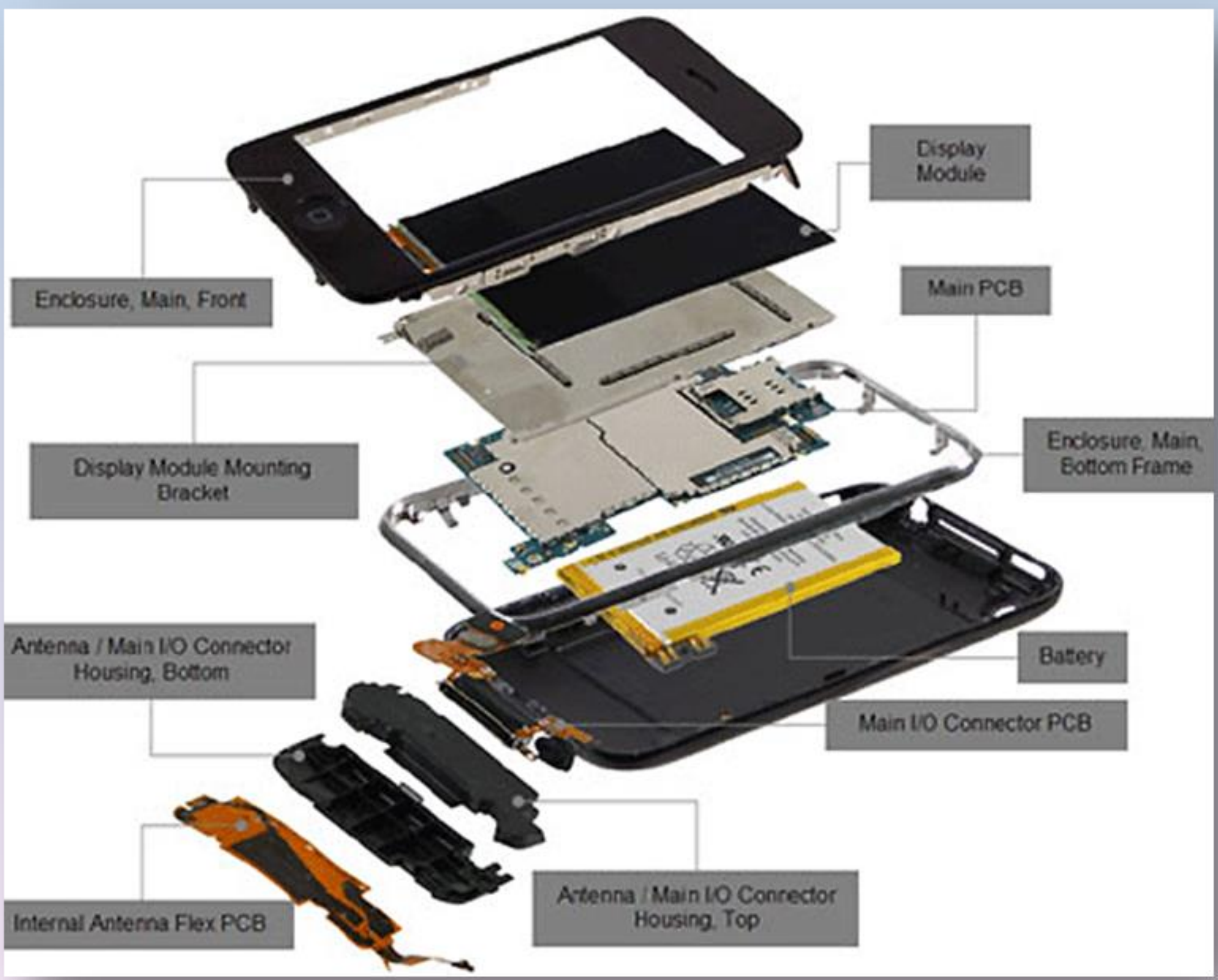
3. **peripheral devices** for interacting with the end-user

General hardware structure of a smartphone



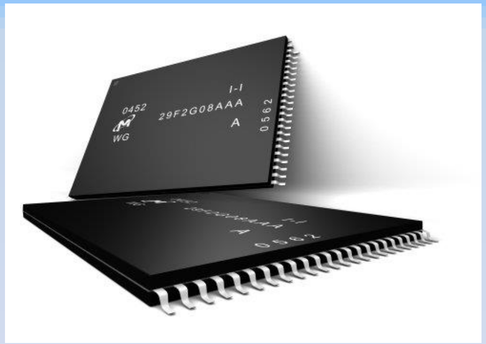
A higher detailed anatomy of smartphone

General hardware structure of a smartphone



Smartphone’s components

General hardware structure of a smartphone – zoom inside-

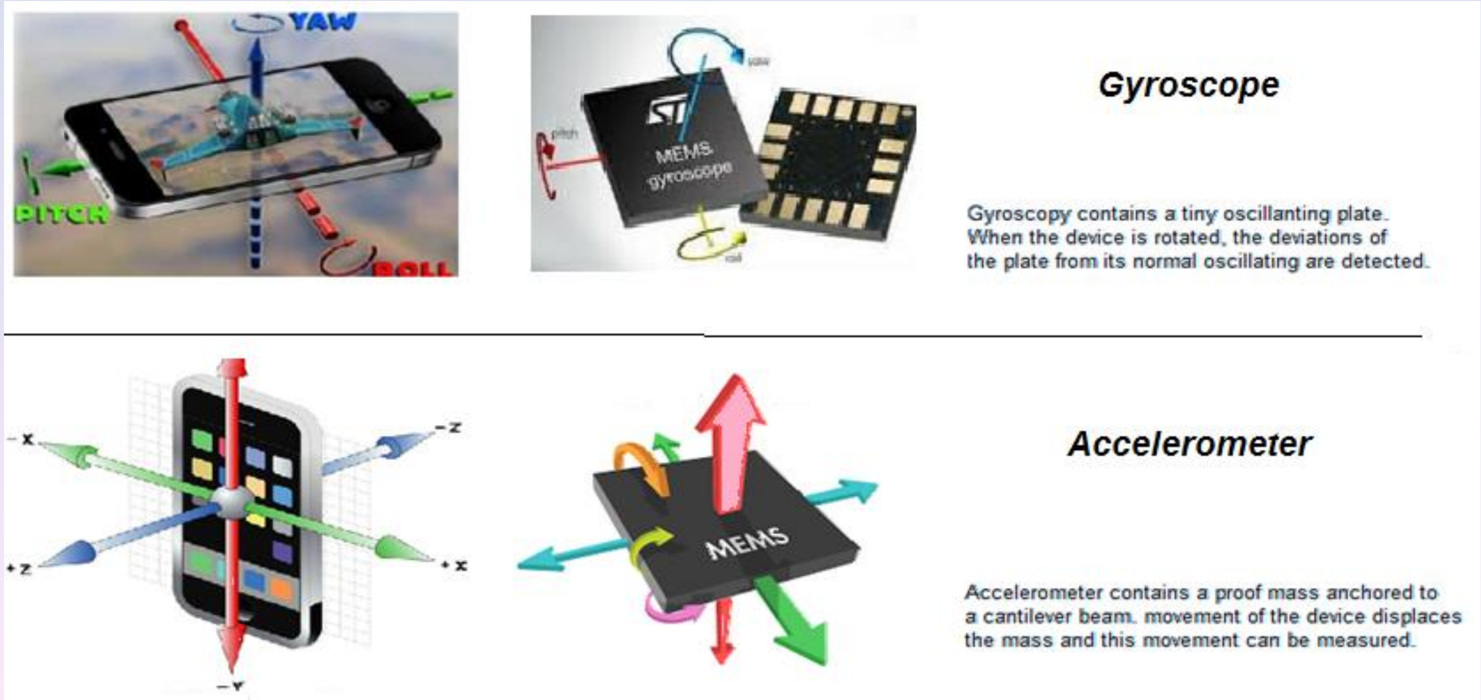


NAND Flash

App Processor

Baseband Processor

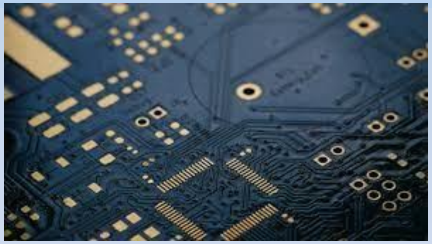
Camera Module



Accelerometer and gyroscope

General hardware structure of a smartphone

1. Mainboard (motherboard) (MB)



A mainboard for electrical connection of all components

- is made specially for every phone model (no standardization)

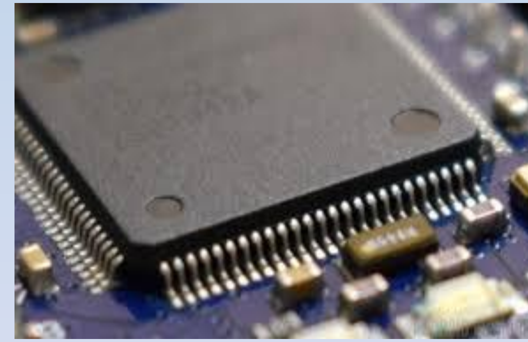
⇔no standards as in case of PCs (*nobody wants to replace smartphones' MB!)*

- all the components are not user replaceable

⇔user cannot add any new components *(**<u>except modular smartphones</u>**)*

General hardware structure of a smartphone

2. Application processor



In general: Systems on a Chip (SoC) that include:

- multiple processor (typically ARM) cores
- several graphics processing units (GPUs)
- cache memories
- memory controllers for communicating with off-chip DRAM
- audio and video decoders and encoders,
- USB controllers

- In modern smartphones, dual/quad core processor is composed of two/four processors, each one with a separate RAM and flash memory access

General hardware structure of a smartphone

Example 1

Application processor Samsung Exynos 1280

- 2022.
- for Samsung Galaxy A33 and A53 (5G)
- SoC is built on a 5 nanometer process technology
- CPU octa-core CPU (2x Cortex-A78 and 6x Cortex-A55)
- Mali-G68 MP4 GPU.
- NPU (Neural Processing Unit) [*many numbers of nerve cells and synapses that transmit and receive signals to and from each other simultaneously, just like the human brain*] [*for parallel data computing i.e. AI-based applications can run faster on low power*]
- Architecture: ARM

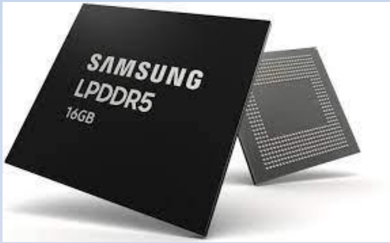
Example 2

Application processor Apple A16 Bionic

- 2022
- for iPhone 14 Pro Max smartphone
- 4-nanometer process technology
- 3.5 GHz, *six* cores - two performance cores and four efficiency cores
- 5-core GPU
- a 16-core Neural Engine (NPU - Neural Processing Unit)

- 16000 Million transistors
- 64 Bit support
- Architecture: ARM

3. RAM (internal and external memory)

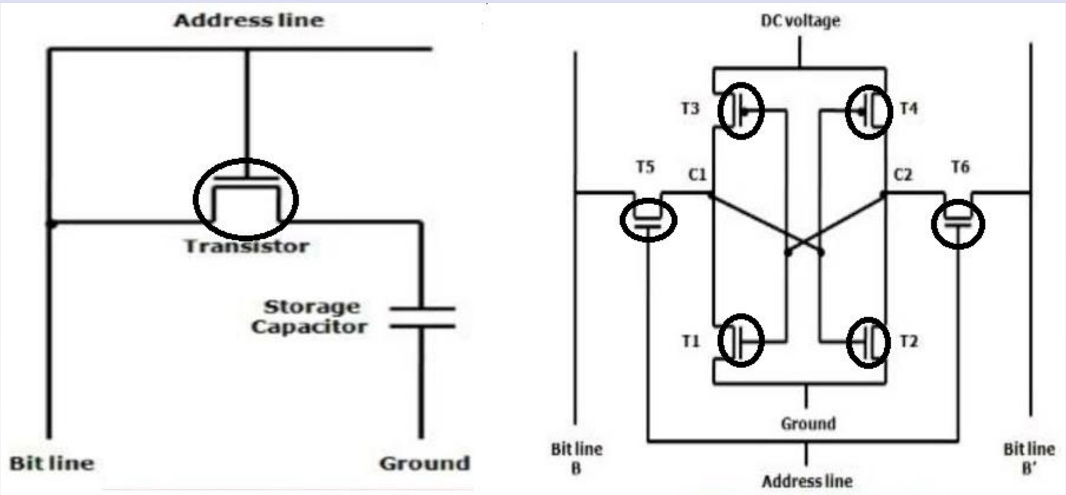


RAM is used to store:

- operating system (OS)
- apps and data
- **Size:** about 64 Gb now

Types of RAM: Dynamic RAM (DRAM) and Static RAM (SRAM).

DDR4 is better



Dynamic RAM (DRAM) Cell

Static RAM (SRAM) Cell

!!!!!!!!!!!!!!!! More RAM => more power needed (short battery life)

General hardware structure of a smartphone

Screen (display)

Types: LCD, OLED, AMOLED, Super AMOLED, TFT, IPS, TFT-LCD, IPS-LCD………

Technologies:

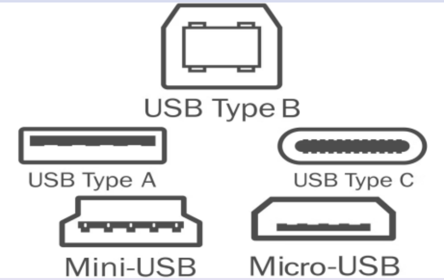
- LCD (Liquid Crystal Display)
  - array of liquid crystals illuminated by a backlight
  - low cost
  - works well in direct sunlight
  - less accurate colour representation
- OLED (Organic Light Emitting Diode)
  - an organic material that emits light when a current is passed through it
  - no backlight
  - consume less energy when black or darker colours are displayed
  - more expensive
  - thinner and more flexible (bendable or rollable displays)
- microLED (displays of the future)
  - one light-emitting diode for red, green, and blue for each dot
  - Full HD resolution (1,920 x 1,080 dots)⇔ requires 6 million microscopic LEDs using a traditional RGB (red, green, and blue) structure.

General hardware structure of a smartphone

USB port (Universal Serial Bus)

- power charging & connection with other devices;

USB Type	Number of Pins	Shape	Used in
Type A	4	Flat and rectangular	Computers, tablets, television sets, flash drives, keyboards
Type B	4	Square	Printers, Scanners
Type C	24	Symmetrical oblong	Smartphones, headphones
Mini A&B	5	Advil shaped(roughly)	Digital camera, computer peripherals
Micro A&B	5	Rounded top and a flat bottom	Smartphones, computer peripherals, video game controllers
Lightning Cable	8	Chip- like flat	Apple's devices



Speed

USB 1.0	1.5 Mbit/s
USB 2.0	480 Mbit/s
USB 3.0	5.0 Gbit/s
USB 3.1	10 Gbit/s
USB 4.0	40 Gbit/s

General hardware structure of a smartphone

- Button stop/start device;
- Buttons to adjust the volume;
- Video-camera (currently, dual);
- Status lights: show the phone state ar announce events (unloading / loading battery, missed events etc);
- Microphones (in general, up and down):
  - first used during phone calls to retrieve the user's voice;
  - second used during open calls (the speaker);

General hardware structure of a smartphone

The Accelerometer

Measures the vibration, or acceleration of motion, of a structure/device

The force caused by vibration or a change in motion (acceleration) produces an electrical charge that is proportional to the force exerted upon it.

Used to detects and measures:

a) the position;

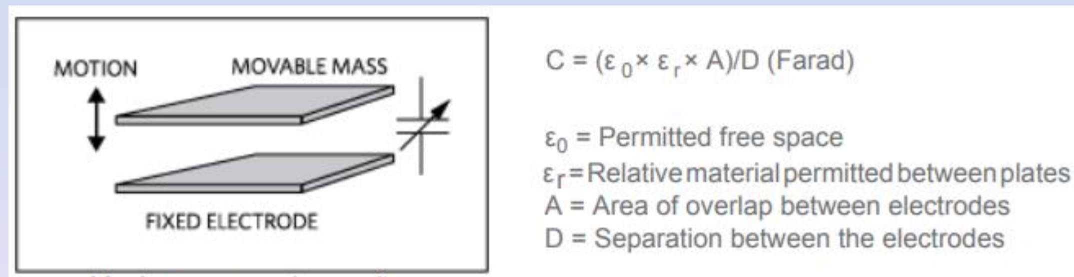
b) the movement;

c) the inclination;

d) shocks;

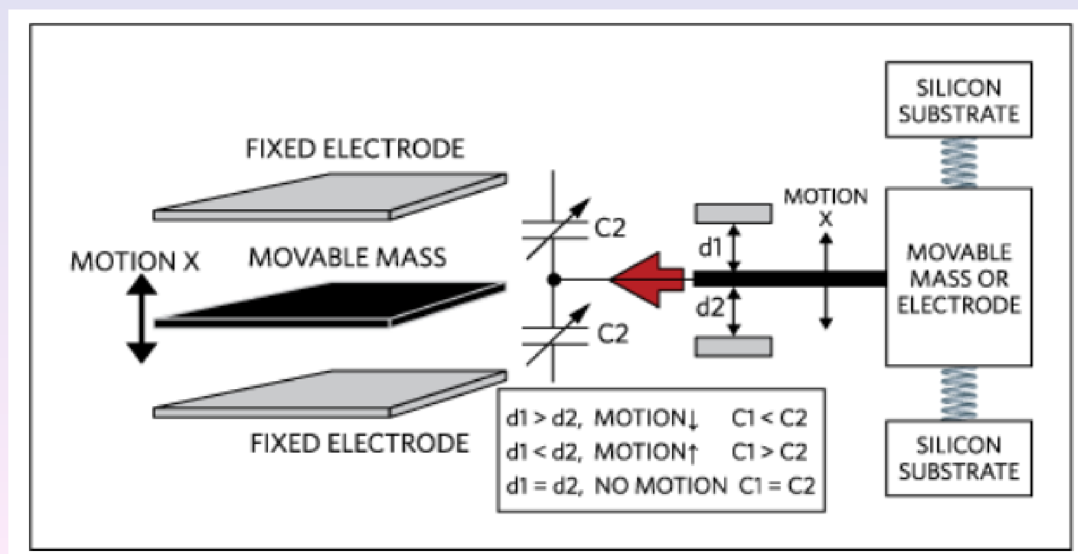
e) vibrations;

f) acceleration



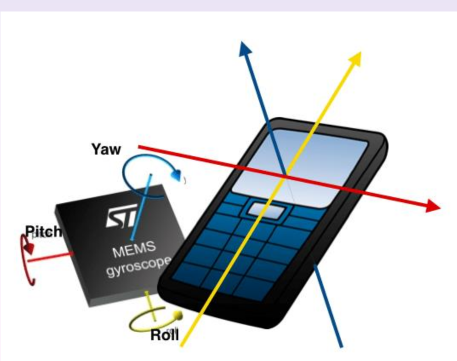
Moving mass and capacitance

The principle <- physics



Acceleration associated with a single moving mass.

the difference between C2 and C1 ⇨ the displacement of mass and its direction



General hardware structure of a smartphone

Proximity sensor

- detect how close/far away an object on the screen is (such as the user's face during a call). For example, during a call, the event sensor detects and locks the keypad to prevent accidental key presses.
- combines an infrared LED and light detector
- the sensor emits a beam of light (invisible to the human eye) that gets bounced back.

Ambient light sensor

- taking a measuring of the light in the room and adjusting screen’s brightness accordingly (if indeed it’s set to auto-adjust).

Gyroscope sensor

- for measuring rotational movement in a 3-axis coordinate system

WiFi transmitter / receiver

- allows local communications via a wireless access point;

General hardware structure of a smartphone

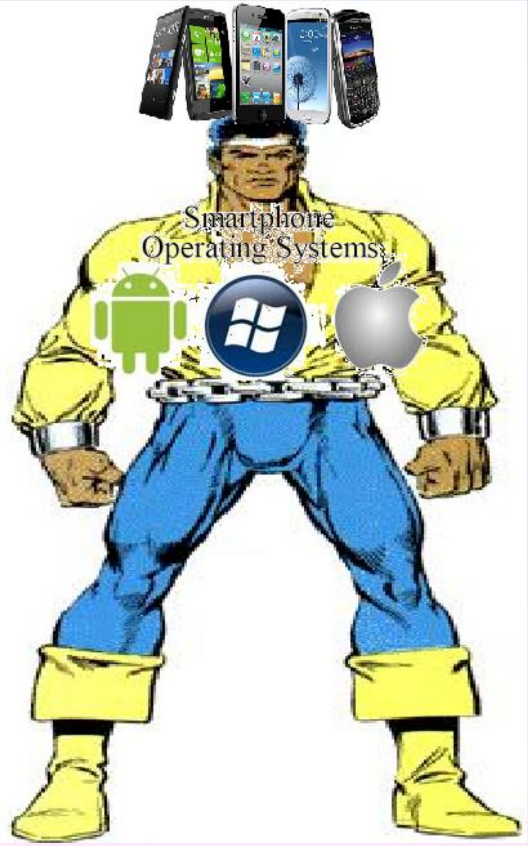
GPS (Global Positioning System) sensor

- allows providing location information via a GPS satellite;
- is a radio navigation system.
- it uses radio waves between satellites and a receiver inside phone
- phone need to be able to receive data from four or more satellites in orbit, dedicated for geolocation use.
- satellites sending a fast signal hundreds of thousands Km, so it'll take up to a minute to get your geolocation (fast driving -> some desynchronization)

GPS systems (the four most important):

- A-GPS: American GPS system
- GLONASS: Russian GPS system
- GALLILEO: European GPS system
- BEIDOU-2: Chinees GPS system

At the foundation of any mobile device (smartphone) a mobile Operating System (mOS) is.



Operating platforms – short presentation

- Mobile Operating System (mOS) are complex software (a set of data and programs) that allows smartphones, tablet or other mobile device to run mobile applications (apps)
- It is quite different from an OS used in computers
- A mOS optimizes the efficacy of the apps in mobile devices
- A mOS manages mobile multimedia functions, mobile and Internet connectivity and so on in mobile devices
- A mOS typically starts up when mobile device powers on.

The main functions of mOS':

- manage phone cellular,;
- manage network connectivity (wireless, bluetooth or USB);
- manage multimedia functions.

- Most mOSs are tied to specific hardware, with little flexibility, or not at all.

Some of mOSs are open source software

Operating platforms – short presentation

With the exception of Android (developed by Google), mOSs are developed by different mobile phone manufacturers:

- Nokia (Symbian, MeeGo, Maemo);
- Apple (Apple iOS);
- Research In Motion (RIM) (BlackBerry OS);
- Microsoft (Windows Mobile, Windows Phone)
- Samsung (Palm WebOS).

Most of mOSs are based on the Linux open-source OS.

The main components of mOSs are:

- **Kernel**:

-provides basic level control over all computer hardware devices;

-reading and writing data from/to memory;

-processing execution orders;

-determining how to interpret data received from networks.

- **User Interface (UI)**: allows interaction with the user (through graphical icons).

- **Application Programming Interfaces (API)**: allows developers to write modular code.

Operating platforms – short presentation

***Kernel* : the main component of any OS.**

- It is like a bridge between applications and the data processing performed at the hardware level.

At the beginning, when OS is loaded into system memory, the kernel loads first and remains in memory until the OS is in use (TSR-terminate and stay resident)

- is responsible for low-level tasks such as disk management, task management and memory management;
- provides and manages hardware system resources, allowing other programs to run and use these resources;
- sets up memory address space for applications
- loads files with application code into memory
- sets up the execution stack for programs

Operating platforms – short presentation

Definition

User interface **(UI) is a broad term for any system, either physical or software based, that allows a user to connect with a given technology.**

UI depends on devices and software programs.

Even more UIs have some basic similarities, although each one is unique.

The current types of user interface are graphical user interface (GUI), like Windows OS, Android OS…
⇔ all are driven mainly by icons or images rather than text commands.
AND GUI can include touch screen interfaces

Operating platforms – short presentation

Application Programming Interface (API)

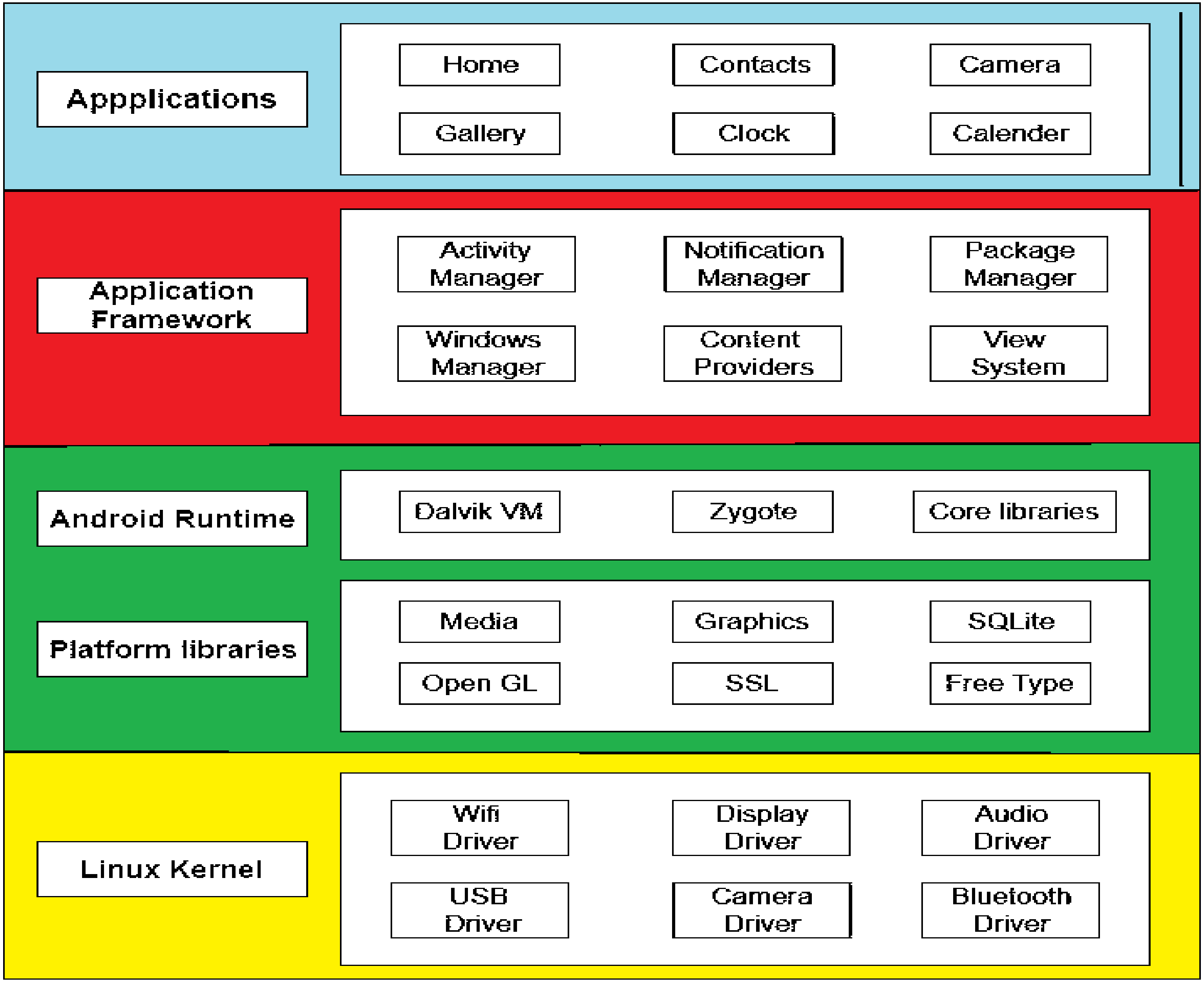
allows a programmer to interact with an application using a collection of callable functions.

- The goal of an API is to allow programmers to write programs that will not stop to run well if the underlying system is upgraded.

- An API can be general or specific. The full set of a general API is bundled in the libraries of a programming language.

- An API is language dependent or independent:
  - language dependent: API is only available by using the syntax and elements of a particular language, making it more convenient to use.
  - language independent: API is written to be called from several programming languages.

Android Architecture



Android Architecture

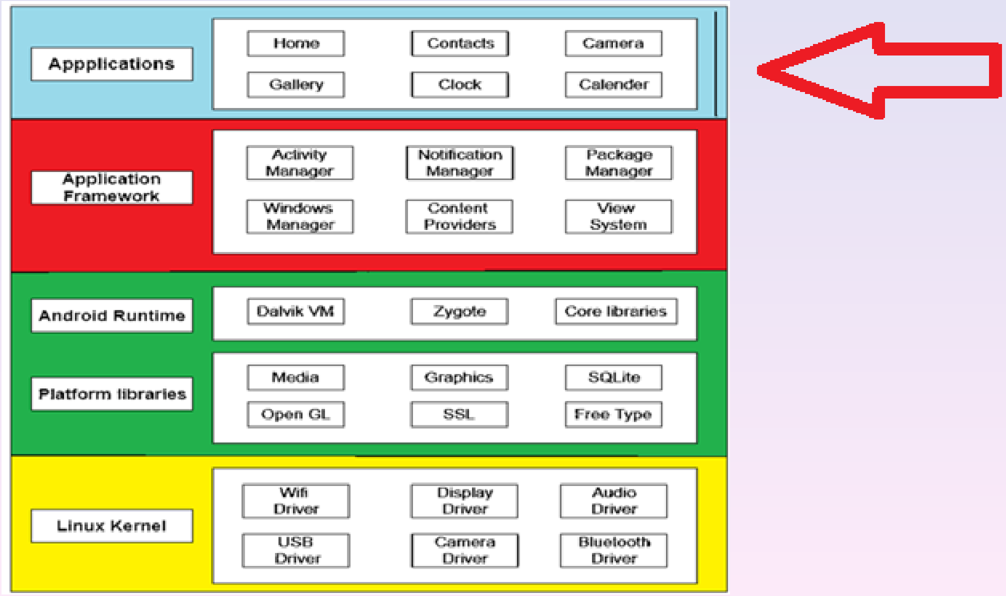
Applications

- the top layer of android architecture.

- the pre-installed applications (home, contacts, camera, gallery, applications downloaded from the play store like chat applications, games etc.) will be installed on this layer.

- the needed classes and services are provided by the **Application framework**

.



Android Architecture

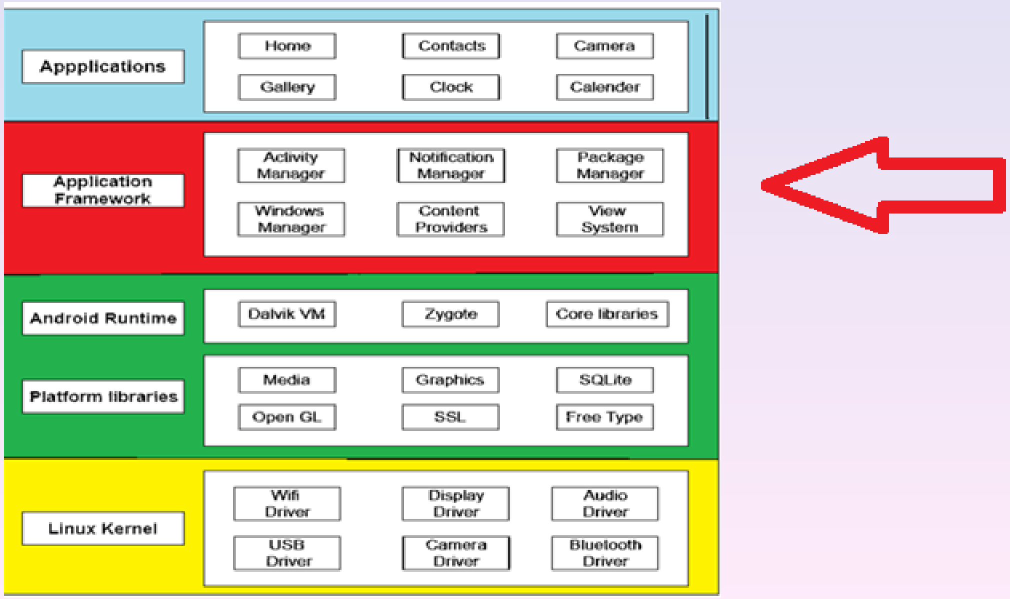
Application framework

- provides several important classes which are used to create an Android application.

- provides a generic abstraction for hardware access

- helps in managing the user interface with application resources.

- includes different types of services: activity manager, notification manager, view system, package manager etc., which are helpful for the development of newr application.



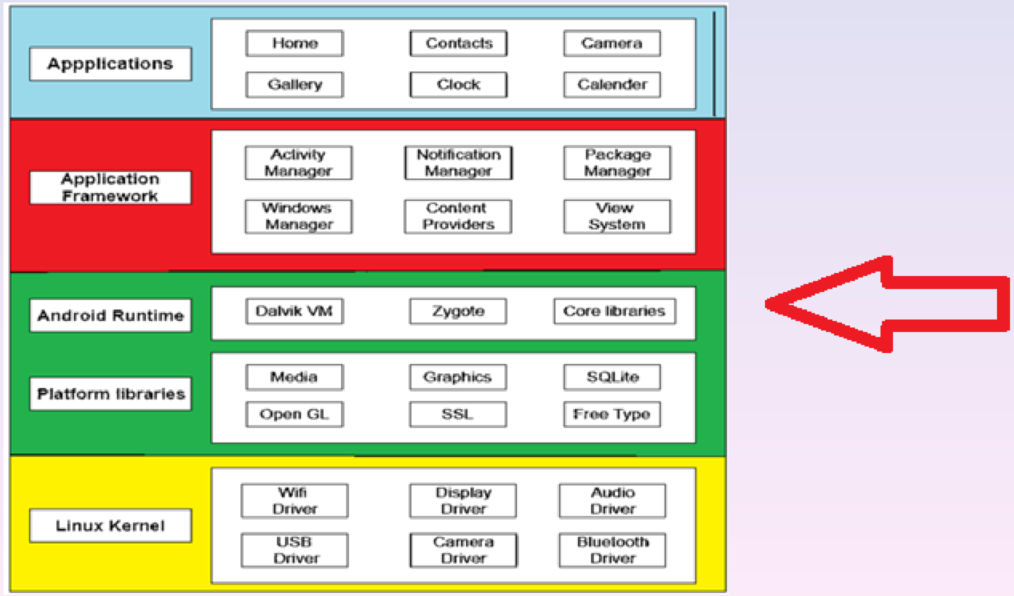
Android Architecture

Application runtime

-contains components like core libraries and the Dalvik virtual machine(DVM).

- **Dalvik Virtual Machine (DVM)** is a register-based virtual machine and specially designed and optimized for Android to ensure that a device can run multiple instances efficiently.

- the core libraries enable us to implement android applications using the standard JAVA or Kotlin programming languages.



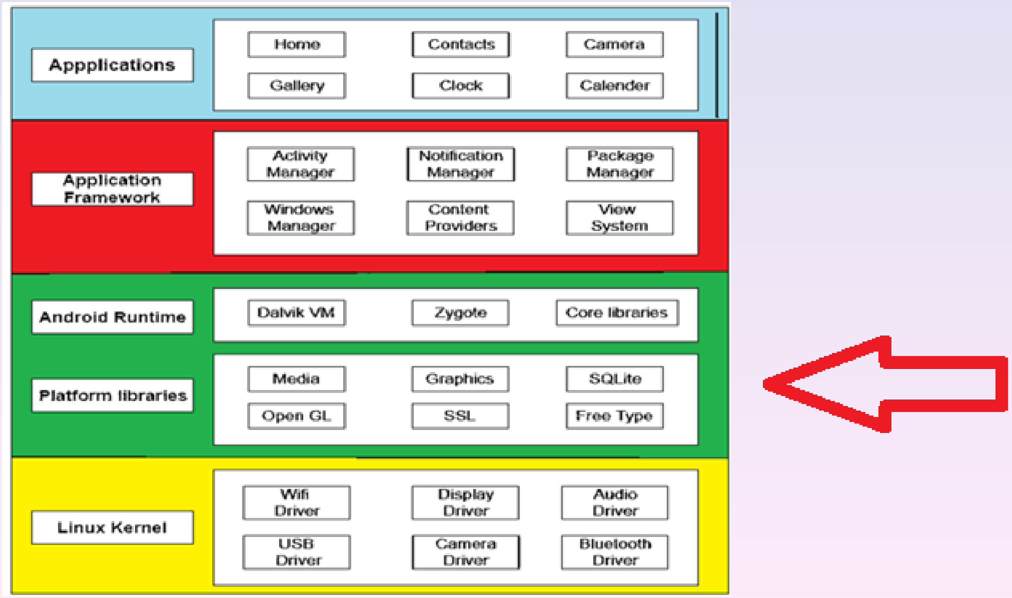
Android Architecture

Platform libraries

includes various C/C++ core libraries and Java based libraries such as Media, Graphics, Surface Manager, OpenGL etc. to provide a support for android development.

- **Media** library provides support to play and record an audio and video formats.)
- **Surface manager** responsible for managing access to the display subsystem.
- **SGL** and **OpenGL** (both cross-language, cross-platform application program interface (API)) are used for 2D and 3D computer graphics.
- **SQLite** provides database support
- **FreeType** provides font support.
- **Web-Kit** This open source web browser engine provides all the functionality to display web content and to simplify page loading.
- **SSL (Secure Sockets Layer)**

is security technology to establish an encrypted link between a web server and a web browser.



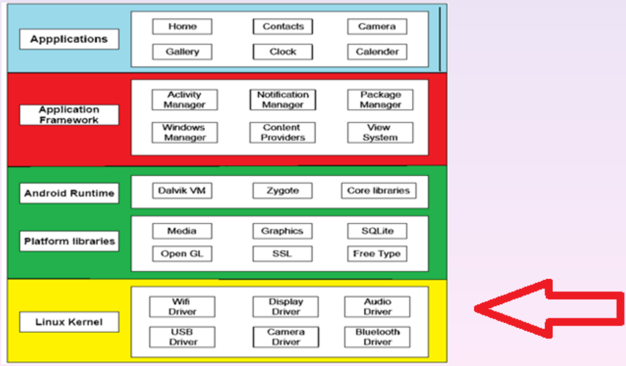
Android Architecture

Linux Kernel

- is heart of the android architecture
- manages all the available drivers such as display drivers, camera drivers, Bluetooth drivers, audio drivers, memory drivers, etc. which are required during the runtime.
- will provide an abstraction layer between the device hardware and the other components of android architecture. It is responsible for management of memory, power, devices etc.

The features of Linux kernel are:

- **Security:** The Linux kernel handles the security between the application and the system.
- **Memory Management:** It efficiently handles the memory management thereby providing the freedom to develop our apps.
- **Process Management:** It manages the process well, allocates resources to processes whenever they need them.
- **Network Stack:** It effectively handles the network communication.
- **Driver Model:** It ensures that the application works properly on the device and hardware manufacturers responsible for building their drivers into the Linux build.



Android Architecture

Dalvik vs. ART in Android

- ART (Android runtime) and its predecessor Dalvik were originally created specifically for the Android project. ART and Dalvik are compatible runtimes running Dex bytecode, so apps developed for Dalvik should work when running with ART.

- ART from 4.4 version KitKat

- **Android runtime (ART**) is the managed runtime used by applications and some system services on Android.

Android Architecture

Dalvik vs. ART in Android

DALVIK

- **Dalvik Virtual Machine or DVM** is a Register-Based virtual machine, designed and written by Dan Bornstein.
- Dalvik is a discontinued process virtual machine (VM) in the Android OS that executes applications written for Android.
- Dalvik bytecode format is still used as a distribution format, but no longer at runtime in newer Android versions.
- Android itself is a Linux system with Dalvik sitting on top of it.
- DVM takes android app, turns them from java code into bytecode that the Linux system can run. This could lead to a slowdown because compilation at runtime especially during runtime is time-consuming.
- There are 2 types of files:

-.dex(Dalvik Executable file) file is an android's compiled code file. These .dex files are then zipped into a single .apk file.

-.odex file is created by the Android operating system to save space and increase the boot speed of an Android app (a .apk file).

- dexopt is used to optimize DEX to ODEX (optimized DEX)which contains the optimized bytecode. So the whole process in DVM can be summarized as:

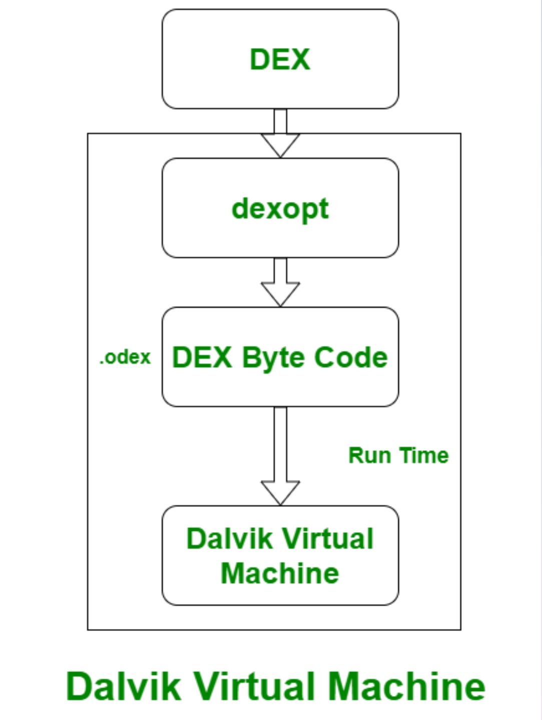
Android Architecture

Dalvik vs. ART in Android

DALVIK

The process in DVM: ***JAVA source code(.java) –> Bytecode(.dex) –> DVM***

? Virtual Machine ?



Final conclusions:

- DVM is better for low storage devices.

- it is slower as compilation is done after installation.

Android Architecture

Dalvik vs. ART in Android

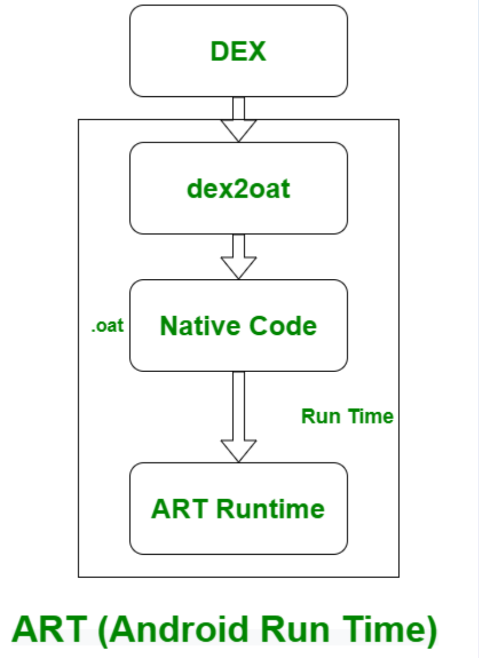
ART (Android Runtime)

- from 4.4 version KitKat
- an alternative to DVM. ART(Android Run Time)
- a successor of DVM which uses the same bytecode and .dex files (but not .odex files
- Android 5.0 “Lollipop” is the first version in which ART is the only included runtime.
- higher performance as no need to convert code to bytecode then compile.
- need more storage space and a little longer to install
- hence, instead of relatively small java code, we have larger bytecode/machine code.
- but, we have precompiled most of code => apps a little faster and more performant than in Dalvik.
- **But this approach takes a little more storage space.**

Android Architecture

Dalvik vs. ART in Android

ART



- dex2oat is used to optimize and compile .dex into a .oat file which may contain machine code in the ELF format.
- ART compiles apps using the on-device dex2oat tool. This utility accepts DEX files as input and generates a compiled app executable for the target device.
- when an app is installed, Android automatically optimizes app data and creates a corresponding OAT file.
- an OAT file is created by the Android operating system in order to speed up the loading time of an Android app (.APK file). Android uses this file to load the app more quickly, creating a better user experience.

Android Architecture

Dalvik vs. ART in Android

ANDROID RUN TIME	DALVIK VIRTUAL MACHINE
Rebooting is significantly longer	Faster Booting time
The cache is built during the first boot	Cache builds up overtime
Consumes a lot of storage space internally due to AOT	Occupies less space due to JIT
Works best for Large storage devices	Works best for small storage devices
Experimental and new – not much app support comparatively	Stable and tested virtual machine
Extremely Faster and smoother Faster and app loading time and lower processor usage	Longer app loading time
Uses AOT compiler(Ahead-Of-Time) thereby compiling apps when installed	Uses JIT compiler(JIT: Just-In-Time) Thereby resulting in lower storage space consumption
Reduced application lagging and better user experience	Application lagging due to garbage collector pauses and JIT
App installation time is longer as compilation is done during installation	App installation time is comparatively lower as the compilation is performed later
ART converts it just once at the time of app installation. That makes CPU execution easier. Improved battery life due to faster execution.	DVM converts bytecode every time you launch a specific app.
It is faster.	It is slower than ART.
It provides optimized battery performance as it consumes less power.	It does not provide optimized battery life as it consumes more power.
It lags in term of booting.	While considering Booting, then this device is fast.

Dalvik vs. ART in Android

Which one is better?

in the long run, ART is better, but the apps do get bigger, requiring large storage space over time

DVM converts bytecode every time you launch a specific app. But ART converts it just once during the installation of the app. The last makes CPU execution easier.

ARM is also an architecture: DVM/ART runs on top of ARM.

Both of them can not replace each other.