# Contents

# 1 Hardware

## 1.1 Motherboard

Motherboards generally consist of

- CPU
- Northbridge - Memory controller hub
- Southbridge - I/O controller hub
- RAM
- Disks (i.e. HDD/SSD)
- Several functional cards

## 1.2 Connections

- CPU and RAM chips have a bus that connects it to the other devices.
  - They act like a bridge connecting the two
- Nowadays more advanced buses are used, which can allow up to 333MB/s speeds
  - This depends on a bus width.
  - More wires allow more data to be transferred, however result in a larger chip size.
- Having more chips and connections can lead to communication overhead / voltage drop, which results in more power usage

# 2 System on a chip (Soc)

A system on a chip or system on chip (SoC) is an integrated circuit (also known as a ”chip”) that integrates **all** components of a computer or other electronic system. It aims to reduce communications overhead. In here transistors provide computation **as well as** storage. PoP technology also enables smaller packages with shorter electrical connections

# 3 Package-on-Package

Due to RAM not being included in Soc (because it uses a lot of space) we have to put it somewhere else. So we put a separate package **on top of the SoC**, which is called Package-on-Package or PoP for short. This helps to seperate memory and logic productions as well as allowing multiple components to share RAM.

# 4 ARM CPUs

Almost all (95%) of smart phones use ARM CPUs. It's sold as **a design** not a physical device.

- SoC processor cores often use the ARM architecture because it is a soft processor specified as an IP core and more power efficient than x86.
- Chip manufactures can add ARM CPU to bespoke SoC vs buying a chip from Intel
- Fast and efficient
  - Good for mobile applications
  - Best use of battery (more instructions → more battery usage)
- High code density
  - Only includes simple instructions that can be excetured in one clock cycle. (e.g. no divide instruction)
  - Less moving stuff around, best use of space

## 4.1 ARM architecture

Only includes simple instructions that can be executed in one clock cycle

- This means lack of certain instructions like division, but this can be implemamented in the compiling stage
- It can only operate on registers, where as x86 can operate on direct memory as well

Registers

- 16 registers
- Load/Store architecture
- Each instruction 32-bits long
- Constants must fit into the 32-bit instruction width. Meaning that they **can't utilise all 32bits**, as 12bits are required for the register

## 4.2 ARM conditional execution

Traditional approach is to execute a compare branch if the condition is met, however branches are 'expensive' (difficult to make parallel efficiently). ARM allows **any instruction** to made conditional. Thus removes the need for an expensive branch and leads to smaller code footprint.

## 4.3 ARM and Thumb

Every ARM instruction is 32 bits long. This can take up a lot of memory thus slowing execution down with constant memory requests. This is attempted to solve by using thumb. Thumb is a 16-bit version of the ARM instruction set and it's of variable length. This was done by taking **the most popular** ARM instructions used and encoding them as 16bit values.

## 4.4 ARM big.LITTLE

ARM's latest processor contains two cores

- 2.3 GHz quad core optimised for performance (big core). Having high power and performance
- 1.7 GHz quad core optimised for energy efficiency (LITTLE core). Having low power and performance

Two Cores are architecturally consistent. System can switch between the two as appropriate for the task in hand. This has an advantage of being able to choose less powerfull core for background tasks like: alarm clock, notifications, etc.

## Reference section

**placeholder**