Embedded Platform Architecture (Constituent part of the platform , roles / details)

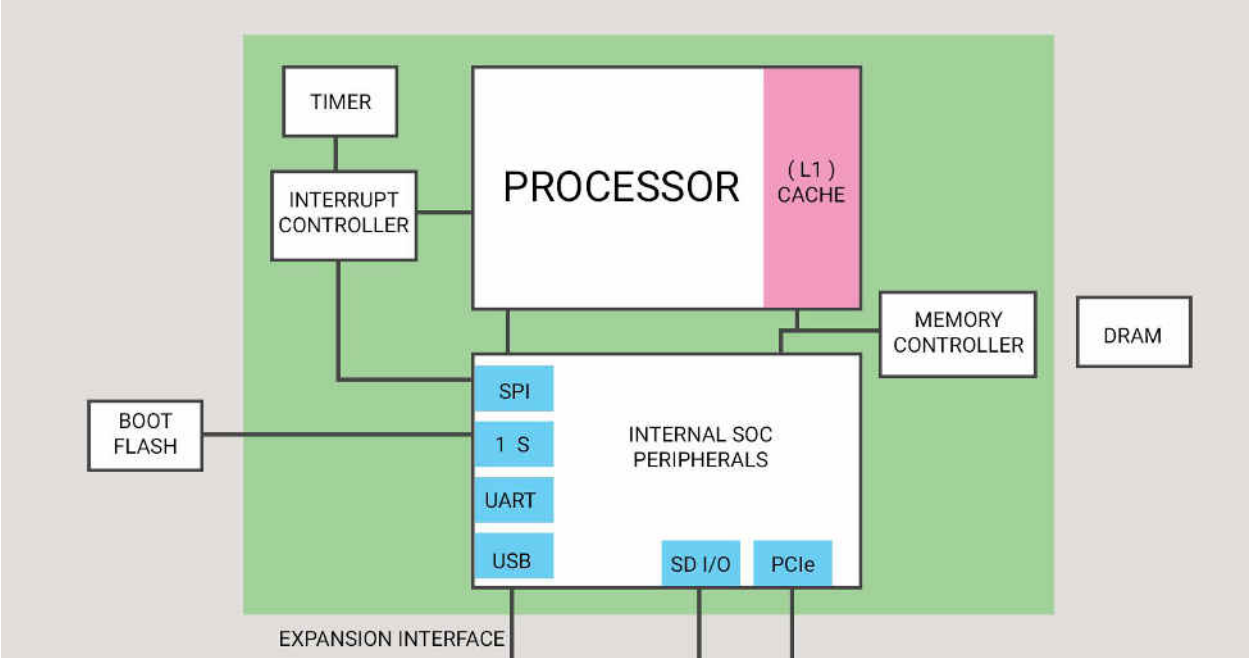
The key to understanding the interactions in an embedded system is the ***memory map*** and the associated register maps of the devices.

This means where is everything from the perspective of the processor. Things differ from with each processor. Intel differs from ARM for example.

Directly & Indirectly
In this map some devices are **directly** visible to the software (and the software can access) because they are mapped to physical addresses in the processor's address space.

Other are attached over a bus so a level of indirection is necessary. I can communicated to a device over a bus, but I am going over a bus to find it so tere will be latency.

System on a chip SOC (Overview)



The Processor (We need at least one or else there is no system !)

- Sometimes there are many processors
- Smartphones
    - Smartphones are quite complex and run many processors
    - Application Processor runs the o/s , the UI & Applications
    - Base band processor runs the wireless stack (for signal strength)
    - Audio processor (specific to its task)
    - Camera processor (specific to its task)

This is how many cooperative processors can work together in one device.

THe trend for smartphones is heading towards consolidating all of these separate processors into a single processor that can be partitioned to run multiple execution environments.

- Good for economic reasons
- Good for getting rid of the interconnectivity requirements between processors.

The processor is typically 32 bits

- 32 bit data / address registers.
- Can have 16 bit for lower performances and lower address spaces.
- Can have 64 bit for higher performances and higher address spaces.

Instructions Set

There are two main philosophies , CISC & RISC

THere are knock on programming consequences associated with each at the assembly language level. Intel uses CISC with variable length length address bits that can be packed into memory more efficiently, there are no unused bits but there is a trade off.

The choice may affect system performance but it is less meaningful from the programmer's perspective

---

Complex Instruction Set COmputing CISC

All Intel (architecture) processors have a CISC ISA (Instructions Set Architecture)
They usually contain variable length instructions which gives us a more compact memory encoding.

Reduced Instruction Set Computing RISC
ARM Processors , MPIs , Power PC are all considered to be RISC , they are considered only because RISC can be ambiguous to define.

Scalar Or SuperScalar ?

Superscalar architectures have multiple copies of key functional units within the CPU which allows for parallel execution.

For example , the Intel Atom has two arithmetic logic units (ALU) , thus more than one execution per clock cycle can be maintained, depending on the application and on the cache hit rate.

If there is only one ALU then the execution must be sequenced, with more that one they can execute independently if both instructions are in cache and within each clock cycle.

THis is superscalar and this is the trend of things to come.