Embedded
COMPUTING DESIGN

What to Know When Selecting a Hardware Platform for Your Next Project

By Lisandro Pérez-Meyer - Platform Engineer and Stephanie Van Ness - Associate Director of Marketing | INTEGRATED COMPUTER SOLUTIONS | January 24, 2025

Selecting the right hardware platform for your software project can feel overwhelming. Whether you're developing a simple embedded application or a complex, resource-heavy system, the hardware you choose plays a crucial role in determining your product's performance, features, and scalability. While many factors must be considered, the impact of these decisions often manifests in subtle ways that may not be immediately obvious.

Here's a look at key considerations and how your choices can influence the behavior of your software.

The Core of Your Board: The System on Chip (SoC)

Unlike traditional desktop PCs, which feature modular components like the CPU, RAM, and storage spread across different parts of the motherboard, embedded systems generally use a System on Chip (SoC). An SoC combines essential components such as the CPU, GPU, and peripherals into a single chip.

Features of SoCs

SoCs can include various integrated features depending on the specific model. For instance, they may come with a Graphics Processing Unit (GPU), machine learning capabilities, video encoders/decoders (e.g., H.264, H.265), and peripheral interfaces like USB, UART/serial ports, Ethernet, Wi-Fi, Bluetooth, and even camera connections. The SoC may also include memory controllers, security modules (such as Trusted Platform Modules), watchdog timers, pulse-width modulation (PWM) controllers, and other specialized components.

While the variety of SoCs is vast, each one is unique, offering different combinations of capabilities and configurations. The SoC's design will significantly impact your project's final form and performance, so selecting the right one requires understanding both its specifications and the external components it will need to work with.

External Components: RAM and Storage

Although SoCs include many integrated features, they still require external components like RAM and non-volatile memory (e.g., eMMC or NAND). The selection of these components will depend on your application's memory requirements and the SoC's specifications. Understanding the interplay between the SoC and these external elements is crucial for ensuring optimal performance.

Pin Multiplexing: Unlocking the Full Potential of Your SoC

Another key consideration when selecting an SoC is its pin configuration. SoCs typically come with a set of pins that can be assigned to different functions. However, due to the limited number of pins, many SoCs rely on pin multiplexing, which allows a single physical pin to be used for multiple functions. For example, a pin could serve as either a UART (serial port) or an I²C interface, depending on the configuration.

While this adds flexibility, it also introduces complexity. You may need to trade off some features to access others. For example, if you choose to use multiple UART interfaces, you might lose access to other peripherals like I²C. This is a critical consideration when designing embedded systems and choosing the right SoC.

The SoC Landscape: Exploring NXP's i.MX Series

To give you a sense of the variety in SoC options, let's look at one popular vendor: NXP. NXP's i.MX series of ARM processors offers several families, each designed for specific use cases: applications, automotive, edge computing, and vision.

For instance, within the i.MX 8 series, you'll find multiple variations that cater to different needs. If you're building an industrial application with minimal user interface, the i.MX 8M Mini might be the best fit.

On the other hand, if you need powerful machine learning capabilities and a robust GPU, the i.MX 8M Plus could be the right choice. Each variant of the i.MX 8M Plus has its own unique features, with differences in packaging and internal capabilities.

This example demonstrates the vast range of SoC options available, emphasizing the importance of carefully considering the features and configurations you need before making a decision.

Additional Components to Consider

Once you've selected the right SoC, you'll need to choose several other critical components to complete your embedded system. Here are some key considerations:

- **RAM:** Choose RAM that meets your performance requirements and works seamlessly with your SoC.
- **Non-Volatile Memory:** While eMMC is common, other memory types might be more suitable for your project, depending on your needs.
- **Oscillator:** Ensuring your system runs at the correct speed and stability is essential for reliable performance.
- **PCB Design:** The layout of your Printed Circuit Board (PCB) plays a crucial role in connecting all these components. For instance, integrating complex components like video displays or RAM requires careful design to ensure signals are routed correctly.

In addition to these, your device might also need components like:

- **Ethernet PHY:** For network connectivity
- **CAN Transceiver:** If you plan to use Controller Area Network (CAN) communication
- **UART to Serial Converter:** For serial communication, if required

One more thing: SoCs typically operate at specific voltage levels (e.g., 1.8V, 3.3V, or 5V), while your peripherals may require different voltage levels. As such, additional circuitry for voltage level shifting may be necessary to ensure compatibility between the SoC and peripherals. This is an important part of the integration puzzle.

The Software Stack: Bootloaders, Kernels and OS

Once your hardware is in place, you'll need to address the software layer, which starts with the bootloader. The bootloader's job is to initialize the system's basic components (like the CPU and RAM) and load the operating system. It also manages device initialization through a configuration file known as the device tree, which describes the peripherals and their configurations.

After the bootloader finishes its task, the system hands over control to the Linux kernel, which is responsible for managing the hardware and providing an interface for the application. The kernel also uses a device tree to identify and control the peripherals connected to the system.

But the kernel alone is not enough. You'll also need all the applications that you expect on a typical Linux setup, normally called "userland". All this software stack makes up the Linux distribution (distro) that forms the foundation of your system.

Custom Design vs. System on Module (SoM)

When embarking on an embedded project, you might consider designing your own custom PCB from scratch. While this is possible, it's often more practical and cost-effective to use a System on Module (SoM). SoMs are pre-assembled, ready-to-use solutions that integrate an SoC, RAM, non-volatile memory, and other essential components on a single board. They typically come with a carrier board to handle power supply and additional connections.

Benefits of Using SoMs

Pre-Integrated, Maintained Software: SoM vendors handle much of the software integration, providing up-to-date operating system stacks and keeping the software secure and stable.

Development Kits and Schematics: SoM vendors often offer development kits with carrier boards and schematics, helping you get started quickly.

Offloading the Integration Effort: SoMs take care of much of the hardware and software integration, reducing the risk and complexity of your project.

Scaling Up Production: SoMs are great for projects with smaller production volumes (under 5,000 units per year), where custom hardware development may not be cost-effective.

Limitations of SoMs: While SoMs offer many advantages, they may not be suitable for large-scale production or highly specialized projects. If your project requires a custom hardware configuration, designing your own PCB might be necessary.

Choosing the Right Hardware Platform is Essential to Project Success

The choice of an embedded hardware platform can significantly affect your project's success. By carefully considering factors such as the SoC, peripherals, pin multiplexing, and external components like RAM and storage, you can ensure your hardware meets the performance and functionality requirements of your application.

When in doubt, leveraging a SoM can save you considerable time and effort, particularly in terms of software integration and development. Ultimately, your goal is to choose a platform that abstracts away hardware complexities so that your application developers can focus on creating innovative solutions.

Lisandro Pérez-Meyer
Platform Engineer
Integrated Computer Solutions

Stephanie Van Ness
Associate Director of Marketing
Integrated Computer Solutions

ICS

Embedded
COMPUTING DESIGN

January 24, 2025