Electronic Design

ALAN R. EARLS, Contributing Editor

Backward and Forward Compatibility for Process Systems

Planning and thoughtful design can yield process systems that bridge old and new technology and contribute to the bottom line for years, and even decades.

Process systems often have very long lifecycles. Not that systems aren't updated from time to time, but unlike other sectors, it isn't all that rare to find a decades-old system still quietly doing its job. This longevity is driven by a range of things including a focus on the bottom line, the cost of interrupting profitable activities to accommodate upgrades, and the fact that many systems were built to last and be adaptable to the future.

To ensure continued backward and forward compatibility in electronic systems, there are a number of areas on which to focus. For example, at the component level, look for those with standardized interfaces, robust data formats, and versioning strategies, while also conducting thorough testing to verify compatibility across different software and hardware versions.

If you were to put this in a graphic illustrating electronic component obsolescence, it would show a component lifecycle. It would move from "Active" to "Not Recommended for New Designs (NRND)," then "Last Time Buy (LTB)," and finally "Obsolete," highlighting the stages where components become increasingly difficult or impossible to obtain or use.

What is the Role of Standardized Interfaces and Formats in Achieving Longevity?

One simple place to start is by looking at pinouts and protocols. Look for components that adhere to industry-standard pinouts and communication protocols (e.g., kSPI, I2C, UART). Wi-Fi, for example, has been made as compatible as possible through an active support organization. This ensures that components from different manufacturers can interoperate seamlessly.

Likewise, at the software/firmware level, look for products that use well-defined and widely supported data formats for communication and storage. This makes it easier to maintain compatibility across different software versions and hardware platforms.

Components that utilize open standards and protocols can also provide some insurance against obsolescence, as these are more likely to be supported across different systems and platforms. One other area of compatibility that deserves attention is EMI. Forward iterations usually must maintain the same EMI profile or may even be required to meet stricter standards. Be sure to consider this issue.

How Can Versioning Control Help to Achieve Longevity?

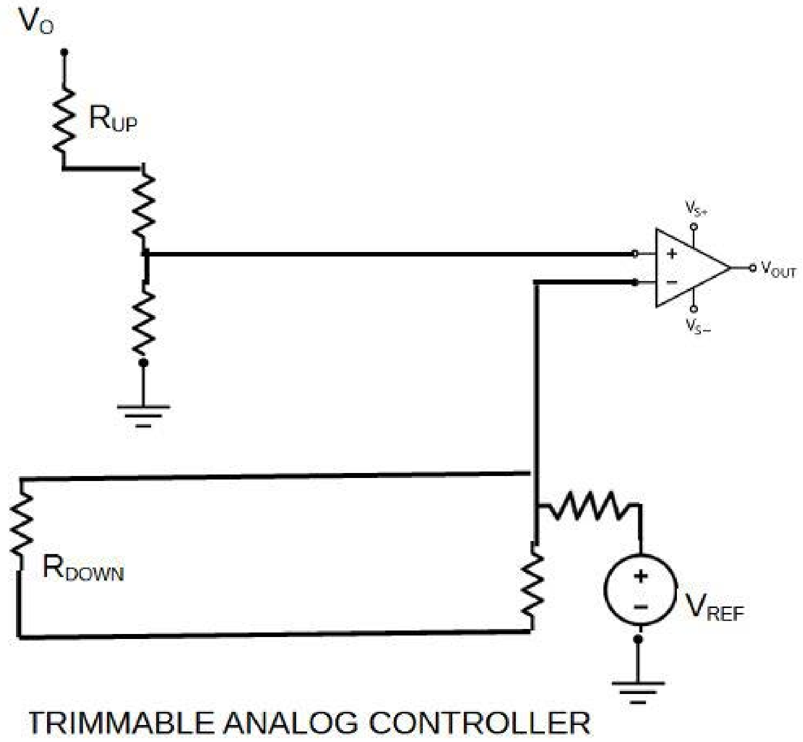
Software and firmware ought to be the easiest things to deal with, but, in fact, they often depend on a value chain of vendors and hidden design features that may make them easier or harder to adapt to changing technology environments. And, of course, sometimes vendors simply decide to stop supporting a given technology.

For those reasons, it's worthwhile implementing robust versioning strategies for both software and firmware. This requires research into product roadmaps and published plans as well as some analysis of how likely the roadmaps and plans are to be implemented. The strategy should allow for future development and updates and upgrades without inhibiting compatibility with older versions.

It's also prudent to design both hardware and data structures to be extensible, thus making it possible to add new features and functionalities without requiring a complete overhaul of the system. Both can present challenges, though.

Hardware is particularly difficult because fundamental design needs are often challenging enough without having to make allowance for comparability with older systems. And looking into a crystal ball to try to see what features might be needed in the future is no easy task. Data structures also matter, but they probably fall into roadmap discussions.

For so many systems, the application programming interfaces (APIs) are quite literally the key to making elements successfully plug and play. If possible, determine whether the APIs you're considering are designed to be backward compatible, allowing new software versions to work with older components and vice versa. Not all APIs are alike.



This controller uses two external resistors, RDOWN and RUP, to adjust the output voltage. RDOWN lowers the output voltage by reducing the signal to the error amplifier's positive input. RUP raises the output voltage by reducing the signal to the error amplifier's negative input. (Credit: Alan Earls)

Why is Having the Best API Important to Longevity?

A poorly chosen API can lead to significant challenges down the line, impacting development time, cost, performance, and even the security of your application. The most fundamental challenge is ensuring the API provides the necessary functionality and data to achieve your application's goals. Carefully analyze the API's endpoints, methods, and data structures to see if they align with your needs.

Does the API offer all of the features you need, or will you have to compromise or build additional functionality yourself? Are the features offered relevant to your specific use case? What data formats are used by the API (e.g., JSON, XML)? Are these formats compatible with your existing systems and programming languages? Data mapping and normalization can be time-consuming if there are significant discrepancies.

How does the API provider handle updates and changes? Proper versioning ensures backward compatibility and enables you to migrate to new versions at your own pace. To what extent are new versions of the API backward compatible with older versions? Breaking changes can require significant code modifications in your application. Finally, it's important to understand the API provider's policy on deprecating older versions. How much notice will you receive before a version is retired?

How Do Hardware Design Choices Impact Longevity?

Keeping design choices in mind can be very important for longevity. For example, while digital power management is taking over from traditional analog controllers to work smoothly in systems with both old (analog) and new (digital) parts, the digital systems need to keep some of the old features.

One common feature is output-voltage trimming. This lets users adjust the output voltage of analog power modules. Analog systems can do this with a simple external resistor *(see figure)*. This is really useful because it means a small number of standard power modules can be used in many different applications, even if they need different voltages Or more to the point, they can be adapted for changing technology needs.

How Does Testing Help Ensure Longevity?

When additions or changes are made, there's value in rigorously testing the system to ensure that new software and hardware versions can still function correctly with older components and versions. Likewise, it's good practice to evaluate—to the extent practical—the system's ability to work with future software and hardware versions, identifying potential compatibility issues early on.

Environmental testing should include factors such as temperature, vibration, etc., to ensure that components can withstand real-world conditions and maintain compatibility over time. This could potentially include accelerated aging tests involving more severe conditions than are in fact likely to be encountered in service.

Why are Component Selection Considerations So Important to Ensure Longevity?

When it comes to longevity, the saying, "a chain is only as strong as its weakest link," applies in spades. Reliability and quality of all components is necessary to ensure long service life of the whole. That doesn't mean a design needs to be "gold plated." But component selection should be based on evidence.

Usually, choosing components from reputable manufacturers with a proven track record of reliability and quality is a very good start. But don't let reputation alone be a guide. Carefully review component datasheets and documentation to understand their specifications, limitations, and compatibility requirements. Ensure that components comply with relevant industry standards and regulations (e.g., RoHS, IPC).

Dig into manufacturer reliability. Are there any indications in available literature of subpar performance? Has the manufacturer or its product line been the subject of legal action? Returning to earlier points, look for suppliers with ample datasheets, application notes, and reliability data.

Is Longevity Improvement Possible?

Yes, as noted above, many approaches can be taken to improve the service longevity of systems. Designing in the feature that will keep systems running well over the long term while also ensuring they're capable of taking on upgrades means designing for an additional dimension: time.

And long-lasting systems save downtime, reduce capital expenditures, and free up engineering resources for higher value-add activities.

References

Forward/Backward Compatibility application report
Backward Compatibility