



Building a software-defined ultrasound platform

Lessons in flexibility,
scalability and system design



Developing new ultrasound systems has long been constrained by the rigidity of existing hardware platforms, with each new concept often demanding a complete redesign, which slows progress and limits innovation.

Eight years ago, TTP began a collaborative project to address this challenge, creating a reconfigurable, software-defined ultrasound (SDU) platform that could adapt to a wide range of applications. We share the technical journey behind that work: the key challenges and trade-offs we faced, the engineering decisions and solutions that emerged, and the lessons that continue to shape our approach to ultrasound system development.





Contents

- Introduction 4
- Developing the SDU concept 5
 - Setting up a modular system for channel flexibility 6
 - Achieving precise control by robust inter-module synchronisation 8
 - Enabling architecture versatility for complex probe arrangements 9
 - Maximising performance through persistent troubleshooting 10
 - Optimising usability through understanding user perspectives 12
 - Understanding component limits through real-world system integration 12
 - Building up capability by iterated design 13
 - Productisation through EMC-ready hardware and embedded flexibility 13
- Reflections on our journey 14
- Conclusion: Turning concepts into reality 16
- About TTP's Imaging team 18
- About the authors 19

Introduction

Ultrasound has become a powerful tool for medical diagnostics, but as systems become more advanced, so do the challenges faced by start-ups or researchers looking to commercialise a novel ultrasound concept. This is largely down to the following reasons:

- **Limited access** – Off-the-shelf ultrasound systems rarely provide transparency into design decisions or access to the underlying architecture, limiting the ability to use them as a starting point.
- **Integration difficulties** – Most commercial ultrasound systems are not flexible enough to allow easy integration of new hardware or software ‘inside the box’.
- **Limited collaboration culture** – Traditional ultrasound vendors have not typically focused on R&D partnerships, particularly those using unconventional imaging strategies.
- **Economic barriers** – Many ultrasound manufacturers prioritise selling large numbers of finished systems, with limited scope to back novel concepts with large development costs and uncertain rewards.

One answer to this is to build your own system from the basic components. But to make software development feasible, you need first to have defined your hardware setup. This reduces the flexibility of the whole design, making it more difficult to balance trade-offs in reliability and performance, and increasing the risk of numerous frustrating iterations before reaching a working prototype.

Building on our experiences developing such systems, TTP recognised the need for a platform to address these challenges and set out, with our collaborators at Smart Acoustics, to build a software-defined ultrasound (SDU) system.

Working together, we set out to turn the initial idea into a modular, scalable platform that would make it much easier for development teams to turn their ideas into a manufacturable product design. The journey hasn’t been easy, but it has provided valuable lessons in ultrasound architecture, modular design, and the realities of building flexibility into complex systems – insights we share below.

Developing the SDU concept

One of the reasons we were particularly interested in this project was that it had the potential to address a challenge that we'd encountered with ultrasound projects for other clients – the need for a full hardware redesign, even when the underlying signal-processing principles were similar.

Separately, we realised that the architecture of a system we had previously created for a particular project could, if parametrised, be scaled arbitrarily to however many channels were required. We termed this concept software-defined ultrasound (SDU), by analogy to software-defined radio, whereby much of the signal processing is done by a general-purpose processor, rather than being performed in dedicated electronic circuits.

The idea behind SDU was to replace fixed, hardware-bound architectures with a system that was re-programmable, scalable, and open to experimentation.

This SDU concept, we realised, hinged on two principal features:

- A modular system design to provide the necessary flexibility for a wide range of applications
- An application programming interface (API) to provide a control mechanism for these transducers, whatever their configuration.

With experience from multiple ultrasound projects, we were confident in our combined expertise in hardware design, signal processing and imaging, as well as in our understanding of the need to develop a system with these features at the core.

Transforming the SDU concept into a practical, scalable platform ready for real-world use proved more complex than expected, with each engineering problem solved revealing further layers of complexity.

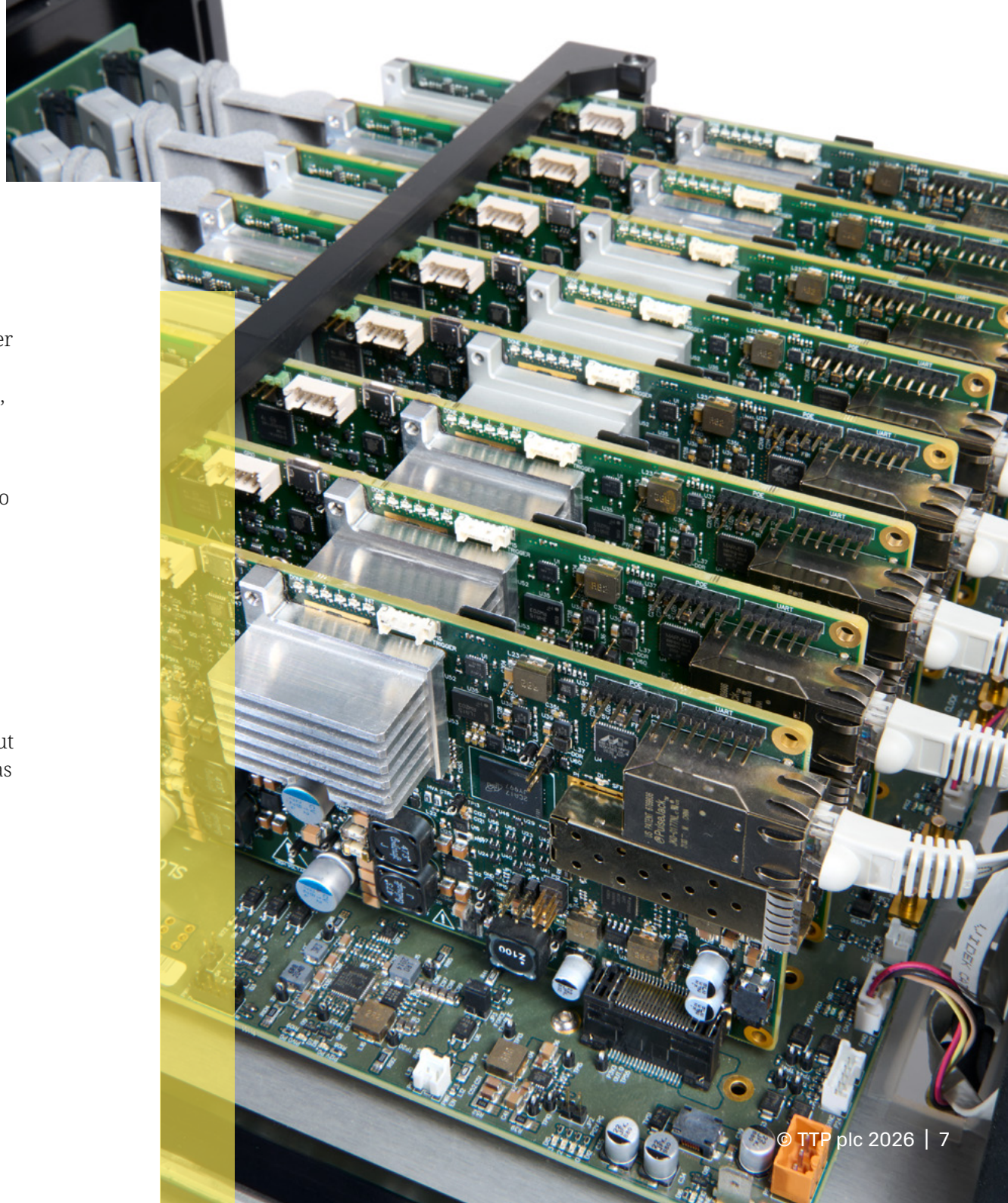
And yet this intensive engineering journey has proven valuable, pushing us to innovate and work at the boundary of what's possible. The result is a system that now represents the state of the art. To understand how it was achieved, we examine eight of the key challenges we faced in developing a truly modular SDU.



Setting up a modular system for channel flexibility

To realise our vision for a truly versatile ultrasound platform, we first needed to move away from a system where standard 128- or 256-channel modules are bought in as ‘fixed’ components of the design. These days, engineering teams want versatility in how they drive transducers and receive data – for example using fully-addressable systems to explore new beamforms using custom probes, carry out plane-wave imaging, or use AI algorithms on the raw (rather than post-processed) data.

We realised that this versatility would be useful not just from one project to another, but even within projects. For example, if you start off implementing your concept on a monolithic 128-channel system, but then realise that there are benefits to expanding it to 256 channels, you don’t want to have to redesign your whole system from scratch. Similarly, you might start at the standard 128, but be interested in how the imaging performance changes if you scale-down to 64 channels.



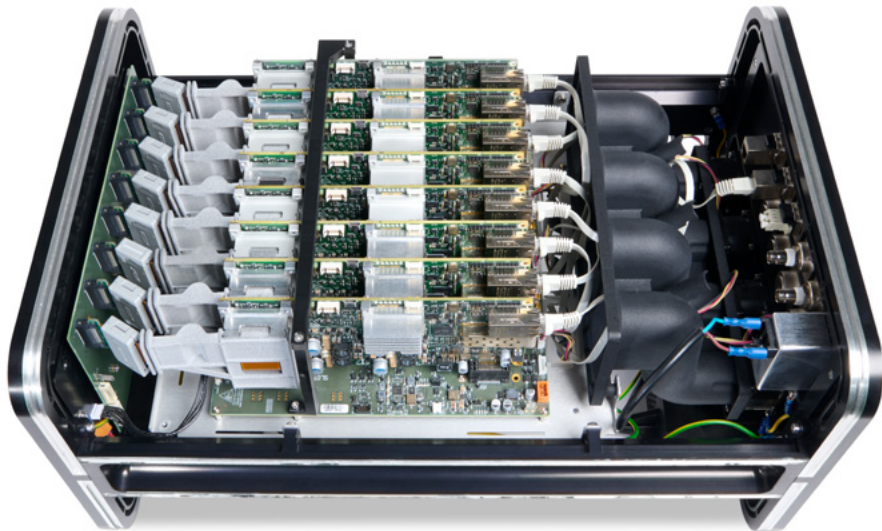
This led us towards devising a truly modular setup based on a ‘stackable’ 16-channel unit, with a core invariant design fitted into an architecture that can scale almost arbitrarily and completely transparently. In this way, it becomes possible to expand the number of channels easily, from 16 through the standard 128 up to 512 or even more – yet still drive them as a single system. At the same time, the system’s flexibility extends to the data, because it is defined by software, not baked-into the hardware.

The question then became, how could we integrate these modules so that they all work together in unison? We first needed to establish a group-based architecture that enabled modules to be added easily, and that allowed each one to ‘know’ where it sat in the system. When doing this, we made sure to write the software so that we weren’t embedding assumptions about the size of the system. That was straightforward enough, but the second aspect presented more of a challenge – synchronising the clocks.

This highlighted that scalability is not just about adding channels, but about designing architectures that remain coherent and adaptable as system size changes.

Achieving precise control by robust inter-module synchronisation

For medical ultrasound systems to function properly, it's vital that all the components are perfectly synchronised with each other using a carefully considered distribution of clocks, and that communications are tightly controlled. This need also extends to every component within the SDU, not least because any mistimings between modules would result in an unusable ultrasound beam. Accordingly, we needed to ensure that both the pulses sent out and the data received back were aligned to a known reference point.



Achieving this for our modular system was not straightforward, because of the independence of the individual units. But we realised that delegating the synchronisation protocol to the baseboard (or indeed, a level higher) meant that communication between the modules could be reduced to the bare minimum (e.g., information on the board number, or which modules are present). This in turn meant that the presence or absence of a module would be irrelevant to other modules, allowing the whole system to be easily scaled up.

We hard-coded that into the system, and we found it worked very well – but the very fact that it couldn't be changed was a barrier to achieving a truly versatile SDU. So, we rapidly moved towards a next generation, in which we placed a field-programmable gate array (FPGA) on the baseboard, enabling us to easily reprogram how the boards were 'talking' to one another. This had benefits for our SDU by opening up another dimension of flexibility – specifically, enabling multiple baseboards to be connected together, and allowing more complex but precise synchronisations both within the SDU and with any external function that a client might need to build onto it. But it still wasn't quite good enough.

Nevertheless, the experience reinforced that early synchronisation decisions can either enable or constrain future flexibility, making them a critical architectural choice rather than an implementation detail.

Enabling architecture versatility for complex probe arrangements

Although we'd incorporated an FPGA on our SDU baseboard to achieve synchronisation, on another project (a 128-channel setup) we needed to allow 'unconventional' imaging strategies. This meant having control over individual channels within an acquisition run, in order to build architectures that weren't based solely on the 16-channel base module. This pointed towards the need for a software-controlled switch array, with a switch on each channel – but, frustratingly, the baseboard we were using didn't have that facility.

So rather than upend our SDU design and use a different baseboard, we reconfigured the system so that Board 0 controlled the switch array, but with the signals now routed via the baseboard, in order to ensure synchronisation – which worked very nicely. We also adjusted the API so that it could easily accommodate an arbitrarily complex arrangement of probes – flexibility that we believed would prove very useful for future projects.

An alternative strategy that we've developed for a recent project involves interfacing the SDU with an external block, which uses the synchronisation signals from the baseboard to control when acquisitions happen. Through the use of 'flags' raised at the start of an acquisition and lowered at the end, it also allows progress to be monitored by the external block without it having to assimilate the detail of what's going on.

Although this modified system did require some extra software to ensure it was robust, it's simple and works beautifully for this particular application, allowing more channel control without committing to a fully bespoke system. This was useful because when a client needed to demonstrate that their innovative ultrasound imaging concept was going to work in practice, before devoting resources to refining the architecture.

Nevertheless, for the time being, we're using the 'Board 0' solution within our SDU, because it does virtually everything that we need it to. But it's still useful to have the 'external block' solution if needed, and in the future, we plan either to integrate it completely, or revisit the possibility of using a new baseboard with greater channel-control capability.



Maximising performance through persistent troubleshooting

Across the journey, the biggest challenge that we faced on an everyday basis was noise performance. At the beginning, signal-to-noise ratios were poor, but they have steadily improved, largely due to incremental changes to the hardware.

One of the largest sources of noise was the presence of non-obvious hardware-specific ground loops, which were a particular problem because of the modular setup. These were easy enough to resolve by adding ferrites to the flexi-connectors – in many ways a trivial solution, but one that made a large difference.

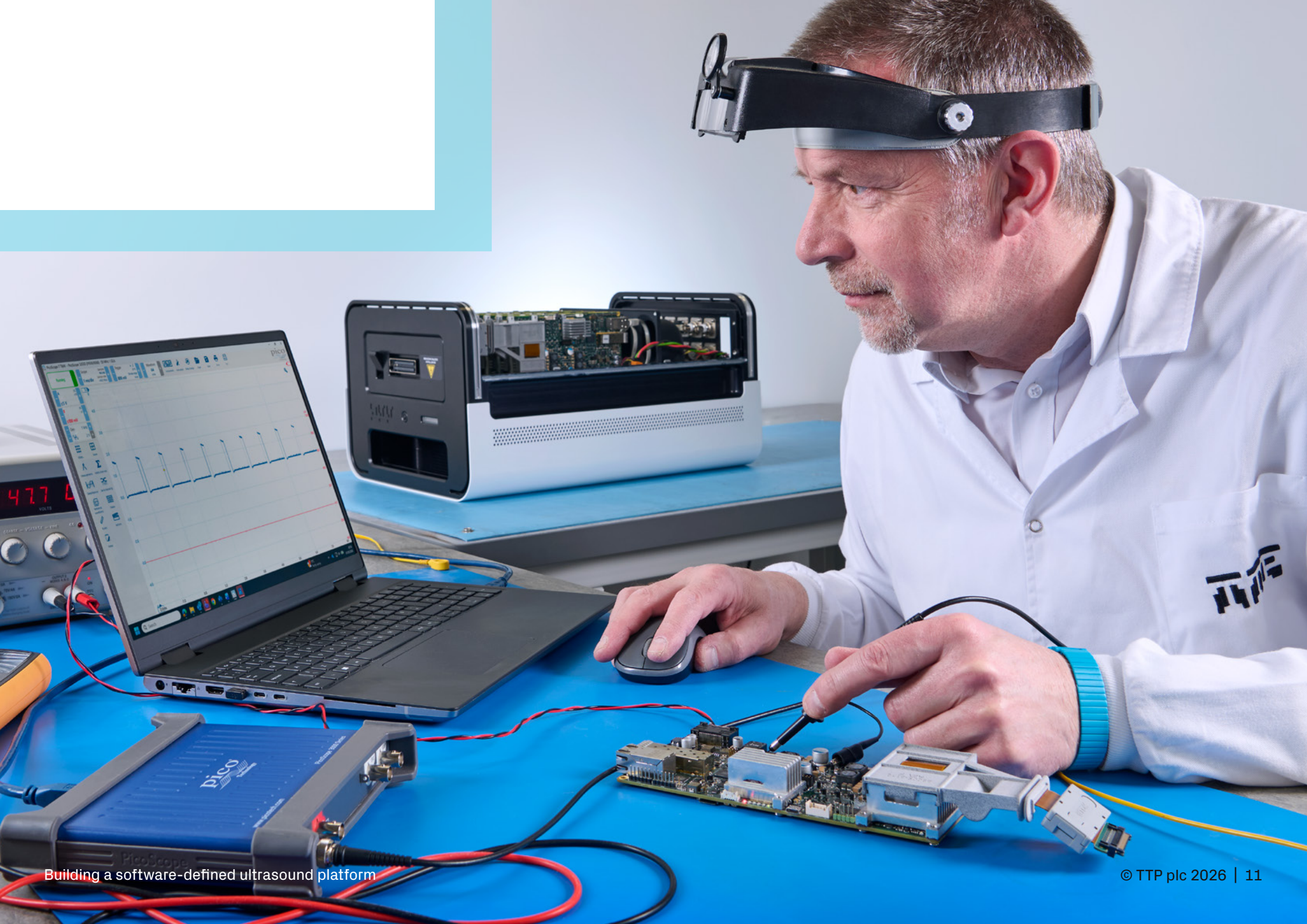
The second recurring challenge was occasional software bugs that only became obvious during ‘soak testing’. So, for example, at one point we found that if we power-cycled the system five times in a row, it would crash once in five runs. When we went through the code, we’d often find not one but several bugs, which we then fixed.

When we repeated the test, we’d find that we hadn’t solved it – instead, it would now crash one time out of 10, and for a different reason. And although each round of troubleshooting and bug-fixing would lower that ratio, it rarely eliminated the problem

completely. But by carrying out extensive automated tests of these subtle interactions, within two weeks we got to the point where we no longer saw these issues arising.

Such rigorous verification testing is especially important for the SDU, because although in a lab scenario such occasional software bugs can be tolerated, for medical devices the tolerances are much lower – for example, failure to start would be tolerated no more than one time in 1000. Our diligence in addressing such ‘deep integration’ issues early on means fewer problems for our clients when they incorporate the SDU into their end-products.

These alternative options form part of the available solutions we can use to solve emerging problems, with the option to further integrate those as the system evolves.



Optimising usability through understanding user perspectives

Ensuring that the hardware is reliable is one aspect of developing a medical ultrasound system – and so is ensuring that the software is carefully tailored to the needs of the user. For the SDU project, this came naturally from the outset, because we were the users – we were motivated by our own research needs to make the software as good as it could be.

In particular, we needed the API to allow flexibility for the user, but not force flexibility on them, because that would make their life unnecessarily complicated (the ‘one size fits all’ fallacy). This was not an easy task, but getting the SDU up and running for a new project now only requires a small number of Python functions, alongside a clear and well-defined JSON configuration. This makes it easy for the client’s programmer to get started, and allows them to build in complexity if they need it.

It’s that sort of practical, real-world usability that is always at the forefront of our minds in this development – whether the user is a clinician, a patient, or (as here) an ultrasound engineer. For the SDU, this plays out in the design of the software interface itself, but more fundamentally when considering the structure of the API; had we not taken care to understand how it would be used, we could easily have ended up making a decision that made an entire use case impossible, even though the decision might have seemed perfectly reasonable at the time.

Understanding component limits through real-world system integration

This focus on optimisation brings us back to hardware. In any ultrasound project, you’re never entirely going to escape the need for some hardware tweaks, but one thing we hadn’t foreseen were the relatively fundamental hardware issues arising from some of the commercially available components we were buying. For example, if we drove all the channels simultaneously, this generated a large peak current that caused a significant amount of internal noise in the system, upsetting its operation.

We realised that such problems mostly arose not because of a quality issue with the components. Rather, it was either because we were using the components at the limits of their specifications, or because we were using them together in ways that the designers had never anticipated.

To an extent, this comes with the territory of exploring novel ideas, and so we just had to work through these difficulties to find a solution. But it also had an upside, because it led to conversations with the manufacturers, which in turn led to a better understanding of the components. It took time to build such relationships, but we’ve found them to be very rewarding - both in this project and more generally.

Building up capability by iterated design

Despite the upfront engineering work, new projects keep challenging us to extend the capability of the platform. On several occasions, we thought we'd created the 'ultimate' ultrasound system, and then the next project would prove that in fact, we hadn't, and that we needed to refine the SDU yet further.

Each time we have adapted the software architecture of the SDU in response to a client's need, we have ensured that we do so in such a way that built up its capability, rather than making a one-off bespoke modification that wouldn't have value outside that specific application.

By iterating the design in this way, we've steadily improved our ability to solve new problems. So, the solution to challenges we face today is an inherently better solution than would have been possible in previous years – it's been an evolutionary process.

Productisation through EMC-ready hardware and embedded flexibility

The result of this evolution is that the SDU is now a highly flexible system that's easy to optimise for a specific need. This allows design changes to be evaluated earlier and with greater confidence, reducing late-stage rework and shortening the path to a stable, test-ready system.

This is particularly important in medical device development, where regulatory and EMC requirements impose significant constraints. Modularity enables teams to explore architectural options early, before committing to a fixed design - helping avoid the costly redesigns that often emerge later in development.

Because regulatory, manufacturability, and usability considerations were addressed early in the design, systems developed from the SDU reference architecture can transition more smoothly from prototype to compliant product.

Addressing these constraints early proved essential in avoiding the costly late-stage redesigns that all too often derail complex medical device programmes.



Reflections on our journey

Developing the SDU to this stage has occupied our specialists for eight years, and has required far more than just software or hardware expertise alone. It has demanded a full systems approach – involving careful planning, requiring multiple disciplines and teams to adjust the architecture and improve system versatility at each stage. As a result, it has needed long-term persistence and careful iteration to overcome the numerous technical obstacles.

The outcome is a platform that can genuinely accelerate ultrasound innovation. By designing for flexibility from the outset, we've created a system that allows developers to explore, test and refine new imaging approaches far more efficiently than before. Not only that, but the SDU's modular architecture opens the door to applications well beyond imaging, supporting the next wave of ultrasound-based diagnostics and therapeutic devices.

Further improvements are being planned – including handling high-speed data rates, upgrading the capabilities of the baseboard, and more. And although the SDU was developed with ultrasound imaging in mind, its versatility means its architecture is adaptable to therapeutic ultrasound and other modalities, as well as potential non-medical applications.

Keeping the project in-house has allowed us to explore ideas with potentially game-changing (but uncertain) applicability. This has meant we haven't been solely driven by funding criteria or narrow commercial imperatives, which would otherwise have required a target application to be clearly identified upfront, constraining creativity and ingenuity. That freedom to investigate ambitious, high-risk concepts has been central to the SDU's progress, and to the culture of innovation that continues to drive it forward.



Conclusion: Turning concepts into reality

In collaborating with Smart Acoustics on the development of the SDU, we've drawn on deep expertise in imaging, electronics, software and product design, combining this with curiosity and persistence to achieve the goal of creating technology that performs reliably in the real world.

This approach underpins every complex technical challenge we take on. Whether the objective is to prove a novel imaging concept or accelerate the development of a manufacturable clinical system, we bring the experience and perspective needed to make that journey more predictable and the final product more effective.

We hope that this article has useful insight for those pursuing the next generation of ultrasound systems. If you would like to explore how your own concept could be developed into a practical system, we would be pleased to start that conversation.

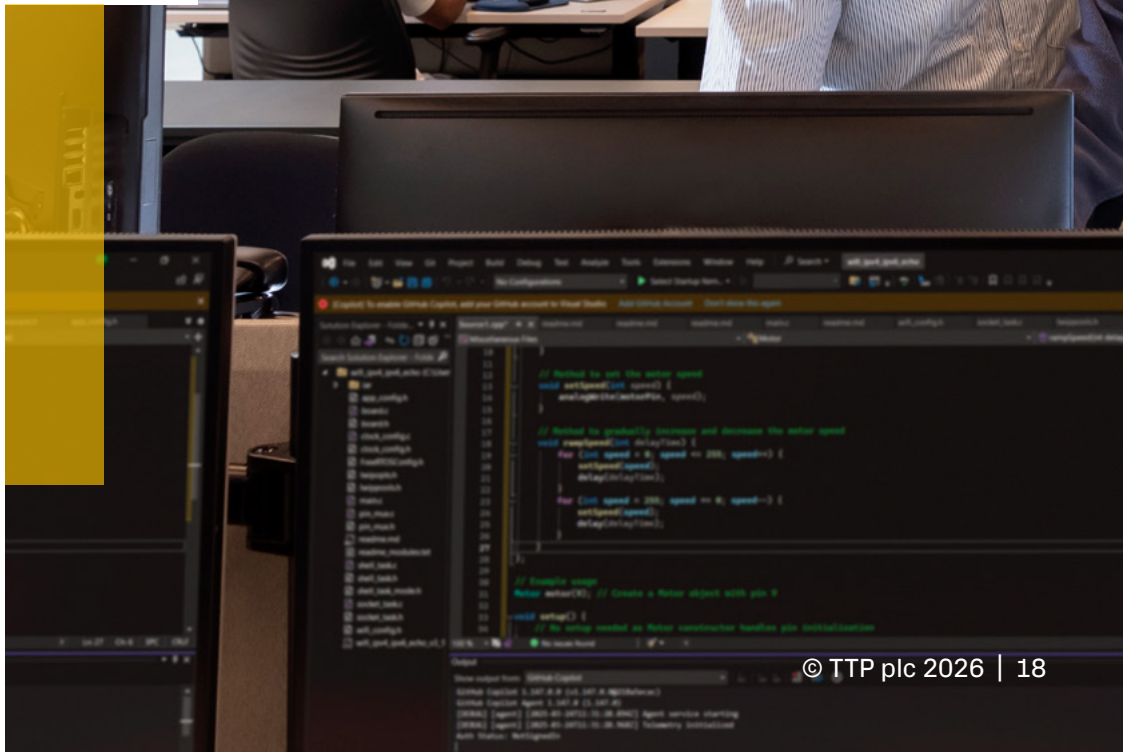


About TTP's Imaging team

TTP's medical imaging team develops advanced imaging systems that turn complex sensing technologies into practical clinical tools. The team works across modalities including ultrasound, optical imaging and sensing-based guidance, combining expertise in electronics, signal processing, software and system engineering. From early concept exploration to clinically viable systems, we help innovators solve the technical challenges involved in delivering reliable, real-time insight in demanding medical environments. With nearly 40 years of experience in complex system development, TTP helps transform ambitious imaging ideas into robust technologies that perform reliably in the real world.

Our expertise spans the development of:

- Transducers
- Hardware
- Firmware
- Imaging pipeline





About the authors



Jamie Collin

TTP Device Innovation Lead: Imaging

With over 20 years of experience in biomedical ultrasonics, Jamie specialises in metrology, therapy, and imaging applications. Jamie has spent the past 10 years consulting on medical product development, bringing extensive technical expertise and insight to cutting-edge healthcare solutions.



Henry Gomersall

Co-Founder: SmartAcoustics

Henry has dedicated the last 20 years to improving the state of software for ultrasound systems. With his strong domain knowledge, as part of SmartAcoustics he has been integral to developing the full firmware and software stack for that drives the SDU and other systems



TTP plc

TTP Campus, Cambridge Road,
SG8 6HQ

+44 1763 262626

ttp.com