

Managing the multimode architecture evolution

Pascal Herczog

03/07/2009

Cognovo Ltd
Melbourn Science Park
Melbourn, Royston
Herts. SG8 6HB
UK

The Architecture Evolution

Cellular wireless technology has, from the inception of 2nd generation GSM, evolved at an enormous pace in terms of features and performance through multi-band, GPRS, EDGE and onto 3rd generation UMTS. The silicon technology and design methodologies have struggled to stay ahead of this wave, leading to a succession of architectures and long cycles of re-design. Initially, architectures were hardware centric to cope with the data processing requirements, but became more software centric moving to EDGE as silicon and DSP technologies improved to take on receive path processing at a viable level of power consumption. This software architecture gave baseband designs the opportunity to rapidly extend modem capabilities such as higher multi-slot classes and DTM, improve on receiver performance and cope with a then expanding range of commercial RF designs.

But with the advent of Release99 3G technologies, the data processing requirements were once again out of reach of standard DSP architectures, and around 1.5million gates of dedicated hardware logic plus associated memories were needed to perform the rake receiver, combining and channel decoding tasks. The value of the DSP software architecture was not diminished though, as it continued to provide the critical link in controlling the dedicated hardware and closing the loop on algorithms such as multi-path selection, AFC, AGC etc.

The introduction of HSDPA, new equaliser designs and parallel operation with R99 modes required another extension to the amount of dedicated hardware, scaling to around a million gates and associated memories for category 10 HSDPA, and a further extension for HSUPA operation, as well as increases in the performance and memory requirements of the CPU and DSP processors. Indeed, some architectures use multiple DSP units to distribute the processing load.

At this point then, the classic modem architecture had evolved to that as shown in Figure 1.

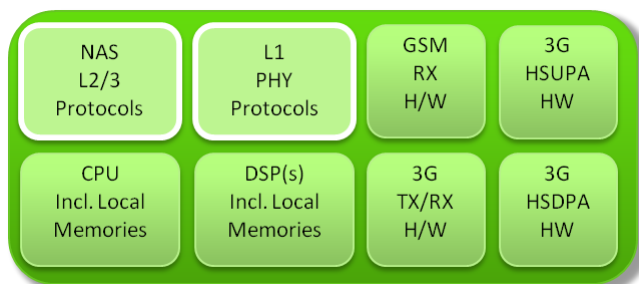


Figure 1 Classic Modem Architecture

Now, the architecture has to change again, as the requirements for HSPA+ such as Higher Order Modulation, MIMO and improved receiver performance are being demanded by operators to increase their network efficiency and extend the lifetime of their 3G network investments. Further dedicated hardware accelerators are required to cope with these demands.

With the introduction of LTE, the processing requirements of OFDM are once again incompatible with the existing hardware capabilities of the modem, having to provide multiple high throughput FFT/iFFT operations, FIR and MIMO detection processing capabilities. It is estimated that a hardware implementation of the LTE physical layer is of the order of 3 - 4million gates, with associated high

speed on-chip data memories scaling to a Megabyte for a 50Mbps capability class. Even on 45nm silicon technology, that represents an addition of around 10mm² dedicated to LTE, even if the existing 3G/GSM CPU and DSP can be re-used.

The SDR Architecture

Software Defined Radio (SDR) is of course not a new concept. It has already been successfully deployed in base station architectures, as well as other wireless fields such as military applications. However, power consumption and die size of these solutions have to date not been competitive compared to the dedicated hardware solutions used in handsets. However, a number of processor technology providers have now matured their SDR engine technologies, including the associated tools, to the point where the efficiency of physical layer execution is within 20% of dedicated hardware, whilst giving the significant benefit of re-configurability (noting that any amount of re-configurability in dedicated hardware functions comes at a significant gate and validation cost).

Whilst the SDR architecture for mobile wireless modems is now viable, the roadmap to a full SDR solution for multi-mode is likely to occur in several stages, since it will be difficult to simply dispense with existing, tested HSPA solutions. At first, configurable SIMD processing can be an ideal method to upgrade to HSPA+, providing the processing resource for higher order demodulation and MIMO equalisation. But introduction of LTE provides the opportunity to migrate the whole architecture to SDR, taking advantage of re-usable demodulation and MIMO receiver software techniques.

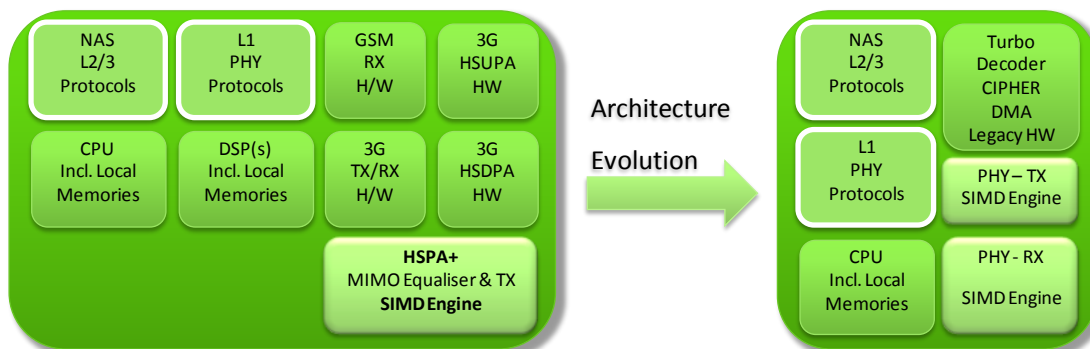


Figure 2 Introduction of the SDR Architecture evolving to LTE

The SDR architecture effectively separates the software from the hardware through the creation of a generic processing platform. Whilst there are still many facets of differentiation at the hardware level, it opens the door to a range of software suppliers to target these generic platforms.

However, integration of software from multiple vendors and interfacing to legacy software for multi-mode are also a potential problems. The Cognovo SDR Framework solution addresses these issues.

A Multimode Software Framework for SDR

An object oriented design approach to the software systems for an SDR platform is the key enabler to greater modularity, better code re-use and the ability to switch protocol components in order to reconfigure the operational mode of a software defined radio. This is achieved by moving away from statically created service access points, which often require updating with each revision of each

protocol element, to active interfaces that are part of the SDR Framework. These allow run-time exchange of protocol objects to enable new features and new standards to be easily integrated.

A fundamental aspect of object oriented design is the encapsulation of state and behaviour into distinct objects, whose generic properties and methods are derived from a common object class. Methods are used to expose behaviour of an object, whilst the object states (modes and data) are managed within the object. This enforced modularity means objects match a strict interface architecture, where new functions can be developed as extensions to existing classes without re-design.

The Cognovo SDR Framework contains interface layer class definitions and methods that provide the means for protocols to interface and transfer data between layers. It also contains layer classes that define the methods associated with each protocol layer, creating the ability to interface to a 3rd party protocol layer without having to understand or know its implementation. New methods can therefore be added without changing the interfaces.

The benefit of this is the ability to support both legacy and new protocol stacks to create multi-mode solutions, as shown in Figure 3.

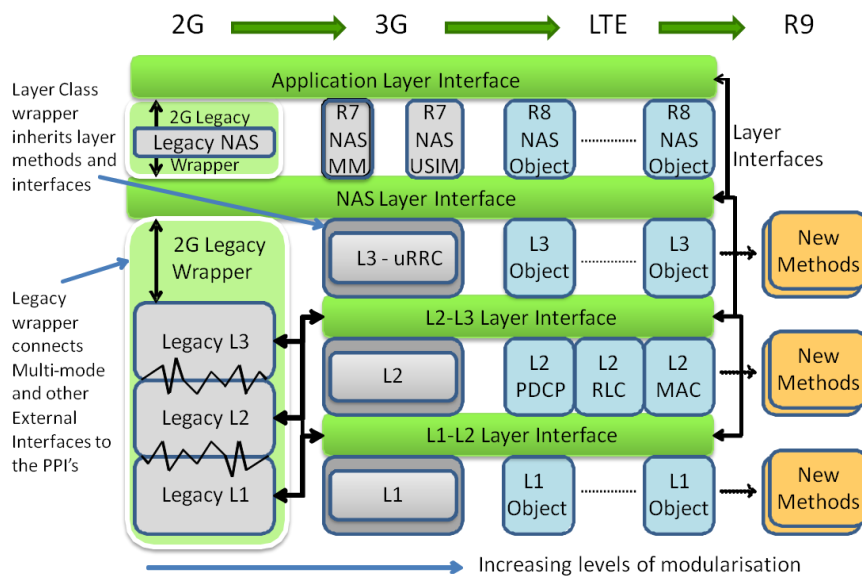


Figure 3 Integration of Protocol Stacks into the Cognovo SDR Framework

The use of active, standardised interfaces and layer classes allows components of the protocol stacks to be switched out for other modes at run time, or upgraded to add for example a Release 8 NAS feature to a Rel7 3G L1/2/3 implementation. However, the sourcing of these components, authentication and validation of the run-time configuration must be formally managed. The Cognovo Configuration and Validation Manager (CVM) executes both the initial and run-time configuration/validation procedures to ensure that the configured modem meets the regulatory requirements for the intended markets through protocol configuration sets.

SDR: A System implementation

To get to an efficient baseband implementation, the SDR architecture must be designed from a system perspective. The system view is shown in Figure 4.

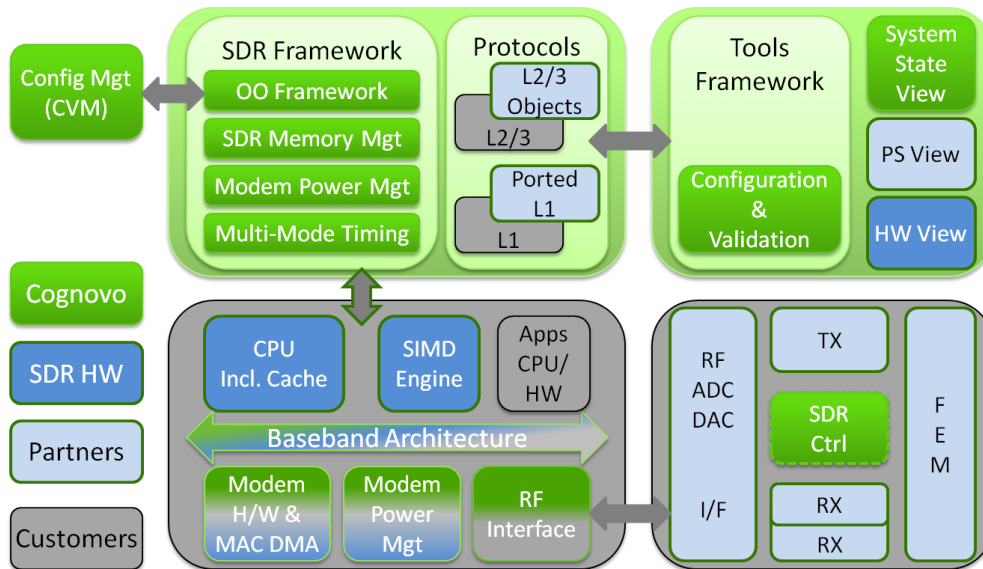


Figure 4 SDR System Architecture View

Dimensioning of the CPU and SIMD hardware is a critical factor in the overall performance, balancing the different requirements of each mode against die size, clock speed and power consumption. The L2/3 CPU requirements for 100Mbps LTE operation are of the order of 150~200MHz, whilst the Layer 1 requirement depends on the SIMD implementation and generic hardware functions such as FFT and Turbo decoding. Re-usable and generic hardware continue to play an important role in balancing MIPS and re-configurability through the use of accelerators for Ciphering, Turbo decoding and flexible DMA scatter/gather constructs applicable to 2G, 3G and 4G operations.

However, an architectural aspect that is critically important, but largely overlooked in traditional designs is the influence of the memory architecture on system performance. Similarly, the software design typically does not take the available types of memory into account, resulting in inefficient usage and incorrectly dimensioned devices leading to higher costs and power consumption as processors struggle to compensate for the lack of required memory performance. With on-chip memory often accounting for more than 65% of the die area, the impact can be significant.

The SDR Framework therefore includes a memory management function which dynamically controls the mapping of code and data depending on the performance and access requirements, onto local caches, tightly coupled memories, system memories and off-chip memory, taking into account any run-time protocol re-configurations and minimising data movement as it is processed through the modem transmit and receive chains.

Power consumption is further optimised by accurate control of system resources. Whilst the modem is responsible for defining changes to its current operating state, the SDR Framework Power Management function collates all system level requirements and maps them onto the available

resources. The clock frequency and voltage transitions for each domain are subsequently handled at the silicon level according to the capabilities of the technology.

Additionally, a reconfigurable system must manage the varied and interacting timing requirements of multiple, active standards. The SDR architecture provides a multimode timing framework that provides the means to translate from the reference time base to that of the serving cell and other standards, and to transfer timing information between protocol components, allowing the use of a single master frequency across all modes.

Finally, a critical element of the architecture is the debug tool chain, where the SDR Framework provides the intercept point for all protocol messaging to provide a system state view – a high level view of the triggers as they propagate through the system. Integration of 3rd party protocol stack debug tools allows the internal decision processes to be put into context of the whole system state, whilst alignment with, and integrated display of the hardware debug flow provides access to detailed performance profiling and resource management, enabling the optimisation of clock speeds, memory usage and power consumption.

SDR in RF

RF technology has been slow to move towards software defined operation due to the power consumption penalty of high speed, linear wideband ADC converters. Additionally, the RF environment places even tighter constraints on the dimensioning and performance of software defined digital signal processing, where traditional software environment overheads would incur a significant cost penalty. However, the drive towards SDR in the baseband architecture, and the opportunity to re-use that architecture in the RF domain, will lead to cost reduced implementations of the architecture being applied to RF calibration, re-configurable filters and ultimately direct processing of ADC outputs.

Already a number of RF vendors are providing SDR based re-configurable RF devices for mobile wireless products that can be matched to front end modules covering GSM, 3G, LTE and a range of connectivity bands by providing sampling frequencies for narrowband to 20MHz channels, configurable modulator selection and programmable filter characteristics, offset correction, decimation, phase compensation etc. Standardised interfaces are critical in delivering the flexibility to re-configure the RF capabilities at manufacturing time whilst maintaining software compatibility.

Conclusion

Baseband architecture has evolved around hardware centric solutions as a result limitations in DSP, compiler and silicon technology. The lack of functional re-use between successive mobile wireless standards has led to a large, disparate set of hardware functions, each often designed with its own uniquely dimensioned memory sub-system resulting in large areas of silicon being unused for any one mode of operation.

The emergence of highly efficient SIMD architectures and vector compiler technologies from a number of suppliers has demonstrated that the die area and power consumption of a re-configurable CPU/SIMD architecture are now competitive for use in a multi-mode LTE/3G-HSPA/GSM modem. This common methodology can also give significant cost savings across other modem developments.

But a competitive wireless platform can only be achieved if it is designed as a system. This involves correct dimensioning of the hardware resources for all modes of operation, taking modem behaviour into account, and an efficient software environment for code development and execution.

The Cognovo SDR framework provides the tools and environment that allows integration of third party components into a consistent, efficient, testable modem system. It will appeal to companies developing software components, supplying SIMD based processing engines and to OEM's looking to take more control of the modem.