Trends in embedded systems

Opportunities and challenges for System-on-Chip and Networked Embedded Systems technologies in industrial automation

Grant Martin, Richard Zurawski

Courtesy Philips

Advances in process technology and the availability of new design tools are broadening the scope of embedded systems; from being implemented as a set of chips on a board, to a set of modules in an integrated circuit. System-on-Chip (SoC) technology is now being deployed in industrial automation, enabling the creation of complex field-area intelligent devices. This trend is accompanied by the adoption of platform-based design, which facilitates the design and verification of complex SoC through the extensive re-use of hardware and software IP (intellectual property). A further important aspect of the evolution of embedded systems is the trend towards networking of embedded nodes using specialized network technologies, frequently referred to as Networked Embedded Systems (NES).

Embedded system technologies

System-on-Chip (SoC) represents a revolution in integrated circuit (IC) design, enabled by advances in process technology, which allow the integration of the main components and subsystems of an electronic product onto a single chip or integrated chipset [1]. This development has been embraced by designers of complex chips because it permits the highest possible level of integration, resulting in increased performance, reduced power consumption, and advantages in terms of cost and size. These are very important factors in the design process, and the use of SoC is arguably one of the key decisions in developing real-time embedded systems.

SoC can be defined as a complex integrated circuit, or integrated chipset, that combines the main functional elements or subsystems of a complete end product in a single entity. Nowadays, the most challenging SoC designs include at least one programmable processor, and very often a combination of at least one RISC (reduced instruction set computing) control processor and one digital signal processor (DSP). They also include on-chip communications structures – processor bus(es), peripheral bus(es) and sometimes a high-speed system bus. A hierarchy of on-chip memory units, as well as links to off-chip memory, is especially important for SoC processors. For most signal-processing applications, some degree of hardware-based accelerating functional unit is provided, offering higher performance and lower energy consumption. For interfacing to the external world, SoC design includes a number of peripheral processing blocks consisting of analogue components as well as digital interfaces (for example, to system buses at board or backplane level). Future SoC may incorporate MEMS-based (microelectro-mechanical system) sensors and actuators, or chemical processing (lab-on-a-chip) 1.

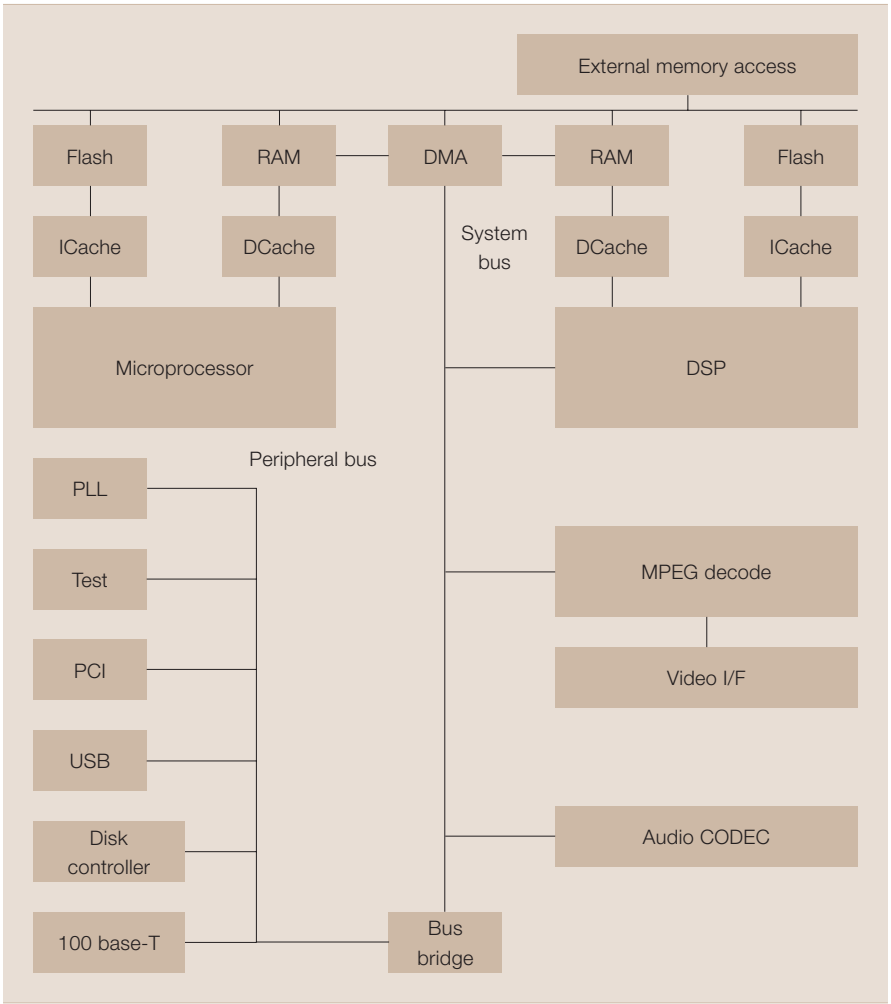
All interesting SoC designs comprise both hardware and software components. These include programmable processors, real-time operating systems, and other elements of hardware-dependent software. Thus, the design and use of SoCs not only concerns hardware – it also involves system-level design and engineering, hardware–software tradeoffs and partitioning, and software architecture, design and implementation.

System-on-a-Programmable-Chip

Recently, the scope of SoC has broadened. From implementations using custom IC, application specific IC (ASIC) or application-specific standard part (ASSP), the approach now includes the design and use of complex reconfigurable logic parts with embedded processors. In addition other application-oriented blocks of intellectual property, such as processors, memories, or special purpose functions bought from third parties are incorporated into unique designs.

These complex FPGAs (Field-Programmable Gate Arrays) are offered by several vendors, including Xilinx (Virtex-II PRO Platform FPGA, Virtex-IV) and Altera (SOPC). The guiding principle behind this approach to SoC is to combine large amounts of reconfigurable logic with embedded RISC processors, in order to enable highly flexible and tailorable combinations of hardware and software processing to be applied to a design problem. Algorithms that contain significant amounts of control logic, plus large quantities of dataflow processing, can be partitioned into the control RISC processor with reconfigurable logic for hardware acceleration. Although the resulting combination does not offer the highest performance, lowest energy consumption, or lowest cost – in comparison with custom IC or ASIC/ASSP implementations of the same functionality – it does offer enormous flexibility in modifying the design in the field, and avoids expensive Non-Recurring Engineering (NRE) costs associated with field changes. Thus, new applications, interfaces and improved algorithms can be downloaded to products already working in the field.

1 A typical SoC device for consumer applications



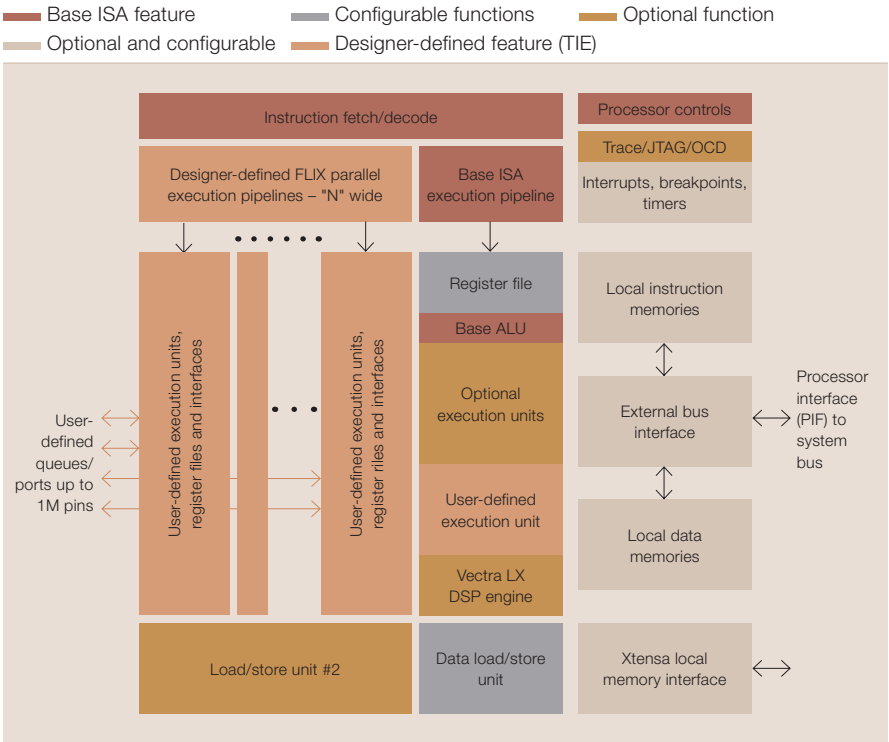
Products in this area also include other processing and interface cores: these consist of multiply–accumulate (MAC) blocks aimed at DSP-type data-flow signal- and image processing applications, and high-speed serial interfaces for wired communications such as SERDES (serializer/de-serializer) blocks. In this sense, system-on-a-programmable-chip SoCs are not exactly application-specific, but not completely generic either.

The guiding principle behind this approach to SoC is to combine large amounts of reconfigurable logic with embedded RISC processors.

It remains to be seen whether system-on-a-programmable chip SoCs will be successful in delivering high-volume consumer applications, or whether they will be restricted to two main areas: Rapid prototyping of designs that will be re-targeted to ASIC or ASSP implementations; and high-end, relatively expensive elements of the communications infrastructure that require in-field flexibility, and which can afford the higher levels of cost and energy consumption in combination with reduced performance.

Intermediate forms, such as the use of metal-programmable gate-array style logic fabrics, together with hard-core processor subsystems and other cores – as provided in the "Structured ASIC" offerings of LSI Logic (RapidChip) and NEC (Instant Silicon Solutions Platform) – represents an intermediate form of SoC between the full-mask approach and the field-programmable gate-array approach. Specific trade-offs here are much slower design creation (a few weeks rather than a day or so); higher non-recurring engineering than FPGA (but much lower than a full set of masks); and better cost, performance and energy consumption than FPGA (perhaps only 15–30 percent worse than an ASIC approach). Further interesting hybrid approaches, such as ASIC/ASSP with on-chip FPGA regions, are also emerging to give design teams more choices. A final interesting variation is a combination of a configurable processor, which is implemented partly in fixed silicon, together with an FPGA region, which is used for instruction extensions and other hardware implementations in the field. Stretch Inc., a semiconductor company, for example, uses the Tensilica configurable processor to implement this type of platform SoC 2.

2 Tensilica's LX processor



Platforms and Programmable Platforms

Recent years have seen a more integrated approach to the design of complex SoC and the re-use of virtual components – this is called "platform-based design" [1, 2]. Platform-based design can be defined as a planned design methodology that reduces the time and effort required – as well as the risk involved – in designing and verifying a complex SoC. This is accomplished by extensive re-use of combinations of hardware [3] and software [4] IP. In contrast to IP re-use in a block-by-block manner, platform-based design assembles groups of components into a re-usable platform architecture. This re-usable architecture, together with libraries of pre-verified and pre-characterized, application-oriented hardware and software virtual components, constitutes a SoC integration platform.

There are several reasons for the growing popularity of the platform approach in industrial design. These include the increase in design productivity, the reduction in risk, the ability to utilize pre-integrated virtual components from other design domains more easily, and the ability to re-use SoC architectures created by experts. Industrial platforms include full application platforms for specific product areas, such as Philips Nexperia and TI OMAP [5], reconfigurable SOPC platforms, and processor-centric platforms. Processor-centric platforms – such as those using multiple Tensilica-configured, extended processors, or ARM PrimeXsys – concentrate on the processor, its required bus architecture and basic sets of peripherals, along with RTOS (real-time operating systems) and basic software drivers.

Platform FPGAs and SOPC devices can be thought of as a "meta-platform"; that is, a platform for creating platforms. These devices contain a basic set of more generic capabilities and IP embedded processors, on-chip buses, special IP blocks such as MACs and SERDES, and a variety of other pre-qualified IP blocks. Design teams can obtain such devices from companies like Xilinx and

Altera, and then customize the meta-platform to their own application space by adding application domain-specific IP libraries. This can then be delivered to derivative design teams.

Networked Embedded Systems

Another important facet of the evolution of embedded systems is the emergence of distributed embedded systems, frequently termed networked embedded systems, where the word "networked" signifies the importance of the networking infrastructure and communication protocol. A networked embedded system is a collection of spatially and functionally distributed embedded nodes, interconnected by means of wireline and/or wireless communication infrastructure and protocols, and interacting with the environment (via sensor/actuator elements) and each other. Within the system, a master node can also be included to coordinate computing and communication, in order to achieve specific objectives.

Controllers embedded in nodes or field devices, such as sensors and actuators typically provide on-chip signal conversion, data and signal processing, and communication functions. The ever-increasing functionality, processing and communication capabilities of controllers have been instrumental in the emergence of a widespread trend for the networking of field devices around specialized networks, frequently referred to as field area networks. (A field area network is normally a digital, two-way, multi-drop communication link [6].) In general, the benefits of using specialized (field area) networks are numerous and include: increased flexibility through combining embedded hardware and software; improved system performance; and ease of system installation, upgrade, and maintenance.

Recent years have seen a more integrated approach to the design of complex SoC and the re-use of virtual components – this is called "platform-based design".

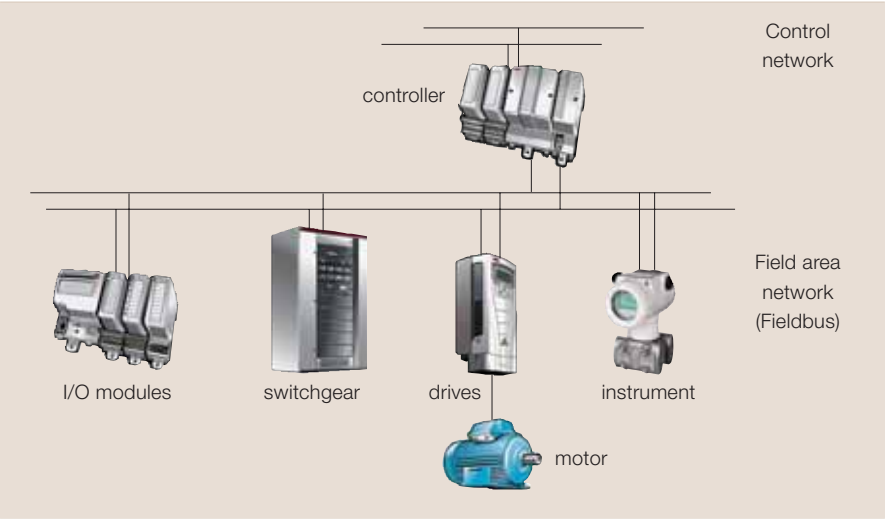
Networked embedded systems are present in a variety of application domains: for example; automotive, train, aircraft, office building and industrial – primarily for monitoring and control. Representative examples of network embedded systems include networks connecting field devices such as sensors and actuators with field controllers; for instance, programmable logic controllers (PLCs) in industrial automation or electronic control units (ECUs) in automotive applications. They are also used in man-machine interfaces; for example, in dashboard displays in cars and SCADA (supervisory control and data acquisition) in industrial automation. The specialized network technologies employed are as diverse as the application areas. For instance: PROFIBUS, PROFInet or EtherNet/IP (both supporting real-time communication) in industrial control and automation; LonWorks, BACnet, and EIB/KNX in building automation and control; CAN, TTP/C and FlexRay in automotive applications; and Train Communication Network (TCN) in train automation. The diversity of requirements imposed by different application domains (soft/hard real-time, safety critical, network topology, and so forth) necessitated a variety of solutions, and the use of different protocols based on different operating principles. This has resulted in a plethora of networks developed for specific application domains [6] 3.

Because of the nature of the communication requirements imposed by applications, field area networks – unlike local area networks (LANs) – tend to have low data rates, small data packets, and typically require real-time capabilities, which may demand deterministic or time-bounded data transfer. However, data rates above 10 Mbit/s, typical of LANs, have become commonplace in field area networks. For field area networks employed in industrial automation (unlike in building automation and control) there is little need for routing functionality or end-to-end control. As a consequence, only layers 1[1)] (physical layer), 2 (data link layer, including implicitly the medium access control layer), and 7 (application layer, which also covers user layer) of the ISO/OSI reference model [7] are used in these networks.

The need to guarantee a deterministic response requires the use of appropriate scheduling schemes, which are frequently implemented in application-domain specific real-time operating systems or custom-designed, "bare-bone", real-time executives.

The networked embedded systems used in safety-critical applications, such as x-by-wire, that adopt mechatronic solutions to replace mechanical or hydraulic solutions with electrical/electronic systems must be highly dependable to ensure a failsafe system. Examples of such embedded

3 Typical field area network architecture in industrial automation



Footnote

[1)] For a brief overview of the OSI model, see figure 1 on page 47.

systems include fly-by-wire in aeroplanes and steer-by-wire in automotive applications, where failure could endanger human life, property, or the environment. To avoid such risks, reliable, failsafe services must be delivered at the request of the system user. The dependability of x-by-wire systems is one of the main requirements, as well as a constraint on the adoption of this kind of system.

Although the use of wireline-based field area networks is most common, wireless technology – including wireline/wireless hybrid solutions – offers incentives in a number of application areas. In industrial automation, for instance, wireless device (sensor/actuator) networks can provide support for mobile operation in the case of mobile robots and the monitoring and control of equipment in hazardous and difficult-to-access environments. A separate category is wireless sensor networks envisaged for monitoring purposes.

Opportunities and challenges in SoC and MPSoC

There are many opportunities arising from the efficient and error-free design of SoC, and in particular Multi-Processor System-on-Chip (MPSoC), which combines the advantages of parallel processing with the high integration capability of SoC. Other areas of interest include testing of embedded cores in SoC, power-aware computing, security in embedded systems, and development of safety-critical systems in the context of x-by-wire and various other applications [8].

Ever-increasing circuit densities and operating frequencies, as well as the use of system-on-chip designs, have resulted in enormous test-data volumes for today's embedded core-based integrated circuits. Reducing data volume and time are two of the main challenges in testing these kinds of circuits. Other problems include: the growing disparity between performance of the design and the automatic test equipment, which makes at-speed testing – particularly of high-speed circuits – a challenge, and results in increasing yield loss; high cost of manually developed functional tests; and a growing cost of high-speed and high-pincount testers.

Growing power dissipation, resulting from the increase in density of integrated circuits and clock frequency, has a direct impact on the cost of packaging and cooling, as well as reliability and lifetime. These and other factors, such as battery-based power supply and device-restricted size (as in the case of hand-held devices), make designing for low power consumption a high priority for embedded systems. The design techniques and methodologies aimed at reducing both static and dynamic power dissipation tend to focus on the following areas: system/application level optimization, which explores task implementations exhibiting different power/energy versus quality-of-service characteristics; energy-efficient processing subsystems like voltage and frequency scaling, dynamic resource scaling, and processor core selection; and energy-efficient memory subsystems, such as cache hierarchy tuning, novel horizontal and vertical cache partitioning schemes, as well as dynamic scaling of memory elements.

Grant Martin

Grant Martin is Chief Scientist at Tensilica, Inc. He received his Bachelor's and Master's Degrees in Mathematics from the University of Waterloo, Canada. He worked at Burroughs in Scotland, BNR/Nortel in Canada, and Cadence Design Systems in San Jose, California, prior to joining Tensilica in 2004.

Richard Zurawski

Richard Zurawski is President of the ISA Group, San Francisco. He held executive positions with San Francisco Bay area companies, Kawasaki Electric, Tokyo, and was a professor at the Institute of Industrial Sciences, University of Tokyo. He is Editor of a book series on Industrial Information Technology, CRC Press/Taylor & Francis. He holds an MSc in Electrical Engineering, and a PhD in Computer Science.

The relatively limited commercial bandwidth resources for computing, memory, and communication of embedded device controllers (eg, field devices in industrial automation) poses considerable challenges for the implementation of effective security policies, which, in general, are resource- demanding. This limits the applicability of the mainstream cryptographic protocols, even vendor-tailored versions. Operating systems running on small-footprint controllers tend to implement essential services only, and do not provide authentication or access control to protect mission- and safety-critical field devices. A growing demand for remote access to process data at factory-floor level may expose automation systems to potential electronic security attacks, which may compromise the integrity of these systems and endanger plant safety. The system/plant availability requirement may render the updating of security software in running field devices impractical or too risky.

Grant Martin
Tensilica, USA
gmartin@tensilica.com

Richard Zurawski
ISA Group, USA
r.zurawski@ieee.org

References

[1] H. Chang, L. Cooke, M. Hunt, G. Martin, A. McNelly, L. Todd: Surviving the SoC Revolution: A Guide to Platform-Based Design. Kluwer Academic Publishers, 1999.

[2] A. Sangiovanni-Vincentelli, G. Martin: Platform-Based Design and Software Design Methodology for Embedded Systems. IEEE Design and Test of Computers 18 (2001) 6, 23–33.

[3] M. Keating, P. Bricaud: Reuse Methodology Manual for System-on-a-Chip Designs. Kluwer Academic Publishers, 1998 (First Edition), 1999 (Second Edition), 2002 (Third Edition).

[4] G. Martin, C. Lennard: Invited CICC paper. Improving Embedded Software Design and Integration for SOCs. Custom Integrated Circuits Conference, May 2000, 101–108.

[5] G. Martin, H. Chang (Editors): Winning the SOC Revolution: Experiences in Real Design. Kluwer Academic Publishers, 2003.

[6] R. Zurawski (ed.): The Industrial Communication Systems, Special Issue. Proceedings of the IEEE, 93 (2005) 6.

[7] Zimmermann H.: OSI Reference Model: The ISO model of architecture for open system interconnection. IEEE Transactions on Communications, 28(4): 425–432, 1980.

[8] R. Zurawski (ed.): Embedded Systems Handbook. Taylor & Francis, 2005.