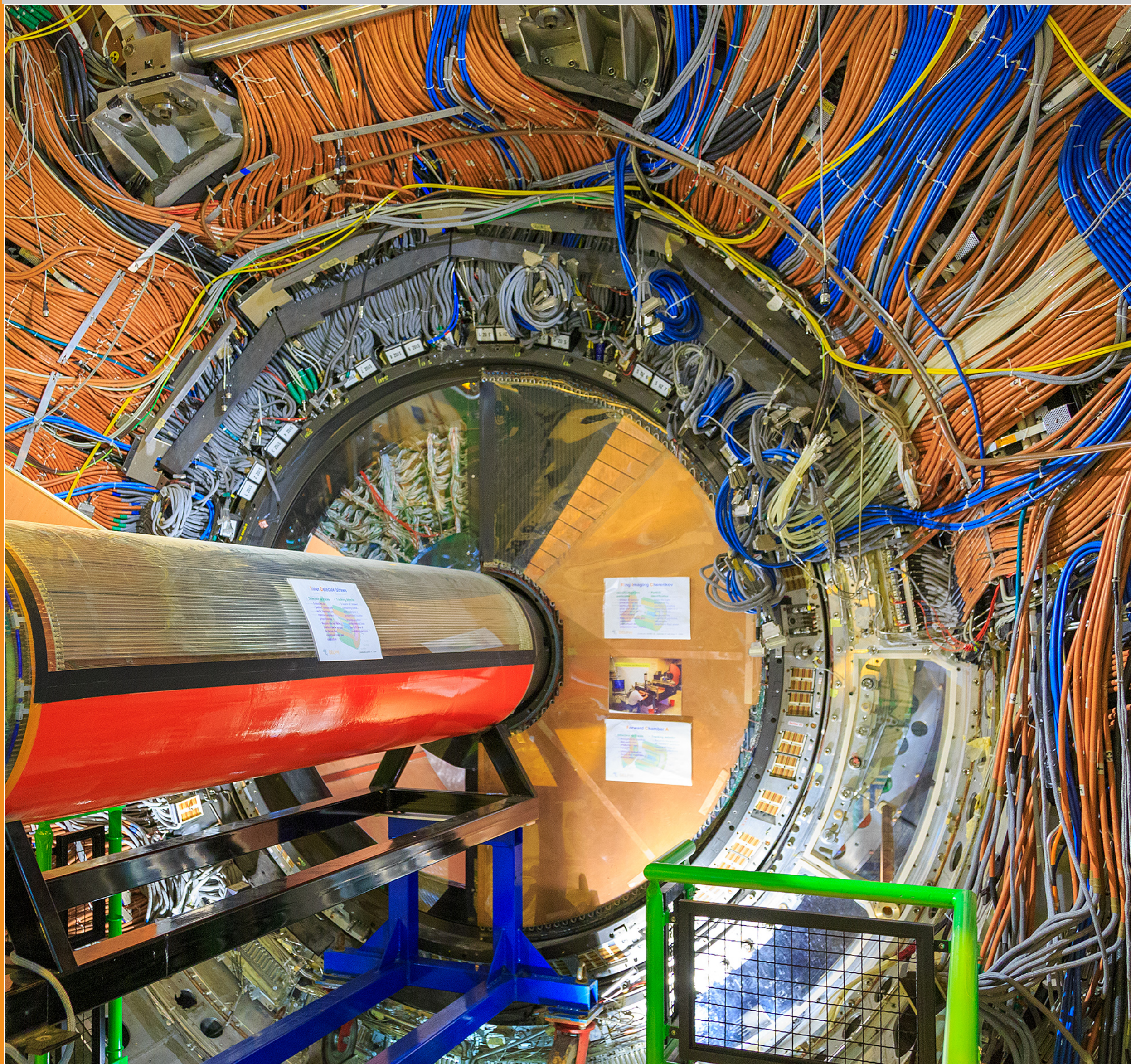
4TU.Embedded Systems

Domain Specific Frame of Reference Embedded Systems

2023



TU Delft Delft University of Technology

TU/e Technische Universiteit Eindhoven University of Technology

UNIVERSITY OF TWENTE.

Domain Specific Frame of Reference
Embedded Systems

2023

Final version 1 March 2023

This document contains links to other documents, as follows:

- Links to the list of references are blue numbers in square brackets: [1]
- Blue texts represent links to external documents, such as Systems Engineering roadmap or to pages, figures or tables in the current document
- Additional information is available at the www: https://visitationes.utwente.nl/DSFR2023/.
- If you read this report as a hard copy, please access the numbered references via `https://visitationes.utwente.nl/DSFR2023/doc/ReferencesDSFR.pdf`.
- The pdf of this document is available via the QR-code and here: `https://visitationes.utwente.nl/DSFR2023/DSFR_ES_2023.pdf`.

Cover photo: © Job van Amerongen
This photo is taken at CERN where a huge embedded system is used "to uncover what the universe is made of and how it works".

This document was prepared in LaTeX, using the Carlito font.

Contents

Domain Specific Frame of Reference

1 The Field of Embedded Systems

Embedded Systems

The National Science Foundation (NSF) [1] describes an embedded system as

> *"An embedded system is an application-specific computing system found inside products such as home appliances, mobile handheld personal systems (e.g., cellphones, health monitors, assistive devices, cameras, electronic games), instrumentation, automobiles, aircraft, missiles, satellites, and nuclear power plants. Embedded systems are distinguished from general-purpose computing systems by their well-defined functionalities and stringent design constraints. Embedded systems encompass all of the four fundamental operational capabilities required to interface with the world at large –sensing, computing, control, and communication– which can now be integrated with single-chip multiprocessors. The ability to combine operational capabilities in this way has opened up new possibilities for application-specific computing".*

Cyber-Physical Systems

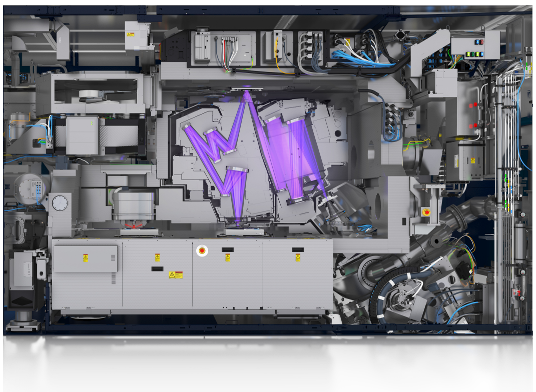
A more extended view on embedded systems which includes the physical environment in modeling and design has resulted in the area of *Cyber-Physical Systems* (CPS). ARTEMIS, p. 14 [2], formulates the difference between Embedded- and Cyber-Physical Systems as:

> *"A differentiating characteristic between Embedded and Cyber-Physical Systems is their scale. Whereas Embedded Systems are considered to have a limited scale, a CPS operates on a much larger scale, potentially including many interconnected Embedded Systems or other devices and systems as well. This may also include human and sociotechnical systems".*

This document will focus on embedded systems but, since the fields are that much related, references to Cyber-Physical Systems will be made when appropriate.

Example from the lithography domain

Chips, the electronic heart of every modern electronic system, are made in silicon through a lithographic process. Silicon slices, called wafers, need to be processed extremely fast with nanometer-accuracy. Today, handling these wafers with the required precision and speed may typically need 1000 sensors and actuators for monitoring and control, 200 servo engines, and 100 processors for control processing, executing 35 million lines of code. To improve productivity, the wafer size will increase, leading to a further explosion of monitoring, control and processing complexity. It is clear that the capability to handle this complexity is an important research challenge in embedded systems. To satisfy performance demands for the next generation of machines, an integrated development methodology is required that allows the different engineering disciplines (i.e. mechatronics, electronics and embedded software) to develop, test and integrate the control algorithms, the execution platforms and the embedded software in tandem.



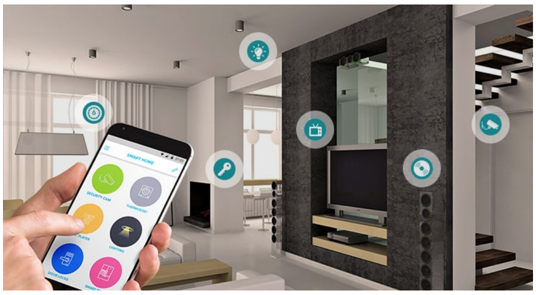
Smart grid system

The integration of renewable energy sources, such as wind turbines and solar cells in the electrical power grid requires a completely new management system able to guarantee a reliable power supply under all circumstances. Because solar and wind power depend on the weather, their availability cannot be guaranteed. A solution to balancing the supply and demand is to control the demand in a ‘smart grid’ system. This may even imply that home appliances are controlled in the sense that they are only available when the supply of power allows this. This requires that all such appliances are connected in an *Internet of Things (IoT)*. A freezer, for example, may be set at a lower temperature when power is abundantly available. This coldness buffer allows the freezer to be disconnected when there is a shortage of energy. Including many small appliances and wind and solar installations, probably together with conventional power stations, requires complex embedded systems.

Internet of Things (IoT)

Home Automation

IoT has been gradually finding its way into daily life, with home automation being a prime example. Many devices/appliances can now be controlled from a smartphone; lights can be turned on and off, the thermostat can be instructed to change the room temperature, music can be selected and played from the couch as well as commanding the television to start streaming the next episode of your favorite series. And since most of these devices are hooked up to the Internet, they can also be accessed from outside the home. Think of talking to the delivery guy by means of the video doorbell, monitoring your baby when visiting the neighbours, and being alerted when a burglar alarm goes off.



Although controlling appliances from a smartphone, or by means of a voice assistant, offers instant convenience, the true power of IoT is that these devices can be programmed (by means of a script) to

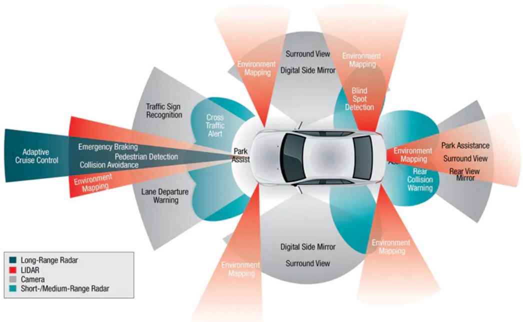
(i) automate repetitive tasks (e.g., turn off the lights when leaving the house),

(ii) schedule recurring tasks to happen automatically.

Businesses are working hard to make these features available to laymen, for example, by supporting simple IFTTT (If This Then That) rules for explicit control, and machine learning techniques to automatically distill usage patterns and building characteristics that enhance efficiency and comfort. These services typically operate from the cloud, hence, critically depend on being able to sense and actuate by using IoT-enabled devices.

Self-driving cars

Self-driving cars host many types of embedded systems for different purposes. This technology is said to have 5 levels of autonomy and ultimately aims at providing safe and efficient ground transportation for people and things without human intervention (at level 5). There are typically hundreds of different embedded systems in a car, including advanced safety systems, infotainment and navigation systems. On top of that, cutting edge embedded systems capable of sensing, analysis and control play a central role in how different car manufacturers can differentiate themselves from the rest and these are also essential for car manufacturers’ business models. For example, self-driving cars that are equipped with level-4 Waymo technology (by Google’s parent company Alphabet) utilise expensive embedded systems that run AI models on radar and lidar data. On the other hand, selfdriving embedded systems on Tesla cars (currently at level 2, aiming to reach level 5 with increasingly advanced AI models) rely mainly on analysis of data from cameras and ultrasonic sensors.



2 Embedded systems education

Embedded Systems educational programmes have been around for some time. However, there is no standard Embedded Systems curriculum. To relate our programmes to developments and visions within the Embedded Systems field, we investigated publications by an industry association (ECS-SRIA) [3] and by professional associations in related fields (CC2020 by IEEE and ACM, and the System Engineering Vision 2035 by INCOSE). We also investigated a Dutch research roadmap (HTSM Systems Engineering Roadmap by NWO), and Embedded Systems and Cyber–Physical Systems educational programmes all around the world. Besides that, we have searched for scientific publications on Embedded and Cyber–Physical Systems education.

1. *ECS–SRIA*

The Electronic Components and Systems (ECS) Strategic Research (and Innovation) Agenda [3] describes the challenges within the electrical engineering domain, a very broad domain ranging from nano–electronics to System of Systems. Embedded- and Cyber–Physical Systems are an integral part of this domain and the document therefore, provides a very useful overview of topics important for developments within the electrical engineering domain in general and the Embedded Systems domain specifically.

2. *CC2020*

The ACM: Computing curricula 2020 (CC2020) [4] document covers several undergraduate programmes in computing. The most closely related programme is Computer Engineering (CE). Computer Engineering is defined as "... the computing discipline that explicitly focuses on the development of hardware and software interface as a hardware embedded element of a computing system". Also because of the recent merger of the Embedded Systems and Computer Engineering programmes at the TU Delft, this document is relevant for the future development of our Embedded Systems programme.

3. *Systems Engineering Vision 2035*

An important topic within the Embedded Systems domain is systems engineering and the engineering of Systems of Systems. In the System Engineering Vision 2035 [5], the International Council on Systems Engineering (INCOSE) [6] provides insights in trends within systems engineering with the aim to guide future developments and indicates what is needed to realise them. Although the document does not cover the whole Embedded Systems domain, it helps to identify topics that should be included in our programme.

4. *HTSM Systems Engineering Roadmap*

The predecessor of the HTSM Systems Engineering roadmap [7] was the HTSM Embedded Systems roadmap 2018 [8]. The shift from 'Embedded Systems' to 'Systems Engineering' was made to acknowledge the increasing importance of the digital aspects of high–tech electronic systems and the increasing complexity. In contrast with the internationally–oriented Systems Engineering Vision 2035, this document focuses on the needs of the Dutch high–tech equipment industry. Because many of our students will finally be employed in the Netherlands, this document is important to connect our Embedded Systems programme with the Dutch High–Tech industry.

5. *Embedded Systems and Cyber–Physical Systems Curricula*

Besides the input from above listed documents, benchmarking our programme against existing curricula provides useful insight in the current status of, and potential developments within our programme. We therefore collected international Embedded and Cyber–Physical Systems curricula, assessed their content and related them to our programme (see references [16]–[31]). In this assessment, also scientific papers on Cyber–Physical Systems curricula development were considered (see references [32]–[35]).

6. *Input from the External Advisory Board*

A draft version of the DSFR has been discussed with the External Advisory Board (EAB). The EAB consists of representatives of ASML, Philips, Prodrive Technologies, Thales Netherlands, the European Space Agency, AEMICS, and Signify. In regular meetings, recent developments within the field of embedded systems are discussed in relation to the curriculum of the embedded systems programme.
A draft version of the DSFR was discussed with the EAB on October 18, 2022. This meeting provided valuable input and the important aspects, mentioned during the meeting, are presented at the end of this chapter. More information can be found in the minutes of the meeting [14].

In the following sections we present the most important visions brought forward in the before–listed studies.

2.1 ECS Strategic Research and Innovation Agenda (ECS-SRIA)

Foundational technology layers

The first part of the ECS-SRIA 2022 document [9] focuses on the *foundational technology layers* that are essential for building the digital systems of tomorrow (a1-a4):

a1. *Process technology, equipment, materials and manufacturing*

Advancements in material, equipment, process and manufacturing technologies utilised in the production of semiconductor components will play a vital role in increasing the industry's ability to produce resource-efficient embedded devices and systems. Society overall will benefit from innovation and adoption of more sustainable processes for embedded systems manufacturing.

Research challenges:

- advanced computing, memory and in-memory computing concepts
- novel devices and circuits that enable advanced functionality
- advanced heterogeneous integration and packaging solutions
- sustainable semiconductor manufacturing equipment and technologies

a2. *Components, modules and systems integration*

Components, modules and systems integration: Integration of a diverse set of heterogeneous functionalities into components, modules and embedded devices and systems is a challenge. Integration takes place at various levels, such as integration of sensing, imaging, actuation, communication, energy management and information processing components into one package form factor. Especially challenging is the fact that the systems (e.g., on chip) and the components to be integrated go through different production processes or different vendors.

Research challenges:

- physical and functional integration
- materials for integration
- technologies, manufacturing and integration processes
- sustainability and recyclability

a3. *Embedded software and beyond*

Embedded intelligence is key to many services and applications, and particularly for managing the overall life cycles and resources of the embedded systems involved. This is enabled by embedded software that defines functional (including sensing, control and actuation) and communication behaviour.

Research challenges:

- efficient engineering of embedded software
- continuous integration and deployment
- lifecycle management
- support for sustainability by embedded software
- software reliability and trust

a4. *System of Systems*

A System of Systems (SoS) is composed of several independent systems (hardware and software) that have dedicated tasks and responsibilities. The SoS realises functionality and tasks that constituent systems cannot individually do. SoS are becoming increasingly complex thanks to developments in embedded hardware and software development technologies.

Research challenges:

- SoS architecture and open integration platforms
- SoS interoperability
- evolvability of SoS composed of embedded and cyber-physical systems
- system of embedded and cyber-physical systems - SoECPS – engineering
- control in SoS composed of embedded and cyber-physical systems
- SoS monitoring and management

Cross-sectional technologies

The second part of the document focuses on *cross-sectional technologies* [10] that are enabled by research and innovation that emerge from the interactions between the foundational layers (b1-b4):

b1.
Edge computing and embedded AI

There is a trade-off between intelligent application performance (functional and security) and resource requirements (energy, computational resources) of embedded systems. Improving the trade-off curve is possible by distributing intelligent applications to networked embedded systems where computation and coordination are jointly organised at the edge to improve efficiency. New sustainable embedded edge computing platforms are needed, which are not only capable but also very resource efficient.

Research challenges:

- increasing the energy efficiency of embedded systems
- managing complexity of such systems and AI
- supporting life cycles of devices and systems
- sustainability of edge solutions

b2.
Connectivity

This aspect concerns advances in physical–layer connectivity technologies (beyond 5G, 6G, low power short and wide range and high-speed links) as well as communication and semantic interoperability at a high level close to the application (OSI presentation and session layers). While communication interoperability is achieved via network protocols and standardised data/packet formats, semantic interoperability aims at shared understanding of the meaning of data between communicating embedded devices.

Research challenges:

- strengthening the connectivity technology portfolio
- improving existing connectivity technologies and developing new ones
- autonomous translation of protocols, data encoding and semantics
- architectures and reference implementations of IoT and SoS connectivity
- network virtualisation for run-time deployment, management of edge/cloud

b3.
Architecture and design (methods and tools)

In order to facilitate easy creation and implementation of intelligent applications by developers, there is need for advanced architectures and design processes of (networked) embedded systems.

Research challenges:

- extending development processes and frameworks to cover the whole life cycle
- ensuring quality of new functionality in safe, secure and trustworthy systems
- managing system complexity: methods & tools for design, verification and testing
- managing diversity: model, analyse, design and test heterogeneous systems

b4.
Quality, reliability, safety and cyber security

Due to the deep penetration of embedded systems in our society and particularly in modern digital services and applications, quality, reliability, safety, and cyber security are fundamental requirements of embedded systems. Achieving them requires new approaches to the design of networked embedded systems. Thanks to increased distributed computation and communication capacity, it becomes increasingly feasible to utilise AI (machine learning in particular) for this.

Research challenges:

- hardware quality and reliability
- dependability of connected software: as dependable as traditional solutions
- security and privacy (by design)
- safety and resilience
- preparing and coordinating human-system integration as part of the innovation cycle

c.
Key application areas

The third part of the document [11] focuses on key application areas: mobility, energy, digital industry, health and well being, agrifood and natural resource, and digital society.

2.2 Computing Curricula 2020, ACM

The document discriminates between knowledge–based and competency–based computing education. Knowledge–based learning "*... is learning that revolves around both the knowledge that the student already has, and the understanding that they are going to achieve by doing work.*" (page 36). Competency–based learning is based on the following definition: "*Competencies, in the most general terms, are 'things' that an individual must demonstrate to be effective in a job, role, function, task, or duty. These 'things' include job–relevant behaviour (what a person says or does that results in* *good or poor performance), motivation (how a person feels about a job, organisation, or geographic location), and technical knowledge/skills (what a person knows/demonstrates regarding facts, technologies, a profession, procedures, a job, an organisation, etc.). Competencies are identified through the study of jobs and roles.*" (page 44).

Shift toward competency–based education

The ACM is encouraging programmes to *shift toward competency–based education* in which a mix of knowledge (know–what), skills (know–how) and dispositions (know–why) together define a competency. The document does not describe specific competencies for embedded systems education. Besides competencies, ethics and professionalism should be incorporated within the computation education.

The undergraduate programmes that are discussed in the ACM: Computing Curricula 2020 (CC2020) [4] are: Computer Engineering, Computer Science, Cyber security, Information Systems, Information Technology, Software Engineering and Data Science. Programmes on embedded systems or cyber–physical systems are not discussed. For Computer Engineering a large number of competencies is listed (page 107), which are all relevant for embedded systems as well:

1. Hardware	Hardware related aspects involve: circuits and electronics, computer architecture and organisation and digital design.
2. Software	Software related competencies that are listed are: signal processing and software design.
3. System design	System design is covered by the following competencies: embedded systems, computer networks, information security, and system resource management.
4. Academic Criteria	Academic Criteria are given: preparation for professional practice, systems and project engineering.

2.3 System Engineering Vision 2035

See also the Executive Summary. [13]

The International Council on Systems Engineering (INCOSE) [6] published the document System Engineering Vision 2035 [5]. One of the main trends that is observed is that system complexity explodes (page 8). To deal with this growing systems complexity, systems engineering covers the design of 'systems of systems'. The aim of systems engineering is formulated in this document as: "Systems engineering aims to ensure the elements of the system work together to achieve the objectives of the whole" (page vi). For the future, systems–engineering challenges are grouped into 5 categories: Applications, Practices, Tools and Environment, Research, and Competencies (page vii). For our DSFR, mainly the challenges with respect to Competencies are of interest.

The main competencies that are listed are (page 19):

1. *Core Systems–Engineering Principles*	Systems thinking, life cycles, capability engineering, general engineering, critical thinking, and system modelling and analysis.
2. *Professional Competencies*	Communications, ethics and professionalism, technical leadership, negotiation, team dynamics, facilitation, emotional intelligence, coaching, and mentoring.
3. *Technical Competencies*	Requirements definition, systems architecting, integration, interfaces, verification, validation, transition, operation, and support.
4. *Systems Engineering Management Competencies*	Planning, monitoring and control, decision management, concurrent engineering, business and enterprise integration, acquisition and supply, information management, configuration management, and risk and opportunity management.
5. *Integrating Competencies*	Project management, finance, logistics, quality.

The current state is that there is more demand than supply of systems engineers. It is generally recognised that systems engineering is increasingly important and the number of programmes in this area is increasing (page 23). It is argued to decouple systems competencies from domain competencies: "Recognition is growing that a stronger differentiation and complement of domain and system competencies is needed." (page 44). Required advancements (desired situation) within the area of systems engineering are listed as (page 60):

- Enhance workforce via life-long education/training.
- Engineering continuing education and pre-college education integrates selected systems engineering concepts and systems thinking into their curricula.
- Systems Engineering community and accreditation bodies team to add systems engineering and system concepts into all engineering accreditation criteria.
- Non-technical requirements are added to the curricula, such as human dynamics and sustainability.
- Challenge-based, hands-on education, and training of integrated methods and approaches evolves,

2.4 HTSM Systems Engineering Roadmap (2020)

This roadmap is part of the Dutch top sector High Tech Systems and Materials (HTSM), which offers modern, innovative new solutions to technological challenges. The Systems Engineering roadmap [7] is a successor of the HTSM Embedded Systems roadmap [8]. The name change reflects the increasing importance of digitisation in modern, high-tech systems; it is transforming from a supportive technology into the leading one for innovative products, phrased as

> *"a more fundamental basis of Systems Engineering is required to improve the efficiency, effectiveness, quality and costs of the architecting, design and integration processes and of the resulting products."*

This naturally translates into a need for advanced methodologies that address the demands of the high-tech industry across the critical phases of the systems' total life-cycle including initial architecting, design, realisation, integration and operational life time (page 6). Consequently, there is a need for students to master techniques ranging from design, to implementation, to deployment of complex systems (of systems):

1.
Design

1. Virtual, model-based development (models, tooling, higher abstraction levels, languages, . . .);
2. Systems thinking, including bridging multiple engineering disciplines and scalable modeling details;
3. Architectures for data-intensive and/or AI-intensive system (of systems), including IoT systems;
4. Design for system (of systems) qualities, including safety, security, performance, install-ability, diagnoseability, sustainability, re-use;
5. Autonomous, self-organising, self-learning (system of) systems;
6. Design for customer adaptations, incl. adaptability, configurability, flexibility;
7. Design and validation techniques for large-scale IoT systems.

2.
Implementation/ integration

8. Automated and highly efficient maintenance of code-bases;
9. System (of systems) integration, verification and validation.

3.
Deployment

10. From data collection to information management and decision support;
11. System health-monitoring, diagnostics and preventive maintenance.

2.5 Embedded Systems programmes from universities around the world

Benchmark

To *benchmark* [15] the programme of our embedded systems curriculum, we have studied the curricula of embedded systems and cyber–physical systems related curricula around the world. In the References at page 14 we have listed the links (see references [16]–[31]). We also studied recent scientific publications on cyber–physical–systems curriculum development (see references [32]–[35]). This study shows that technical skills are the basis for any embedded systems programme and that recent developments in the field, for example, in the areas of *Artificial Intelligence (AI)* and *Internet of Things (IoT)*, are covered in more programmes.

Recent developments

Artificial Intelligence (AI)

Internet of Things (IoT)

Furthermore, it is of utmost importance that future embedded system engineers and researchers are mastering *soft skills* (communication– and presentation skills, reflect on their own work, reflection on ethical aspects and their own role therein). These skills are discussed in Section 3.2 on Academic criteria for master's graduate in an engineering programme. More specific results of this benchmarking study have been used to evaluate our core programmes and changes have been based on this analysis.

Soft skills

2.6 Input from the External Advisory Board

1. *Autonomy and AI*

Autonomy manifests itself in different ways, ranging from completely autonomous self-driving cars to medical systems that provide assistance. These applications are built upon embedded systems and in all these applications, the role of autonomy and AI is prominent, especially at the higher system levels. From an embedded systems' point of view, AI could be considered as an application and is an important driver for (future) embedded systems education.

2. *Components and security*

Besides the HTSM Systems Engineering Roadmap, the HTSM Electronics R&D roadmap 2.1 and the HTSM Roadmap Security can be of interest. The HTSM Electronics R&D roadmap separates technological development into 'More Moore' and 'More than Moore'. 'More Moore' is focusing on further down scaling of CMOS technology where 'More than Moore' is aiming at integration of separately manufactured components from different domains (digital, analogue, RF, mixed signal) into a single system (in a package). The HTSM Roadmap Security addresses Cyber security, Active and Passive Sensor Systems, and Mission–Critical Systems. Cyber security incorporates several capacities: withstand attacks, safeguard privacy and identity, and the ability to provide society with cyber defence capabilities. The sensor systems discussed in this roadmap refer to systems that are used by armed forces to gather data to provide operational security. Mission–Critical Systems are systems that provide means to exchange information during (military) mission preparation and execution in a secure way, protected against hostile actions.

3. *DSP, GPU and FPGA*

Traditionally, Digital Signal Processors (DSPs) play an important role in the efficient execution of signal processing algorithms. However, with the advent of GPUs and FPGAs the solution–space for the design of embedded systems has increased considerably with a more and more important role for GPUs and FPGAs.

4. *Sustainability and energy efficiency*

In the evaluation of organisations, ESG (Environmental, Social, and Governance) is used to assess the (financial) effort of a company on its ethical impact and the impact on sustainability. An important aspect of sustainability is energy efficiency. Being able to operate efficiently with respect to energy has always been an enabling feature of embedded systems and remains to be so, also in light of a corporate finance point of view.

5. *Agile, Scrum*

The Agile project management philosophy is used to facilitate cooperation between development team members, and to build products faster by separating a development project into several stages. By active involvement of stakeholders in each stage, this approach facilitates easy adaptation of plans. The Scrum methodology is a realisation of the Agile philosophy. Agile is used within software development teams but can be useful in other domains as well. Therefore, it is an important project management tool within the embedded systems industry as well.

6. *Programming skills*

Within industry, it is important that students have sufficient programming skills and master multiple modern programming languages.

3 Consolidated requirements

3.1 Summary of Embedded–Systems–specific requirements

The current embedded systems programmes are based on a set of consolidated requirements that are presented in the DSFR of the previous accreditation. Because Intended Learning Outcomes and, based on that, the learning outcomes of the individual courses are based on the consolidated requirements, it was decided to assess the existing consolidated requirements against the sources in Sections 2.1–2.6. Table 1 summarises the consolidated requirements and in Table 4, the relation between the consolidated requirements and the topics mentioned in the sources 2.1–2.6 is indicated.

Table 1 Consolidated requirements of an embedded systems master's curriculum

Consolidated Requirements	Comment
1. Embedded Computer Architectures	Embedded computer architectures provide the digital processing backbone of embedded systems. Knowledge and understanding of different types of embedded systems applications is required for optimal mapping onto embedded computer architectures. Besides functional requirements, embedded computing systems must meet many non-functional requirements (dependent on the specific application domain): efficiency (low-power or efficient design methods), cost-effectiveness (high volume or low development costs), reliability, fail-safeness, small size, 'just enough' performance, real-time requirements and security aspects. These requirements often lead to highly optimised heterogeneous computer architectures consisting of general purpose processors, application specific processors, digital signal processors, GPUs, and FPGAs.
2. Software Architectures and Systems of Systems	The next generation of embedded systems is increasingly software dominated and will consist of multiple smart elements that are globally networked and that have to respond to the challenges of the embedding system. These infrastructures constitute Systems of Systems (SoS) —they are composed of many, often spatially distributed, physical subsystems that tightly interact with, and are controlled by, a large number of distributed and networked computing elements and human users. The Internet–of–Things is a prime example.
3. Methods and tools to model and verify system properties	Effective design methods and tools are needed to transform initial ideas and requirements into innovative, producible and testable products. They provide the link between the ever-increasing technology push and the demand for new innovative products and services of ever-increasing complexity. Furthermore, tools are needed to verify that the resulting system satisfies the required non-functional system properties like timeliness and energy footprint.
4. Non-functional properties: e.g. dependability, robustness and security	As embedded systems are an integral part of our daily lives, they have to satisfy non-functional properties such as dependability, robustness, and security. Dependability and robustness are essential aspects for future embedded systems applications, especially when considering the growing degree of automation. Moreover, embedded–systems applications, in particular in mission-critical tasks, must continue to operate despite disruptions caused by hardware/software failures, changing environments and limitations in sensing, processing, and communication.
5. Digital signal processing and control	In most control applications, software tightly interacts with the physical environment and this emphasises the connections between the physical and the computing world. This interaction points to global properties that are important for correct system behaviour. Besides that, Artificial Intelligence is more and more incorporated into embedded devices, either centralised or distributed within the Internet–of–Things (IoT).
6. Specialisation and Applications	There is a wide variety of applications for embedded systems and these applications are almost always of a multidisciplinary nature. New and advanced applications may require basic research. The curriculum should contain such items in the form of courses related to the specialisation of the master's thesis project.

3.2 Academic criteria for master's graduates in an engineering programme

In addition to being educated on the embedded–systems–specific topics listed in the previous section, a master's graduate must satisfy general academic criteria as well. This is also indicated in the topics that are identified in Section 2. The universities federated in 4TU have agreed on the 'Meijers criteria' [12] as a general standard for the academic knowledge, skills and attitude that a master graduate should possess. The 7 main topics are listed in Table 2. A detailed version of these criteria is provided in [12].

Table 2
Meijers criteria in short (from [12])

	Criterion
1	A university graduate is familiar with existing scientific knowledge, and has the competency to increase and develop this through study.
2	A university graduate has the competency to acquire new scientific knowledge through research. For this purpose, research means: the development of new knowledge and new insights in a purposeful and methodical way.
3	As well as carrying out research, many university graduates will also design. Designing is a synthetic activity aimed at the realisation of new or modified artefacts or systems, with the intention of creating value in accordance with predefined requirements and desires (e.g. mobility, health).
4	A university graduate has a systematic approach characterised by the development and use of theories, models and coherent interpretations, has a critical attitude, and has insight into the nature of science and technology.
5	A university graduate is competent in reasoning, reflecting, and forming a judgment. These are skills which are learned or sharpened in the context of a discipline, and which are generically applicable from then on.
6	A university graduate has the competency of being able to work with and for others. This requires not only adequate interaction, a sense of responsibility, and leadership, but also good communication with colleagues and non–colleagues. He or she is also able to participate in a scientific or public debate.
7	Science and technology are not isolated, and always have a temporal and social context. Beliefs and methods have their origins; decisions have social consequences in time. A university graduate is aware of this, and has the competency to integrate these insights into his or her work.

4 Summary of the consolidated requirements

The final list of requirements for a Master's programme Embedded Systems (Table 3) is obtained by combining the Embedded–Systems–specific requirements of Table 1 (based on a careful consideration of the various visions and studies in Sections 2.1–2.5) with the academic criteria according to Meijers in Table 2.

Table 3
Consolidated Requirements

An Embedded–Systems curriculum should cover the following Consolidated Requirements:

1. Embedded Computer Architectures
2. Software Architectures and Systems of Systems
3. Methods and tools to model and verify system properties
4. Non–functional properties: e.g. dependability, robustness and security
5. Digital signal processing and control
6. Specialisation and Applications

and must fulfil the

7. Meijers' Criteria

4.1 Mapping of the sources on the consolidated requirements

Table 4 shows the mapping of the aforementioned topics of the ECS–SRIA, CC2020, Systems Engineering Vision 2035, the HTSM Systems Engineering Roadmap, the benchmarking with existing curricula and the input from the EAB, on the consolidated requirements 1–6. The mapping of these topics on the Meijers' criteria (requirement 7) is shown as well.

Table 4
Mapping of sources on consolidated requirements

The requirements 1-6 in the top row, relate to the topics 1-6 in Table 3.

Requirement 7 relates to the Meijers criteria in Table 2.

Source		Requirements (See Table 1 and 2):	1	2	3	4	5	6	7
ECS-SRIA	a1	Process technology, equipment, materials & manufacuring	●						
	a2	Components, modules and system integration	●	●					
	a3	Embedded software and beyond		●					
	a4	Systems of Systems	●	●			●		
	b1	Edge computing and embedded AI	●			●		●	
	b2	Connectivity	●	●					
	b3	Architecture and design (Methods and tools)	●	●	●			●	
	b4	Quality, reliability, safety and cyber security			●	●			
	c	Key application areas						●	
CC 2020	1	Hardware	●				●		
	2	Software		●			●		
	3	System Design		●	●				
	4	Academic Criteria							●
Systems Engineering Vision 2035 (SEV 2035)	1	Core Systems Engineering Principles		●					
	2	Professional Competencies							●
	3	Technical Competencies	●	●	●	●	●		
	4	Systems Engineering Management Competencies							●
	5	Integrating Competencies							●
Systems Engineering Roadmap (HTSM-2020)	1	Design		●	●				
	2	Implementation/integration			●	●			
	3	Deployment				●			
Embedded Systems Curricula (ES-CUR)	1	Include recent developments	●	●			●		
	2	Soft Skills							●
External Advisory Board (EAB)	1	Autonomy and AI					●	●	
	2	Components and Security	●		●				
	3	DSP, GPU and FPGA	●						
	4	Sustainability and Energy Efficiency				●			
	5	Agile, Scrum							●
	6	Programming skills		●					

4.2 Evaluation of the mapping of sources on the consolidated requirements

From the analysis of the mapping of the sources on the consolidated requirements, we conclude that the set of consolidated requirements cover the relevant topics that are mentioned in sources 2.1–2.6. Furthermore, hardware (Embedded Computer Architectures) and software (Software Architectures and Systems of Systems) are the core of embedded systems since these consolidated requirements are mentioned most often in the sources. Furthermore, Technical Competencies are addressed in 5 out of 7 consolidated requirements. The importance of using the Meijers' Criteria for academic skills in our educational programmes is supported by 4 out of the 6 sources, indicating their importance.

Appendix: Chat GPT

While writing the DSFR, the AI tool ChatGPT suddenly popped up. In ChatGPT, questions can be asked such as: "What are the learning objectives of an academic embedded systems curriculum?" This provides not just a set of links, but an answer based on a very large data set. Although the result obtained via ChatGPT has to be critically assessed, we can also see this as an additional source for formulating the requirements of an embedded systems curriculum.

In any case, we will have to think about how we want to deal with these developments if students also start using these tools.

The learning goals of an academic embedded systems curriculum typically include:

1. Understanding of the fundamental concepts and principles of embedded systems, including hardware and software components, real-time systems, and embedded software development.
2. Knowledge of the architecture and programming of microcontrollers and microprocessors, as well as the ability to design and implement embedded systems using these devices.
3. Familiarity with embedded operating systems, real-time scheduling and synchronisation, and communication protocols.
4. Ability to design and implement embedded systems that meet performance, power consumption, and other constraints.
5. Familiarity with modern embedded system development tools, such as integrated development environments (IDEs), compilers, debuggers, and version control systems.
6. Understanding of the trade-offs involved in embedded systems design, and the ability to make design decisions based on system requirements and constraints.
7. Understanding of the issues involved in developing embedded systems for real-world applications, such as safety, security, and reliability.
8. Knowledge of the industry trends and standards in embedded systems, such as IoT, Robotics and Automation.

An academic embedded systems curriculum typically includes courses on the following topics:

1. Microcontroller architecture and programming: Students learn about the architecture of microcontrollers, such as the ARM Cortex-M, and how to program them using C or assembly language.
2. Real-time operating systems: Students learn about different types of real-time operating systems, such as FreeRTOS, and how to use them to control and coordinate the various components of an embedded system.
3. Hardware interfacing: Students learn how to interface embedded systems with different types of hardware, such as sensors, actuators, and communications devices.
4. Embedded system design: Students learn about the design process for embedded systems, including requirements analysis, system architecture, and verification and validation.
5. Embedded system security: Students learn about the security challenges in embedded systems and how to design secure systems.
6. Embedded system Project: Student will work on a real-world embedded system project which will give them hands on experience on how to design, implement and test embedded systems.

Additionally, students may also take courses in related fields such as computer engineering, electrical engineering, and software engineering to gain a more comprehensive understanding of embedded systems.

Index

References

[1] Definition Embedded Systems (National Science Foundation)
https://iucrc.nsf.gov/centers/center-for-embedded-systems/

[2] Definition Cyber-physical Systems (ARTEMIS)
https://visitationes.utwente.nl/DSFR2023/doc/Embedded%20Intelligence_%20Trends%20&%20Challenges.pdf#page=15

[3] Electronic Components and Systems (ECS) Strategic Research (and Innovation) Agenda
https://visitationes.utwente.nl/DSFR2023/doc/Artemis-InsideECSSRIA2022.pdf

[4] Computing Curricula 2020 (CC2020)
https://visitationes.utwente.nl/DSFR2023/doc/ACM-cc2020.pdf

[5] System Engineering Vision 2035
https://visitationes.utwente.nl/DSFR2023/doc/incose-se-vision-2035.pdf

[6] International Council on Systems Engineering (Incose)
www.incose.org

[7] Dutch Roadmap Systems Engineering 2020
https://visitationes.utwente.nl/DSFR2023/doc/HTSM-Roadmap-Systems-Engineering-2020.pdf

[8] Dutch Roadmap Embedded Systems 2015
https://visitationes.utwente.nl/DSFR2023/doc/RoadmapEmbeddedSystems2015REV1.1.pdf

[9] Electronic Components and Systems (ECS) Strategic Research (and Innovation) Agenda (part 1)
https://visitationes.utwente.nl/DSFR2023/doc/Artemis-InsideECSSRIA2022.pdf#page=31

[10] Electronic Components and Systems (ECS) Strategic Research (and Innovation) Agenda (part 2)
https://visitationes.utwente.nl/DSFR2023/doc/Artemis-InsideECSSRIA2022.pdf#page=157

[11] Electronic Components and Systems (ECS) Strategic Research (and Innovation) Agenda (part 3)
https://visitationes.utwente.nl/DSFR2023/doc/Artemis-InsideECSSRIA2022.pdf#page=285

[12] Criteria for Academic Bachelor's and Master's Curricula, also known as the Meijers criteria
https://visitationes.utwente.nl/DSFR2023/doc/Meijers.pdf

[13] System Engineering Vision 2035 (Executive Summary)
https://visitationes.utwente.nl/DSFR2023/doc/incose-se-vision-2035-executive-summary.pdf

[14] Minutes of the meeting of the EAB
https://visitationes.utwente.nl/DSFR2023/doc/4TUEABminutes18-10-2022.pdf

Embedded Systems Curricula

[15] Benchmark of a number of Embedded Systems programmes around the world
https://visitationes.utwente.nl/DSFR2023/doc/EmbeddedSystemsCurriculum2023_2024.pdf

[16] European Master's in Embedded Computing Systems,
European Master
2 years: University of Southampton, Kaiserslautern University and Norwegian University of Science and Technology
https://www.southampton.ac.uk/courses/european-masters-in-embedded-computing-systems-masters-msc

[17] Master's program Embedded Systems,
Chemnitz
2 years: Technische Universität Chemnitz
https://www.tu-chemnitz.de/etit/studium/stugang/index.php.en?page=m_es

[18] Embedded Systems Engineering,
Freiburg
2 years: Albert-Ludwigs-Universität Freiburg
https://www.tf.uni-freiburg.de/en/study-programs/embedded-systems-engineering/m-sc-embedded-systems-engineering

[19] Embedded Systems Engineering,
Leeds
180 UK-credits (90 ECTS): University of Leeds
https://eps.leeds.ac.uk/electronic-engineering-masters-module-information/doc/embedded-systems-engineering-msc

[20] MSc Embedded Systems Engineering,
Coventry
180 UK-credits (90 ECTS): Coventry University
https://www.coventry.ac.uk/globalassets/media/documents/registry/course-specs/eec/pg-eec-21-22/msc-embedded-systems-engineering.pdf

Carnegy Mellon

[21] Masters of Software Engineering, Scalable and Embedded Systems Curriculum,
16 months: Carnegy Mellon
https://mse.isri.cmu.edu/applicants/mse-ap/plan.html

San Francisco

[22] Embedded Electrical and Computer Systems,
30 units (2 semesters): San Francisco State University
https://dev-sfsu-engineering.pantheonsite.io/sites/default/files/documents/Course%20Requirements%20-%20Embedded%20Electrical%20and%20Computer%20Systems_0_0.pdf

UCLA

[23] Circuits and Embedded Systems Area Tracks of Electrical and Computer Engineering,
5 quarters: University of California los Angeles (UCLA)
https://grad.ucla.edu/programs/school-of-engineering-and-applied-science/electrical-computer-engineering/#program-requirements

Oakland

[24] Embedded Systems,
32 credits (2 semesters): Oakland University
https://oakland.edu/secs/masters-of-science-programs/embedded-systems/

KTH

[25] MSc Embedded Systems,
2 years: KTH
https://www.kth.se/en/studies/master/embedded-systems/description-1.70455

Halmstad

[26] Master's Programme in Embedded and Intelligent Systems,
2 years: Halmstad University
https://www.hh.se/english/education/programmes/masters-programme-120-credits-in-embedded-and-intelligent-systems.html

Chalmers

[27] Embedded Electronic system design MSc,
2 years: Chalmers
https://www.chalmers.se/en/education/programmes/masters-info/Pages/Embedded-electronics-systems-design.aspx

Tampere

[28] Advanced Studies in Embedded Systems,
80 ECTS: Tampere University
https://www.tuni.fi/studentsguide/curriculum/modules/otm-cceeb0e8-a6eb-4147-b1de-1282ae5773ed?year=2021&activeTab=1

Cyber Physical Systems Curricula

Nottingham

[29] Cyber physical Systems MSc (starts in 2022),
2 years: University of Nottingham
https://www.nottingham.ac.uk/pgstudy/course/taught/cyberphysical-systems-2year-msc

Northeastern University

[30] Cyber-Physical Systems, MS,
32 semester hours (maximum is 30 semester hours per year): Northeastern University
http://catalog.northeastern.edu/graduate/engineering/multidisciplinary/cyber-physical-systems-ms/#programrequirementstext

Institut Polytechnique de Paris

[31] Cyber-Physical Systems, Master,
2 years: Institut Polytechnique de Paris
https://www.ip-paris.fr/en/education/masters/computer-science-program/master-year-1-cyber-physical-systems

Publications on Curriculum development

[32] Isam Ishaq et al., “Work in Progress – Establishing a Master Program in Cyber Physical Systems: Basic Findings and Future Perspectives,” 2019 International Conference on Promising Electronic Technologies (ICPET), 2019, pp. 4-9
https://doi.org/10.1109/ICPET.2019.00009

[33] J. A. Stankovic, J. W. Sturges and J. Eisenberg, “A 21st Century Cyber-Physical Systems Education,” in Computer, vol. 50, no. 12, pp. 82-85, December 2017
https://doi.org/10.1109/MC.2017.4451222

[34] Didem Gürdür Broo, Ulf Boman, Martin Törngren, Cyber-physical systems research and education in 2030: Scenarios and strategies, Journal of Industrial Information Integration, Volume 21, 2021
https://doi.org/10.1016/j.jii.2020.100192

[35] P. Marwedel, T. Mitra, M. E. Grimheden and H. A. Andrade, "Survey on Education for Cyber-Physical Systems," in IEEE Design & Test, vol. 37, no. 6, pp. 56-70, Dec. 2020
https://doi.org/10.1109/MDAT.2020.3009613

contact TUD	prof.dr. K.G. Langendoen ☎ +31-15-278 7666, ✉ K.G.Langendoen@tudelft.nl
contact TU/e	dr. T. Ozcelebi ☎ +31-40-247 4426 ✉ T.Ozcelebi@tue.nl
contact UT	dr.ir. A.B.J. Kokkeler ☎ +31-53-489 4291, ✉ a.b.j.kokkeler@utwente.nl

Download the pdf version of the DSFR: https://visitationes.utwente.nl/DSFR2023/DSFR_ES_2023.pdf