



Issue Date:	2022-09-05	CEN-CLC/FGQT	N260
Deadline:	n/a	Supersedes:	–
Status: FOR APPROVAL			

TITLE	Cleaning-up chapter 7 on Quantum Computing and Simulation
PROJECT	FGQT Roadmap
REFERRING TO	Roadmap draft, N238 (Version “k”)
SOURCE	Rob F.M. van den Brink (Delft Circuits, The Netherlands), Rob.vandenBrink@Delft-Circuits.com

ABSTRACT

The content of chapter 7 (quantum computing and simulation) has organically grown during time, while we where developing relevant content. The content in section 7.1.3 (hardware stack) and section 7.1.4 (software stack) was developed more recently, and has a much higher degree of maturity than the content in the “introductory” sections 7.1.1 (overview) and 7.1.2 (pre normative gaps). These introductory parts have been created at an early stage in time while we were still developing content for quantum computing as a whole. So it may not be a surprise that most of these introductory sections are now irrelevant and/or overruled by more recent (mature) content. Moreover, the present “introductory” text does not introduce the content on hardware and software stack.

This contribution proposes a significant clean-up of old introductory text, by removing most of the superfluous content and by moving remaining sections into a new structure for further development.

1. Problem description

The content of chapter 7 (quantum computing and simulation) has organically grown during time, while developing relevant content. The content in section 7.1.3 (hardware stack) and section 7.1.4 (software stack) was developed more recently, and has a much higher maturity than the content in the “introductory” sections 7.1.1 (overview) and 7.1.2 (pre normative gaps) that were created at a more early stage. The result is that most of these introductory sections are now outdated, irrelevant, immature and do not introduce the content on hardware and software stack. Parts became irrelevant since their content are overruled by other content.

Moreover, the content of section 7.2 (quantum simulation) is almost empty, and we made no progress here. In the joined EJP paper on quantum simulation we solved this by merging the “simulation” parts into the “computing” parts. We encourage to do that in the roadmap document as well, since content in “simulation” is completely lacking in the present draft.

2. Proposed solution

We propose to remove most of the superfluous content of these introductory sections and to move relevant sections into a new (reshuffled) structure for further development.

Proposed changes

- Keep introductory fragments that are related to current developments and ongoing projects and initiatives. They deserve a place in the new section “introduction”.
- Delete fragments on pre-normative gaps in *Safety*. That part did not receive any content. The same applies to *certification*.
- Delete fragments on pre-normative gaps in *interoperability* since it has become superfluous. That topic is now well addressed in the sections on hardware and the software stack.
- Shift fragments on “*Benchmarking*” to another place. It is quite a relevant topic, and it deserve a section of its own at the same level as “hardware stack” and “software stack”.

Proposed new structure

- 7. QT Components and Subsystems: Quantum Computing and Simulation
- 7.1 Introduction (*use fragments from intro in EJP paper*)
- 7.1.1 Current developments (*a few fragments from old 7.1.1*)
- 7.1.2 Current cooperation initiatives (*optional, few fragments from old 7.1.1*)
- 7.2 Modularity and layering of hardware stack (*old sect 7.1.3*)
- 7.3 Modularity of software stack (*old sect 7.1.4*)
- 7.4 Benchmarking (*old sect 7.1.2.3*)

By applying all these proposed changes, we obtain the following new text for chapter 7.

The remarks in [blue] are for clarifying this contribution only and should not be included in the roadmap document.

Start of literal text proposal

7. QT Components and Subsystems: Quantum Computing and Q. Simulation

7.1 Introduction

[intro text taken from present EJP paper]

Quantum Computing and Simulation as an area covers many different implementations, and several enterprises are developing solutions for a mature quantum computer. The concept of a "Modular Quantum Computer", well known from digital computing, has created a new market which has attracted many small enterprises to develop dedicated modules which are competing with more monolithic full-stack organisations. The availability of a supply chain of such modules from different vendors will enable research teams to concentrate their research on breaking new grounds, without spending much effort on duplicating known solutions. This is where standardisation can play an important role.

From a standardisation point of view, this market requires a subdivision of the field of Quantum Computing and Simulation into a variety of modules that can interwork with each other through well-defined interfaces (hardware and software), and a consensus on the functional and performance requirements of each module of interest. Instead of communicating such requirements with a single or small number of local suppliers, research teams can save effort by communicating these requirements with relevant standardisation bodies.

This is where standardisation can play an important role. It will increase the availability of mature hardware and software solutions in return, as well as knowledge on requirements and solutions from others.

[If we agree to merge "quantum simulation" into "quantum computing", we can do it as]

It may be worth noting that a Quantum Simulator is a dedicated Quantum Computer, designed for solving specific problems as well as studying well defined quantum systems. They may be programmable up to a certain level. The modularity described in this present document covers both Quantum Computers and Quantum Simulators, since they use the same hardware components.

7.1.1 Current developments

[text based from present 7.1.1]

Although many different implementations do exist, the two leading quantum computing architectures in the Flagship [reference needed!] are based on different device technologies: Trapped ions and superconducting qubits. These two architectures satisfy the five required criteria for quantum computing defined by DiVincenzo:

- A scalable physical system with well characterised qubits;
- The ability to initialise the state of the qubits to a simple fiducial state;
- Long relevant decoherence times;
- A "universal" set of quantum gates;
- A qubit-specific measurement capability.

Further qubit and platform types including impurity spins in solids, neutral Rydberg atoms, topological qubits and photonic qubits, which need to be considered as future candidates for quantum computing and require further development.

The developments on quantum computing architectures can be roughly grouped into three categories:

- Those technical approaches that show theoretical merit to be pursued for several reasons, yet have not satisfied the DiVincenzo criteria;
- those systems that have fulfilled the DiVincenzo criteria, yet need to improve on system size and system control to find applications in advanced quantum algorithms, and;
- quantum architectures that are at the threshold of fault-tolerant quantum computing with sufficiently large quantum registers to realise logical qubits for large-scale quantum computation.

7.1.2 Current cooperation initiatives

[optional, if we are providing a meaningful list of projects and frameworks]

Many research teams in the world are working on hardware and software solutions for quantum computing. An (incomplete) list of programs and frameworks for cooperation is summarized below.

- List of several of on-going R&I projects (EU, national, regional...)
- List of several relevant R&I activities
- List of several relevant EU legislation framework
- more?

7.2 Modularity and layering of hardware stack

[use the existing content of (old) section 7.1.3, and renumber accordingly]

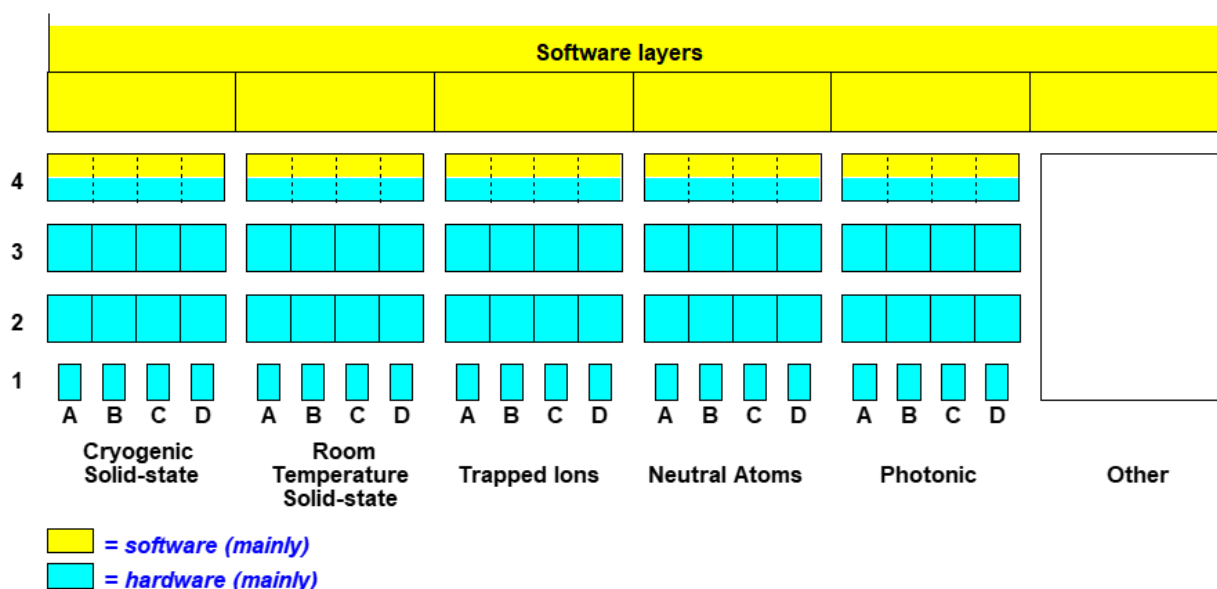


Figure 7: A possible break-down of quantum computing into layered stacks, accounting for different architectures
[note, for unclear reasons the figure has gone. Here is it again]

7.3 Modularity and layering of software stack

[use the existing content of (old) section 7.1.4, and renumber accordingly]

7.4 Benchmarking

[Based on the old 7.1.2.3, but somewhat updated to improve readability]

Well-defined and standardised methods to characterise and to benchmark quantum computing devices are necessary to compare different architectures. This facilitates reliable key characteristics of a device-under-test. Agreed ways of specifying characteristics are relevant for components in supply chains, to identify the optimal implementation for the application at hand. But agreed methods are also relevant for comparing the performance across modules, architectures or platforms as a whole.

In general, we identify two approaches. The first approach (“bottom-up”) identifies key parameters (or metrics) including measurement protocols, as well as what measurements are to be performed to specify a relevant characteristic of the device-under-test. The second approach (“top-down”) employs to-be-defined algorithms that can be run on the device-under-test. The output of the algorithm will offer the desired characterisation of the device-under-test.

The bottom-up approach is usable for individual components of a quantum computer system. Measurements have to be reliable, well-defined, and ideally traceable to SI units. In this way, a high degree of trust and comparability can be achieved via connecting to established metrology (see chapter 8 on Quantum Metrology). This is particularly important in cases where quantum computers are assembled employing components of different parties (see Section 4.7 supply chain). We note that in Chapter 5 “Enabling Technology” possible parameters can be found.

An aim of standardization is to identify a variety of methods for characterization a component, module or platform, without judging how relevant such characteristic is. The simple fact that a characteristic has a unique name, have an unambiguous description of what it means, and how it could be measured, will enable us to select preferred characterization methods in future.

End of literal text proposal
