CEN/CLC/JTC 22/WG 3 "Quantum Computing and Simulation"
WG Secretariat: **AFNOR**
Convenor: **Lefebvre Catherine Mrs**

Contribution_Rob_vandenBrink_and_Michele_Amoretti_LayerModel_CommunicationLayerDescription

Document type	Related content	Document date	Expected action
Meeting / Document for discussion	Meeting: VIRTUAL 28 Sep 2023	2023-09-14	**INFO**

Contribution on Layer Model Draft 01 from Rob F.M. van den Brink Michele Amoretti

Functional description of Layer 5; Communication Layer

Date of submission:	2023-09-14
Submitted by:	Rob F.M. van den Brink (Rob.vandenBrink@Delft-Circuits.com) Michele Amoretti (michele.amoretti@unipr.it)
Expected action:	For information
Expected date:	2023-09-28 (JTC22/WG3)
WG3-Project:	Layer Model

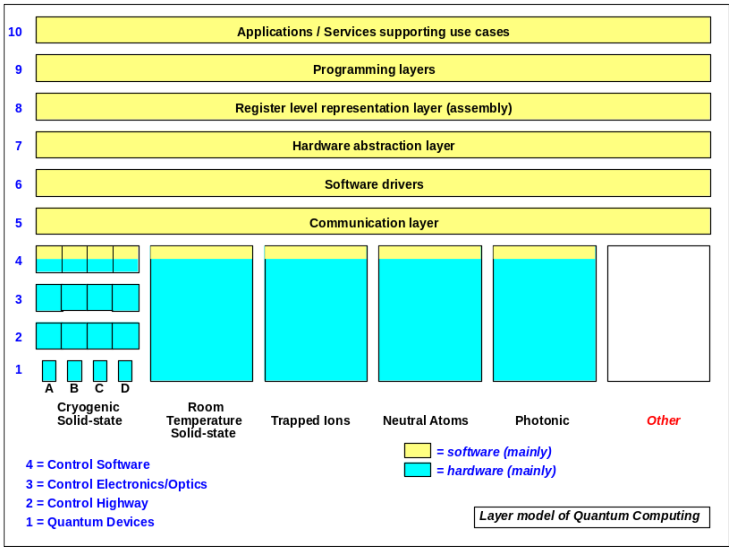
---

1. Abstract

Contribution N19 (to JTC22/WG3) has proposed a first draft of a TR “Layer model for quantum computing” with functional descriptions of each layer. Layer 5 is dedicated to the “communication layer”.
This contribution proposes a functional description of that layer, for inclusion in the draft of the aimed Technical Report (TR).

2. Proposal

The layer model that has been described in the FGQT Roadmap Document has several layers in a stack, and the low-level layers are subdivided into multiple hardware stacks. One of these layers is the “*Communication layer*”, enabling message-passing between programs. A functional description of that layer is being proposed here. The intro is taken from the originating FGQT Roadmap Document, and extended with further details.



Start of literal text proposal

X. Layer 5: Communication Layer

A quantum computer must be provided with a local operating system (OS), which is a resource manager for the underlying quantum hardware, provided with built-in networking functions allowing multiple clients to use the resources. Programs can make use of facilities only as offered by the OS. For example, the OS provides communication primitives (for instance based on the POSIX standard for the sockets interface [ref[1]]) and only by means of these primitives it should be possible to pass messages between programs.

The communication layer can handle these messages to send and receive instructions between client applications outside the quantum stack and layers inside the quantum stack. This message exchange between inside and outside the quantum stack can be *internal* as well as *external.* Internal refers to processes running on the same OS, and external refers to processes running on a nearby computer or on a remote server somewhere in the cloud. Exchange means both receiving and sending of messages.

The communication layer can handle all messages that are needed for starting a quantum computing session (for instance handshaking, authentication, resource allocation, billing, rights-management, etc.). A quantum computing session offers an application the experience as if it has its own resources and as if it is fully protected from other applications.

Once a session is initiated, the communication layer can start handling incoming messages for instructing layers higher up in the stack. For instance, to load and run a quantum assembly task. Results can be passed back to the communication layer, which in turn can send messages with those results to the client application outside the quantum stack (see Figure X.1).
The communication layer can also communicate directly with lower layers in the quantum stack, if the user is allowed to according to allocated usage rights. For instance,

[1] IEEE/Open Group 1003.1-2017, Standard for Information Technology - Portable Operating System Interface (POSIX(TM)) Base Specifications, Issue 7.

to send low-level commands directly to the control electronics for firing a specific pulse to a qubit. And again, detected results from the control electronics can also be passed back to the communication layer, which in turn can send messages with those results to the client application outside the quantum stack.

Figure X.1 illustrates the instruction flow through the layers in a graphic way. In this example the communication layer receives a message to instruct the register-level representation layer to run a quantum assembly task. When completed, the register-level representation layer informs the communication layer about the result, which in turn sends a message to the sender of the initial message.
In other words:

- Step 1 is receiving a message from a client outside the stack.
- Step 2 is forwarding the request to for instance a register-level interpreter.
- Step 3 is an instruction sequence to the quantum devices.
- Step 4 is the register-level processing of the quantum computation result.
- Step 5 is forwarding the result back to the communication layer.
- Step 6 is sending the results back to the requesting client.

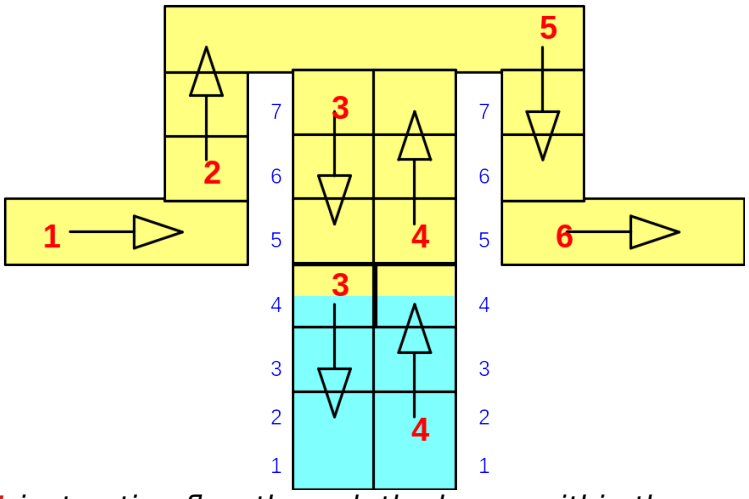


Figure X.1 *instruction flow through the layers within the quantum stack*

Since the communication layer can handle messages to communicate with all layers in the quantum stack, it may also be a convenient way to perform cross-layer communication within the quantum stack. It may not be the preferred choice to let layers interwork directly with each other when they are not direct neighbors, but sometimes it is an absolute necessity to skip a few layers for sending low-level commands directly from high-level layers and vice versa. If this cannot be avoided, the use of messages from the communication layer may offer a structured solution.

End of literal text proposal

3. Left for further study

Contributions on topics that should be added to the above literal text are invited. Topics for consideration are:

- relevant existing communication protocols (OSI) and associated documents
- elaborate on entanglement between two quantum computers and associated docs

- additional communication functionality for enabling High Performance Computing
- relation with documents from working group WG4 and other relevant “quantum networking” working groups and standardization initiatives.
- etc.

When needed the above proposed text can be extended with relevant explanations and references to associated documents.