Proceedings of 5th International Conference CNDGS'2026
ISSN 2669-2023 (print) / ISSN 2538-8959 (on-line) / **https://doi.org/10.47459/cndcgs.2026.30**

First Look at Quantum Circuits: Implications for Defence, Cybersecurity, and Military Education

Patrik NOVOSAD[1]*****

[1]*Department of Mathematics and Physics, University of Defence, Address: Kounicova 65, 662 10 Brno, Czech Republic*

*Correspondence: * patrik.novosad2@unob.cz*

Abstract

In this paper we present a structured framework for introducing quantum programming into defence-oriented education and we argue that early literacy in quantum technologies is essential for future military officers. Motivated by the rapid progress of quantum processors and the growing impact of quantum algorithms on cryptanalysis, optimisation, and data processing, we outline both the opportunities and security risks that quantum computing brings to national defence systems. The first part of paper provides a concise introduction to quantum computing fundamentals, including the nature of qubits, superposition and measurement, the action of key quantum gates, and the construction of multi-qubit states through tensor products. The second part demonstrates these principles through simple quantum circuits implemented in Python using Qiskit, illustrating phenomena such as the creation of a maximally entangled two-qubit state and an analogue of the Monty Hall paradox. By combining mathematical concepts with hands-on experimentation in a controlled simulation environment, the proposed teaching approach highlights the operational relevance of quantum thinking for cadets preparing for roles in cyber defence, signals and communications, electronic warfare, and intelligence analysis. The article concludes that integrating quantum programming into military education enhances technological preparedness and strengthens the ability of defence institutions to anticipate quantum-era challenges, including the transition to quantum-resistant cryptography and the incorporation of quantum technologies into multidomain operations.

KEY WORDS: *quantum computing; quantum circuits; qubit; qiskit;*

Citation: Novosad, P. First Look at Quantum Circuits: Implications for Defence, Cybersecurity, and Military Education. In Proceedings of the Challenges to National Defence in Contemporary Geopolitical Situation, Brno, Czech Republic, 7-10 September 2026. ISSN 2538-8959, https://doi.org/10.47459/cndcgs.2026.30

1.Introduction

Quantum computing is rapidly evolving, with major technology leaders, such as IBM [1] or Google [2], developing increasingly powerful quantum processors, positioning quantum technologies as a significant factor in future security and defence environments. Although current quantum hardware remains limited, projected advancements indicate that quantum-enabled capabilities, particularly in cryptanalysis, optimization, and data processing, may fundamentally reshape cyber defence and military decision-making [3]. For national defence systems dependent on secure communication, encryption, and rapid information processing, the emergence of quantum algorithms such as Shor's algorithm [4] presents both opportunities and substantial risks. The ability of quantum computers to break widely used asymmetric cryptosystems, like RSA [5], could directly threaten military mobility, command-and-control structures, and secure data exchange within NATO forces. This contribution introduces the basics of quantum programming, with emphasis on quantum circuits, and demonstrates how they can be taught to cadets as part of modern defence-oriented education. The approach integrates mathematical principles of tensors, linearity and unitary operators with practical implementation of quantum circuits using Python and its library Qiskit [6], enabling students to experiment with quantum gates and evaluate the behavior of quantum states in a controlled simulation environment. The teaching framework highlights the operational relevance of quantum concepts for future officers, especially those preparing for roles in cyber units, signals and communications, electronic warfare, and intelligence analysis. By understanding entanglement, superposition, and circuit behavior, cadets gain foundational literacy necessary to assess both the defensive applications of quantum computing and emerging security threats.

This paper argues that early introduction of quantum programming strengthens military technological preparedness and enhances the ability of defence institutions to anticipate quantum-era challenges, including the development of quantum-resistant cryptography and the integration of quantum technologies into multidomain operations. Although the proposed teaching method has not yet been tested in a live cadet environment, it offers a scalable and accessible model for incorporating quantum computing into national defence education.

2.Preliminaries

The theoretical basics of quantum computations could be remarkably seen as a union of linear algebra and theory of probability; both are a usual part of mathematics syllabus at military universities. We begin with necessary mathematics minimum to understand the quantum circuits. We take approach without any rigorous definition, and we show every idea in examples, from which the generalization should be obvious.

A qubit is a vector in two-dimensional vector space over complex number. We denote qubits by so called bra-ket notation, it means that this object $|\psi\rangle$, called ket vector, is a qubit (we also sometimes refer to it as a state or a state-vector). The standard basis of this two-dimensional vector space are vectors $|0\rangle$ and $|1\rangle$, which corresponds to the classical values of a bit from informatics. As any vector from a vector space is just linear combination of the basis vectors, we can write, for an arbitrary qubit

$$|\psi\rangle = (a + \mathrm{i}b)|0\rangle + (c + \mathrm{i}d)|1\rangle. \tag{1}$$

where $a, b, c, d \in \mathbb{R}$, with a condition on these parameters $a^2 + b^2 + c^2 + d^2 = 1$. The bra-ket notation is suitable for general considerations, for practical calculations, it is good to bear in mind that each qubit can be written also in matrix notation as a column vector,

$$|\psi\rangle = \begin{bmatrix} a + \mathrm{i}b \\ c + \mathrm{i}d \end{bmatrix}. \tag{2}$$

In some cases, it is also good to bear in mind, that the basis ket vectors $|0\rangle$ and $|1\rangle$ can be written as column vectors

$$|0\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \quad \text{and} \quad |1\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix}.$$

We introduce another type of vector, so called bra vector, which is denoted as $\langle\psi|$, and defined as $\langle\psi| = |\psi\rangle^{*T}$, where the asterisk means complex conjugation and the capital T stands for transposition of a vector. Therefore, for the ket vector given by eq. (2) the corresponding bra vector is the following row vector

$$\langle\psi| = [a - \mathrm{i}b, c - \mathrm{i}d], \tag{3}$$

or in a similar notation to eq. (1)

$$\langle\psi| = (a - \mathrm{i}b)\langle 0| + (c - \mathrm{i}d)\langle 1|, \tag{4}$$

where $\langle 0|$ and $\langle 1|$ are bra vectors dual to ket vectors $|0\rangle$ and $|1\rangle$. The 'dual' in this case means the following relations

$$\langle 0|0\rangle = 1,\ \langle 0|1\rangle = 0,\ \langle 1|0\rangle = 0,\ \langle 1|1\rangle = 1,$$

which we take as the definition for scalar product on qubits. Loosely speaking, when we take the representations of bra and ket basis vectors as column and row vectors, these relations correspond to standard scalar product in the vector space over complex numbers. Moreover, consider two generic vectors $|\phi\rangle = \alpha|0\rangle + \beta|1\rangle$ and $|\varphi\rangle = \gamma|0\rangle + \delta|1\rangle$, with $\alpha, \beta, \gamma, \delta \in \mathbb{C}$, for their scalar product we have

$$\langle\phi|\varphi\rangle = (\alpha^*\langle 0| + \beta^*\langle 1|)(\gamma|0\rangle + \delta|1\rangle) = \alpha^*\gamma\langle 0|0\rangle + \alpha^*\delta\langle 0|1\rangle + \beta^*\gamma\langle 1|0\rangle + \beta^*\delta\langle 1|1\rangle = \alpha^*\gamma + \beta^*\delta.$$

Similarly, we can calculate that $\langle\varphi|\phi\rangle = \gamma^*\alpha + \delta^*\beta$, therefore we have the following relation for scalar product between two vectors

$$\langle\varphi|\phi\rangle^* = \langle\phi|\varphi\rangle.$$

In the same manner we could do the scalar product of a vector with itself, in particular for vector given by eq. (1) we have

$$\langle\psi|\psi\rangle = ((a - \mathrm{i}b)\langle 0| + (c - \mathrm{i}d)\langle 1|)((a + \mathrm{i}b|0\rangle) + (c + \mathrm{i}d)|1\rangle) = a^2 + b^2 + c^2 + d^2 = 1, \tag{5}$$

where we use the constraint under eq. (1) to get the final result. This means that every qubit has a norm (or magnitude) equal to one. This fact is crucial for probability part of qubit calculations as we will see shortly.

We continue with the measurements of the qubits. This property of qubits is inherited from quantum physics, and it is a cornerstone of power of qubits. Before measuring, the qubit could be in any combination of the classical bit state 0 or 1. This combination is of course given by eq. (1). After the measurement the qubit collapses to the well define classical bit state. But the same qubit could collapse to different classical bit state, once the result after measurement is 0,

another time the result of the measurement of the same qubit is 1. The probabilities of the results correspond to the parameters of the qubit. To be specific, assume the following form of a qubit

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle, \tag{6}$$

with $\alpha, \beta \in \mathbb{C}$ and with condition $|\alpha|^2 + |\beta|^2 = 1$, where $|\alpha|^2 = \alpha\alpha^*$. To measure the qubit, we apply the basis bra vector to this qubit and then make a (complex) square of the result

$$|\langle 0|\psi\rangle|^2 = |\alpha\langle 0|0\rangle + \beta\langle 0|1\rangle|^2 = |\alpha|^2,$$
$$|\langle 1|\psi\rangle|^2 = |\alpha\langle 1|0\rangle + \beta\langle 1|1\rangle|^2 = |\beta|^2.$$

Therefore, the probability that qubit $|\psi\rangle$ collapses during the measurement to the classical bit state 0 is $|\alpha|^2$, and that it collapses to the classical bit state 1 is $|\beta|^2$. From this, we can see the importance of the condition given by eq. (5) or under eq. (6), it postulates that there is one hundred percent chance to obtain result to be 0 or 1. The result is certainly the classical bit, but before the measurement we don't know the value of it. As example consider following qubit $|\psi_1\rangle = \frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle$, this qubit has 50% chance to be measured as 0 and 50% chance to be measured as 1. We can say that this qubit is a model of the flipping of coin, which has two outputs, both with 50% chance. But here something strange happens, we could consider another qubit $|\psi_2\rangle = \frac{1}{\sqrt{2}}|0\rangle - \frac{1}{\sqrt{2}}|1\rangle$, which is clearly different from $|\psi_1\rangle$, but the probability of the outcomes 0 and 1 is the same. Thus, it is clear that there are more qubits which have probability of both outcomes 50%, in fact, due to complex nature of the qubits, there are infinitely many of them. The difference between them is called phase, it is something, what can't be measured, but for complex quantum circuits the phase is important, and it brings quite peculiar results into the quantum calculations. One last remark on the measurement, in general, the measurement could be made with respect to any basis. The procedure is the same, as described above, just the ket and bra vectors has to be written in the same basis.

Now we focus on to how to change the qubits. As the qubits are the vectors, it is natural that the operators, which will be applied to them, are matrices. Consider again the qubit given by eq. (6). We could apply a two-by-two complex matrix U, to get a new qubit $U|\psi\rangle$. As this is again the qubit, it must fulfil the condition (5), therefore

$$\langle\psi|U^{T*}U|\psi\rangle = 1 = \langle\psi|\psi\rangle = \langle\psi|E|\psi\rangle, \tag{7}$$

where in the last step we insert an identity matrix E into the scalar product (the multiplication by identity matrix does not change the vectors). From eq. (7) we can deduce that

$$U^{T*}U = E,$$

or when we do a little rearrangement

$$U^{-1} = U^{T*}. \tag{8}$$

It means that the inverse of the matrices, which can be applied to qubits, has to be calculated as complex conjugate and transpose of itself. Any other matrices would not preserve the norm of qubits; therefore, they would change the maximum of probability (which is 100%) to another number. The matrices which satisfy the condition (8) are called unitary matrices and they have another nice property, in particular $|\det U|^2 = 1$, which in other words says that the maximum probability cannot be changed. In the quantum circuits, the unitary matrices are called gates, and they change the state of the qubit. For example, two following gates are common in the quantum circuits, the NOT gate X and Hadamard gate H. The NOT gate has the following matrix representation

$$X = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}.$$

This gate flips the basis vector $|0\rangle$ and $|1\rangle$ to each other, i.e.

$$X|0\rangle = |1\rangle, \quad X|1\rangle = |0\rangle.$$

This can be seen from matrix representation of gates and qubits, because the application of the matrix onto vector is given just by matrix multiplication

$$X|0\rangle = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}\begin{bmatrix} 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \end{bmatrix} = |1\rangle, \qquad X|1\rangle = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}\begin{bmatrix} 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \end{bmatrix} = |0\rangle.$$

The second quantum gate, Hadamard gate, has the following matrix representation

$$H = \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{bmatrix},$$

and when we apply it on the basis qubits, we get

$$H|0\rangle = \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \begin{bmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{bmatrix}, \quad H|1\rangle = \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{bmatrix} \begin{bmatrix} 0 \\ 1 \end{bmatrix} = \begin{bmatrix} \frac{1}{\sqrt{2}} \\ -\frac{1}{\sqrt{2}} \end{bmatrix}.$$

We already encountered these qubits, we denoted them $|\psi_1\rangle$ and $|\psi_2\rangle$ when we discussed the measurement. Both qubits have 50% chance of being measured as 0 or 1. They only differ in the phase, which can't be measured. But there is a way to distinguish between them. If we apply again Hadamard gate, we obtain back the original basis qubits $|0\rangle$ and $|1\rangle$. This is due the fact that Hadamard gate is inverse to itself (it satisfies the condition (8)). The take home message from this is even if the phase cannot be measured its importance to the quantum computations is huge, as the changing of phase somewhere in the circuit can lead to totally different results.

The last paragraph is dedicated to multi qubits systems. We present all of it on just two qubits, the generalization on the more qubits is straightforward. For basis qubits $|0\rangle$ and $|1\rangle$ the notation is as follows:

$$|0\rangle \otimes |0\rangle = |00\rangle, \quad |0\rangle \otimes |1\rangle = |01\rangle, \quad \text{etc.} \tag{9}$$

The tensor product $\otimes$ is not commutative and serves as a separator of qubits. In more readable notation, we omit this tensor product and write basis state into one ket vector, but we can't change the order. For generic qubits, let's say $|\psi\rangle$ and $|\phi\rangle$, we keep the explicit tensor product in the notation $|\psi\rangle \otimes |\phi\rangle$. If we know their coordinates in the basis, for example

$$|\psi\rangle = \tfrac{1}{\sqrt{2}}|0\rangle + \tfrac{\mathrm{i}}{\sqrt{2}}|1\rangle, \quad |\phi\rangle = \mathrm{i}|1\rangle,$$

it is easy to calculate the tensor product due to its linearity

$$|\psi\rangle \otimes |\phi\rangle = \left(\tfrac{1}{\sqrt{2}}|0\rangle + \tfrac{\mathrm{i}}{\sqrt{2}}|1\rangle\right) \otimes \mathrm{i}|1\rangle = \tfrac{\mathrm{i}}{\sqrt{2}}|01\rangle - \tfrac{1}{\sqrt{2}}|11\rangle.$$

In the same sense we can introduce the tensor product of gates, for example $H \otimes X$, it acts on the qubits according to the order of the components, this means following equality

$$(H \otimes X)|\psi\rangle \otimes |\phi\rangle = H|\psi\rangle \otimes X|\phi\rangle.$$

In particular, the acting of tensor product of gates on the basis qubits is following

$$(H \otimes X)|00\rangle = H|0\rangle \otimes X|0\rangle = \left(\frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle\right) \otimes |1\rangle = \frac{1}{\sqrt{2}}|01\rangle + \frac{1}{\sqrt{2}}|11\rangle.$$

As we see, the tensor product in bra-ket is quite intuitive and easy to calculate. On the other hand, it is quite impractical for common calculations. More suitable is matrix notation, due to existence of simple matrix multiplication. The basis of two qubits system in matrix notation is

$$|00\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \otimes \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \quad |01\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \otimes \begin{bmatrix} 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix}, \quad |10\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \otimes \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}, \quad |11\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \otimes \begin{bmatrix} 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}.$$

The order of components is crucial, this convention is called 'lexicographic', the first index is of the first vector, inside it is the index of second vector. Loosely speaking, we take the vector on the right and multiply it by first number of the vector on the left, then we take again the vector on the right and multiply it with second number of the vector on the left, the resulting vectors are written below each other, and we obtain final four rows vector. The same situation is for tensor product of gates in matrix notation. We show it on the gates H and X

$$H \otimes X = \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{bmatrix} \otimes \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} = \begin{bmatrix} 0 & \frac{1}{\sqrt{2}} & 0 & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & 0 & \frac{1}{\sqrt{2}} & 0 \\ 0 & \frac{1}{\sqrt{2}} & 0 & -\frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & 0 & -\frac{1}{\sqrt{2}} & 0 \end{bmatrix}.$$

The computation is analogical to vectors, we take matrix on the right and multiply it with numbers in the matrix on the left, then we compose the new matrix by joining each block in the sense of matrix on the left. Maybe, it will be better seen in comparison with tensor product $X \otimes H$

$$X \otimes H = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \otimes \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{bmatrix} = \begin{bmatrix} 0 & 0 & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ 0 & 0 & \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & 0 & 0 \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} & 0 & 0 \end{bmatrix}.$$

The last remark is about entanglement. It is possible to construct a two-qubit system, on which is enough to measure just one qubit to also know the value of the second qubit. This doesn't look strange at first sight, but before the measurement the first qubit could be 0 or 1, and by the measuring, not only the first qubit but also the second qubit collapses to a certain value. The example of entangled qubit is

$$|\psi\rangle = \frac{1}{\sqrt{2}}|00\rangle + \frac{1}{\sqrt{2}}|11\rangle\,, \tag{10}$$

which has 50 percent to have both qubits measured as 0 or 1. But it is also possible to measure just one qubit, and the measurement of the second is given. This property is intrinsic to quantum systems and can be used for communication. It also brings a lot of conceptual problems from the psychical point of view [9].

3.Quantum Circuits

The quantum computing, which we presented in the previous chapter, is all about matrix calculations. Which has very straightforward rules but is more computationally demanding as the matrix dimensions increase. And the dimensions of the matrices increase exponentially. In particular, for 10 qubit systems we have to deal with matrices $2^{10} \times 2^{10} = 1024 \times 1024$, which in no way is doable in hand, and even on classical computers larger systems take a significant portion of time. This is one of the advantages of quantum computing, it can reduce the time needed for calculation. Instead of explicitly manipulating exponentially large matrices, a quantum device naturally evolves the entire quantum state in hardware, allowing some classes of problems to be solved dramatically faster than on classical machines.

The standard way to carry out calculations on the quantum computer is with use of quantum circuits. A quantum circuit is a unitary matrix, that is made up of individual building blocks, which are smaller unitary matrices called gates. The initial state on which the quantum circuit act is a zero qubit $|00\ldots0\rangle$, the final state is a qubit, which will be measured. As a framework for work with quantum circuits, we choose Qiskit [6], which is a library of programming language Python [10]. As an interactive environment for working with code we choose Jupyter notebook [11]. The guides on how to install all the programs needed can be found in respective references.

The first illustrative example will be construction of the qubit (10), i.e. the fully entangled qubit. First, we import functions which will be used, from Qiskit library:

```
from qiskit import QuantumCircuit
from qiskit_aer import AerSimulator
from qiskit.visualization import plot_histogram
```

Then we implement the quantum circuit on the two qubits, which we name 'circuit':

```
circuit = QuantumCircuit(2)
```

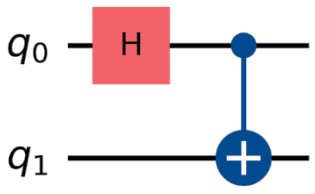


Figure 1: The quantum circuit consisting of Hadamard gate and CNOT gate.

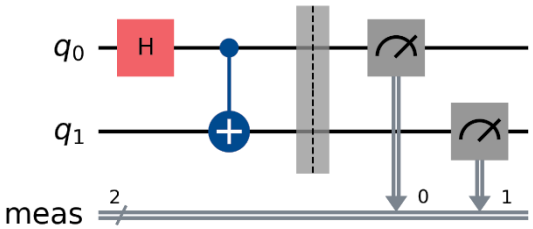


Figure 2: The same circuit as on Figure 1, but with the measuring of both qubits. The barrier after the gates is an element with no function, just to separate two areas.

And we conclude it by adding Hadamard gate H on first qubit and then between first and second qubit we add CNOT (controlled NOT) gate CX, which we will discuss later, and finally we draw the circuit:

```
circuit.h(0)
circuit.cx(0,1)
display(circuit.draw(output="mpl"))
```

One note is needed here, Python indexes everything from zero, the first item in the array has label zero. Because of this, when we want to add anything to first qubit, we have to write number 0 into the parenthesis. The resulting picture of the circuit is in *Figure 1*. The CNOT gate is an example of controlled gates. These gates have a control qubit, in *Figure 1* it's the first qubit, marked by a dot. If the control qubit has value 0, nothing happens, if the control qubit has value 1, the gate, which is acting on the second is performed. The CNOT gate has the following matrix

$$CX = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix},$$

and can't be written as a tensor product of two another gates, this is a property of the controlled gates. When we know what CNOT gate does, we can describe the circuit in Figure 1. The initial state, as in every quantum circuit, is $|00\rangle$. The Hadamard gate makes the superposition of the first qubit, it has 50% chance to be 0 and 50% chance to be 1. When the first qubit is 0, the CNOT gate is not applied and the result is $|00\rangle$. But when the first qubit has value 1, the CNOT gate is applied and changes the value of the second qubit to 1. Altogether, the result of this quantum circuits is $|00\rangle$ or $|11\rangle$ both with 50 percent chance, it's exactly qubit given by eq. (10), which could be computed. But we want to demonstrate here how to obtain the statistic result due to simulation. This is done by measuring the qubits multiple times. First, we have to adjust the code by adding the following

```
circuit.measure_all()
```

to last but one line. This means that all qubits will be measured and we could do statistics of the measurements. The circuit with measuring is in *Figure 2*. For the measuring itself we have to add to the end of the code following line

```
result = AerSimulator().run(circuit).result()
```

which will carry out 1024 measurements of our circuit. To display the results we use the histogram, which we plot by following code:

```
statistics = result.get_counts()
display(plot_histogram(statistics))
```

The histogram is in *Figure 3*. As we see there are roughly same number of measurements which collapse to $|00\rangle$ or $|11\rangle$ state, which corresponds to our theoretical analysis. Each time we run the simulation, we get different results. This is due to fact that the simulations are random, which should imitate the Hadamard gate. To get more precise result, there must be more measurements. But in our case the number of measurements is sufficient. Last remark on this circuit. The numbers

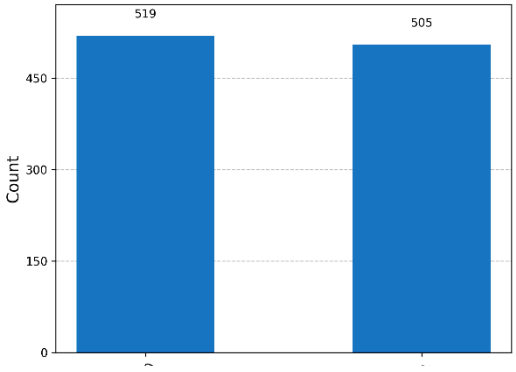


Figure 3: Histogram of the 1024 measurements of circuit in Figure 2. It is roughly 50:50. The more measurements would be done the more precise the result would be.

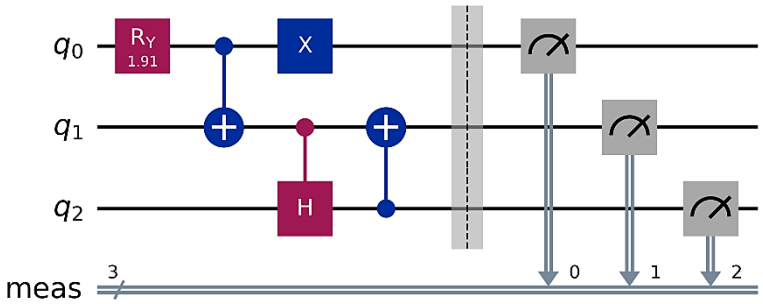


Figure 4: Quantum circuit which bring the state $|000\rangle$ into fully entangled state $\frac{1}{\sqrt{3}}|001\rangle + \frac{1}{\sqrt{3}}|010\rangle + \frac{1}{\sqrt{3}}|100\rangle$.

in *Figure 3* on x-axis are written in column on purpose. The order of number corresponds to the order of qubits in *Figure 2*. This means that qubit q_0 has 519 value 0. When we compare this vertical ordering with ordering of tensor product, for example eq. (9), the top qubit in vertical ordering is rightmost qubit in tensor product (9), the bottom qubit in the vertical ordering is the leftmost qubit in tensor product.

For the second example of circuit, we choose analysis of following situation. Three soldiers are captive. The jailer says that two of them are going to be executed and just one will be released. This means that each soldier has 1/3 chance to be released. We take role of one of the captives. We ask the jailer which one from the other two will be executed. Our arguments are that one of them has to be and we still don't know who will be released. The jailer doesn't see any problems and tells it to us. The second thing we want for the jailer is we want to change our fate with the second of the other two, the one whose fate we don't know. Our argument is that there is 50:50 chances to be released or executed, so it doesn't matter if there will be a change. The point is if the jailer agrees about this change, our chance of being released will not be 50 percent, but 66,6 percent. We show this unintuitive outcome is true by using a quantum circuit.

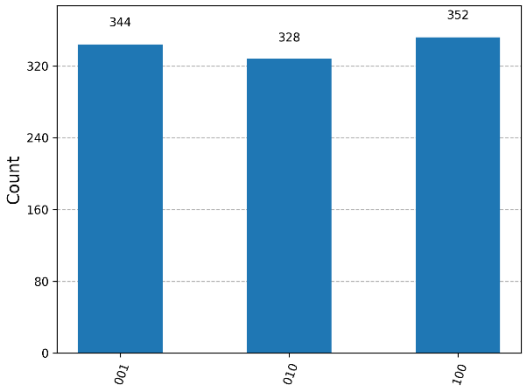
We first present the construction of such quantum circuit; the code will be given afterwards. The main idea, why for this situation a quantum circuit can be used is the three-qubit system $\frac{1}{\sqrt{3}}|001\rangle + \frac{1}{\sqrt{3}}|010\rangle + \frac{1}{\sqrt{3}}|100\rangle$. The one in this system means the captive, which would be released. The zeros are for the ones who will be executed. So, by the measurement of this qubit there is 33,3% chance, that the first will be released, 33,3% chance, that the second will be released and so on. To use this qubit in the circuit, we have first to prepare such a qubit. The first thing we need is a qubit, which has 1/3 chance to be measured as 0 and 2/3 chance to be measured as 1. This could be achieved with $RY(\theta)$ gate, which depends on the parameter θ and is given as

$$RY(\theta) = \begin{bmatrix} \cos\frac{\theta}{2} & -\sin\frac{\theta}{2} \\ \sin\frac{\theta}{2} & \cos\frac{\theta}{2} \end{bmatrix}.$$

We have to calculate θ for which the application of this gate yields the desired qubit

$$RY(\theta)|0\rangle = \frac{1}{\sqrt{3}}|0\rangle + \sqrt{\frac{2}{3}}|1\rangle.$$

From this equation we get $\theta = 2\cos^{-1}\frac{1}{\sqrt{3}} = 1.910633236$. This gate is applied to the first qubit. After this gate we add one CNOT gate between first and second qubit. This CNOT gate will flip second qubit into 1, if the first qubit is 1. And we proceed with NOT gate on the first qubit. The idea behind this gate is flip the first qubit to 0, if the previous CNOT gate is applied, or to flip the first qubit to 1, if the CNOT is not applied. Together the resulting state should be $|10\rangle$ with probability 1/3 and $|01\rangle$ with probability 2/3. The figure is sometimes much better than word description, so the full three qubit quantum circuit is in *Figure 4*, the described circuit consists of just the first three gates on first two qubits. Then we use the second qubit and halve its probability on behalf of third qubit. We apply the controlled Hadamard gate between second and third qubit, the control qubit is the second one. If the second qubit is 0, nothing happens and we are left with state $|100\rangle$ after measurement, which is one of states we want. If the second qubit is 1, the Hadamard gate is applied on third qubit and we have two states $|010\rangle$ and $|011\rangle$, both with conditional probability 50:50 (the condition is that first qubit is zero). Therefore,



if the third qubit has a value 1, we have to flip the second qubit to value 0, to get desired state $|001\rangle$.

Figure 5: Histogram of 1024 simulated measurements of circuit in Figure 4. It shows that the circuit yields only states $|001\rangle$, $|010\rangle$ or $|100\rangle$, with probability 1/3 each.

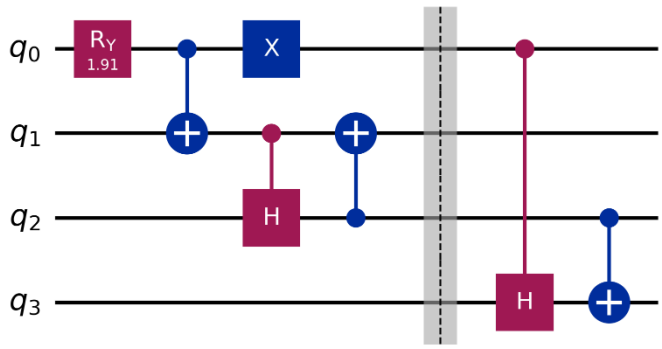


Figure 6: The quantum circuit after adding the answer from the jailer. Answer means, who among two others will be executed. The two gates after barrier ensure, that if the value of q_3 is 0, the answer is q_2 and if the value of q_3 is 1, the answer is q_1.

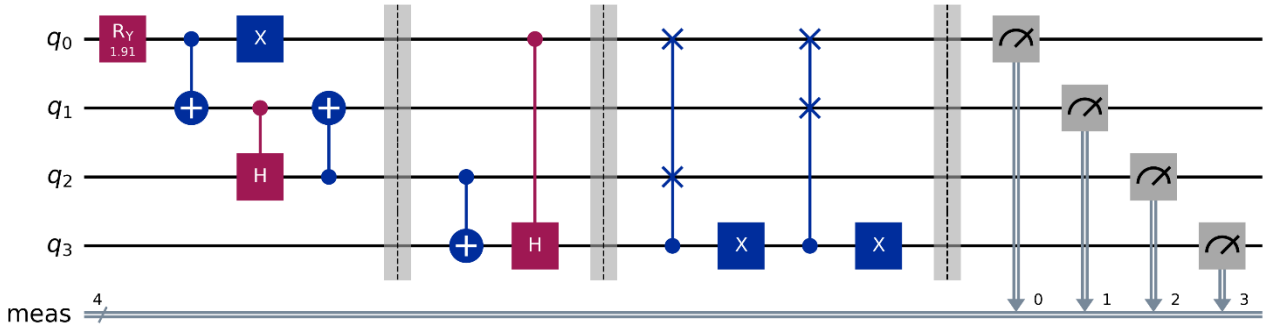


Figure 7: The whole quantum circuit for the situation described in the second part of section 3. For better readability we separate the circuit four parts, the initial state, the jailer's answer, swapping the fate and the measurement.

This is done by another CNOT gate, this time between third and second qubit (control qubit is the third one). As was said, the whole circuit is in *Figure 4*. This circuit really takes the state $|000\rangle$ to state $\frac{1}{\sqrt{3}}|001\rangle + \frac{1}{\sqrt{3}}|010\rangle + \frac{1}{\sqrt{3}}|100\rangle$, which can be verified with simulation. The result of simulation is in *Figure 5*; the probabilities are roughly one-third.

We have prepared the crucial three-qubit state for our solution to the situation. Now we add the jailer to the circuit. We ask the jailer, who among the other two will be executed. If among the other two, one will be released and second executed, the jailer has no choice, they must say the one who will be executed. If on the other hand, we are going to be released, the jailer can answer in random, as both will be executed. The random 50:50 choice sounds like Hadamard gate, and in fact we use it. But first, without loss of generality, let's take the role of the first captive, which is represent by qubit q_0. We also represent the jailor as new qubit, q_3. If the value of the q_3 is 1, the answer to our question is, that second captive q_1, will be executed. If the value of the q_3 is 0, the answer is that third captive q_2, will be executed. If only one of them is executed and other is released, the jailer has no choice as was already said. Thus, if the value of q_1 is 1, i.e. q_1 will be released, they have to say that q_2 will be executed, therefore qubit q_3 should be 0 and there is no need for any gate. On the other hand, if the value of q_2 is 1, the q_1 is going to be executed, therefore q_3 has to be 1. We do this by CNOT gate between q_1 and q_3, where q_1 is control qubit. And the last eventuality is that we are going to be released, this means that the value of q_0 is 1. Then the q_3 can be 0 or 1, as both other captives will be executed. We do it with use of controlled Hadamard gate between q_0 and q_3, where q_0 is control qubit. The quantum circuit after adding these gates is in *Figure 6*.

The last thing, which have to be done, is swapping our fate with the second of the others. This will be done with usage of controlled SWAP gates. SWAP gate does exactly what it sounds to do, it exchanges two qubits. On the two-qubits states it acts as

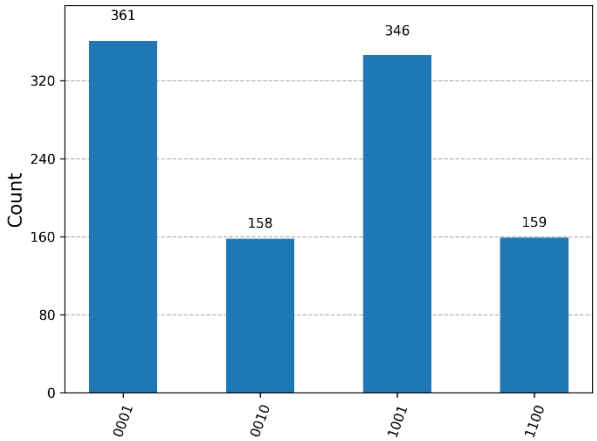


Figure 8: 1024 simulations of measurements of circuit in Figure 7. The 1 of first three qubit from the top means, which captive is going to release. We see that the first one, which represents us, has chance about 2/3. The last qubit is tracking the jailer answer, for releasing the captive, it has no meaning for the output. The following insight into the situation could be helpful. Due the swapping of the fate, we are always released in the case, when one of the other should be released. The only case, when we are going to be executed is, when originally, we had to be released, with chance 1/3.

$$\mathrm{SWAP}|00\rangle = |00\rangle,\ \mathrm{SWAP}|01\rangle = |10\rangle,\ \mathrm{SWAP}|10\rangle = |01\rangle,\ \mathrm{SWAP}|11\rangle = |00\rangle.$$

The controlled SWAP gate is just controlled version of it, with one control qubit which initializes the swap. The control qubit is q_3, if it has value 1, we swap ourselves with third captive, i.e. q_0 with q_2, if it has value 0, we swap q_0 with q_1. The following implementation is not the most simplified one, but it is the most illustrative. When the value of q_3 is 1, we only need controlled SWAP gate between q_0 and q_2 with control qubit q_3. But when the value of q_3 is 0, the controlled SWAP gate will not be applied. In this case, we first use NOT gate on qubit q_3, then apply controlled SWAP gate between q_0 and q_1, with control qubit q_3, and then we turn off q_3 with another NOT gate. This works due to fact that if q_3 has value 1 before this, the first NOT gate changes it to 0, and the controlled SWAP gate is not applied. Altogether, the full quantum circuit for our considered situation is in *Figure 7*. At last, we simulate again 1024 measurements of this circuit, to see if the result is the desired one. The histogram of the simulation is in *Figure 8* and it needs comments. The 1 in the first three top numbers correspond to the captive, which is released. We see that two biggest columns have top number 1. That means the first captive, which represents us in this situation, is released in 2/3 cases. The number on the bottom corresponds to the answer of the jailer by the key given above. For the outcome of the situation, this number is meaningless. Therefore, the simulation agrees with our hypothesis, with one question and one exchange it is possible to double our chances in the hypothetical situation above. We conclude this section with presentation of the code which constructs the circuit in *Figure 7* and does the simulation with results which are in *Figure 8*. Keep in mind that the simulation is random, i.e. it has different results each time. But the trend should be obvious, and the simulations are more stable with more measurements.

```
from qiskit import QuantumCircuit
from qiskit_aer import AerSimulator
from qiskit.visualization import plot_histogram
qc = QuantumCircuit(4)
qc.ry(1.910633236,0)
qc.cx(0,1)
qc.x(0)
qc.ch(1,2)
qc.cx(2,1)
qc.barrier()
qc.cx(2,3)
qc.ch(0,3)
qc.barrier()
qc.cswap(3,2,0)
qc.x(3)
qc.cswap(3,1,0)
qc.x(3)
qc.measure_all()
display(qc.draw(output="mpl"))
qc = qc.decompose()
result = AerSimulator().run(qc).result()
statistics = result.get_counts()
display(plot_histogram(statistics))
```

4. Conclusions

We presented an introductory course into quantum computing focusing mainly on the military cadets. The course is divided into two lectures, the first one is rather theoretical, and provides the necessary basis for orientation in this quite abstract theme. We chose the learning-by-example approach, instead of rigorous definition we gave a cadet a lot of examples of the structure and let the generalization on them. This could be beneficial as the mathematics text is quite often considered to be dense and not well readable. The second lecture is practical; cadets work in virtual notebooks. This allows instant response to their prompts and the work in the virtual notebook could be seen as sandbox, everything can be constructed, it is just on the user how deep they want to go. In this second part we proposed construction of two quantum circuits, first one shows the creation of fully entangled state, second is a thought experiment about solution of certain situation.

The entire course is intended to expand the standard syllabus of the first years of military studies, initially intended for those interested in the subject, but in the following years a similar course will become a necessity due to the rapid development of quantum technologies and their impact on cryptography, optimalization, cyber defence and military decision-making.

Acknowledgements. This work was supported by the Project for the Development of the Organization DZRO "Military autonomous and robotic systems" under Ministry of Defence and Armed Forces of Czech Republic.

References

1. https://www.ibm.com/quantum/blog/ibm-quantum-roadmap-2025 [cited: 9.2.2026]
2. https://quantumai.google/roadmap [cited 9.2.2026]
3. Michal Krelina, July 2025, Stockholm, SIPRI, https://doi.org/10.55163/ZVTL1529
4. Shor, P.W. (1994). "Algorithms for quantum computation: Discrete logarithms and factoring". Proceedings 35th Annual Symposium on Foundations of Computer Science. pp. 124–134. doi:10.1109/sfcs.1994.365700. ISBN 978-0-8186-6580-6.
5. Rivest, R.L.; Shamir, A.; Adleman, L. (1977). A Method for Obtaining Digital Signatures and Public-Key Cryptosystems (Technical report). MIT Laboratory for Computer Science. hdl:1721.1/148910. MIT-LCS-TM-082
6. https://www.ibm.com/quantum/qiskit [cited: 9.2.2026]
7. Krelina, M. Quantum technology for military applications. *EPJ Quantum Technol.* **8**, 24 (2021). https://doi.org/10.1140/epjqt/s40507-021-00113-y
8. Zhao, Q., Zhou, Y. & Childs, A.M. Entanglement accelerates quantum simulation. *Nat. Phys.* **21**, 1338–1345 (2025). https://doi.org/10.1038/s41567-025-02945-2
9. Unnikrishnan, C.S. (2022). Relativity and Quantum Entanglement. In: New Relativity in the Gravitational Universe. Fundamental Theories of Physics, vol 209. Springer, Cham. https://doi.org/10.1007/978-3-031-08935-0_15
10. https://www.python.org [cited: 14.4.2026]
11. https://jupyter.org/install [cited: 14.4.2026]