Programming with NEPO® follows a simple principle. The blocks are interconnected and executed by the robot according to their order. This principle is called »sequential operation«. Only blocks that are connected with each other are executed while running the program. The blocks that are connected together and to the start block are also called a sequence or »task«.

To connect blocks with each other, they have sequence connections and/or connectors. Depending on the mode of a block, the number of the connectors may vary.

Programming with NEPO is really easy. Every program begins with the »start« block.

This block is always added by default in the Open Roberta Lab workspace. The first block that you will execute is connected with the »sequence connector« of the »start« block. The sequence connector is a triangle at the bottom of the block and this triangle will become yellow as soon as a suitable block is in close proximity.

Your first program - Hello Roberta!

Now we are going to write our first program. This program will display »Hello Roberta« on the robot's display and the robot will drive forward.

- First select the show text block from the »Action« category. You can change the text in the block by simply clicking on the given »Hallo« text and writing your own text.
- Next add one more block for the robot to move forward, this block also can be found in the »Action« category.
- The program is finished!

In the next section we describe how to correctly capture the configuration of your robot, however this program only uses the motors from the standard configuration, therefore you can jump to the section Programming general#Starting a program on the robot.

Starting a program on the robot

To start your first program simply click on the start button in the lower-right corner of the screen. Your code will be compiled and sent to the robot to be executed.

- You will get a message on the top line of the Open Roberta Lab indicating that the program will start shortly.
- On the robot screen the »NEPO«-Logo will appear.
While the program is running on the robot, the robot icon in the left part of the Open Roberta Lab menu line will blink red and green.

The robot should display »Hello Roberta!« on the screen and drive forwards.

The program can be also started from: »Edit run«

NEPO®

NEPO is the name of our graphical programming language. It is easy and fun to learn programming. NEPO can also be used to program and interact with other robot systems.

Perhaps NEPO seems familiar to you. When designing NEPO we were inspired by existing concepts from scratch and blockly.

The graphical NEPO blocks have different components and thus different attributes. A block always represents a certain functionality. Its function can easily be determined by its block category, for example »sensors«. Features of the graphical blocks are:

Selection Menu

In the selection menu you can choose between different functions for the block, such as in block »drive« between forward and reverse.

Sequence connection

The blocks are interconnected using the sequence connection. A program consists of all interconnected blocks. The blocks will be executed according to their order, from top to bottom. A NEPO block can have more than one sequence connection.

Parameter

Input - Parameter Input

Depending on the block different values can be passed to a block. The type of the value can be identified by the colored compounds (called »connectors« ) of a block. In the chapter »block catagory « you can see the different values a block can have. Blocks can only be connected when the colors of the input and the output connector match.

Output - Return parameters

A block may optionally also pass a value (only one) to another block. These blocks have colored output connectors. Blocks can only be connected when the colors of the input and the output connector match.
Contextmenu

By rightclicking a block its contextmenu can be opened.

Several actions can be performed, using the menu:

- Duplicate (copies the block and connected input parameters)
- Add comment (adds a comment to this block)
- Inline Inputs (changes the appearance of the block)
- Collapse Block (minimizes the block)
- Disable Block (deactivates the block so that it won't be executed)
- Delete x Blocks (deletes the selected and all connected blocks)

Datatypes

There are seven different input and output types within NEPO. These types are as follows:

1. Logical value (light blue)
2. Number (dark blue)
3. String/Text (green)
4. Color (yellow)
5. List (purple)
6. Connection (pink)
7. Image (magenta)

The type «List» can contain a Logical, Number, String or Color type

Block Categories (Beginner, Expert)

A block-category groups blocks which have similar functionality. For example, the block-category "sensors" contains all blocks which can be used to program the standard LEGO® MINDSTORMS EV3 sensors.

The Open Roberta Lab offers two choices when selecting existing blocks:

- Beginner
- Expert

The selection “Beginners” is the default selection. This category includes the main program blocks which are most important for programming a robot.

Choosing "Expert " includes the three additional block-categories "Lists", "Functions", and "Messages" and additional NEPO blocks in other block categories.

NEPO offers eleven block categories:

| Block category | Beginner | Expert |
The description of the blocks and the available configuration options is based on the standard EV3 approach. They are described using the following scheme:

**Setting options:**
- Operation
- Enumeration of blocks belonging to the category
- Description of the individual blocks
- Settings of the blocks

### Programming Languages for Code Generation

<table>
<thead>
<tr>
<th>Robot System</th>
<th>Programming Language</th>
</tr>
</thead>
<tbody>
<tr>
<td>WeDo</td>
<td>Stackmachine JSON Code (It runs in the browser, editing is not recommended)</td>
</tr>
<tr>
<td>EV3lejos 0.9.0, 0.9.1</td>
<td>Java</td>
</tr>
<tr>
<td>EV3Dev</td>
<td>Python</td>
</tr>
<tr>
<td>NAO</td>
<td>Python</td>
</tr>
<tr>
<td>EV3 c4ev3</td>
<td>C/C++</td>
</tr>
<tr>
<td>Calliope mini</td>
<td>...</td>
</tr>
</tbody>
</table>

---
<table>
<thead>
<tr>
<th>NXT</th>
<th>C (NXC, not exactly C)</th>
</tr>
</thead>
<tbody>
<tr>
<td>micro:bit</td>
<td>MicroPython</td>
</tr>
<tr>
<td>Edison</td>
<td>MicroPython (very limited)</td>
</tr>
<tr>
<td>Bot’n’roll</td>
<td></td>
</tr>
<tr>
<td>BOB3</td>
<td></td>
</tr>
<tr>
<td>SenseBox</td>
<td>Arduino C/C++</td>
</tr>
<tr>
<td>mBot</td>
<td></td>
</tr>
<tr>
<td>NEPO4Arduino</td>
<td></td>
</tr>
</tbody>
</table>

Arduino C/C++ is actually C/C++, but a somewhat reduced selection from C/C++.
MicroPython is Python for microcontrollers.

**Block descriptions**

You can find the different blocks and their descriptions, ordered by categories [here](#).

For each block there is a detailed description of its functions, the different configurations, its parameters and return values.

**Example Programs**

In the section «Do you know how ...» there are a number of different sample programs. There is a brief description and a «model solution» for each program. Each example program is also described with the following criteria:

**Criteria:**

- robot
- motors
- sensors
- block category
- difficulty
- note

The following programs are available:

- Driving a square
- Cornering
- Controlled program termination
- LED status display
- Own blocks, macros, methods
- Line follower