

NXT Brief Documentation

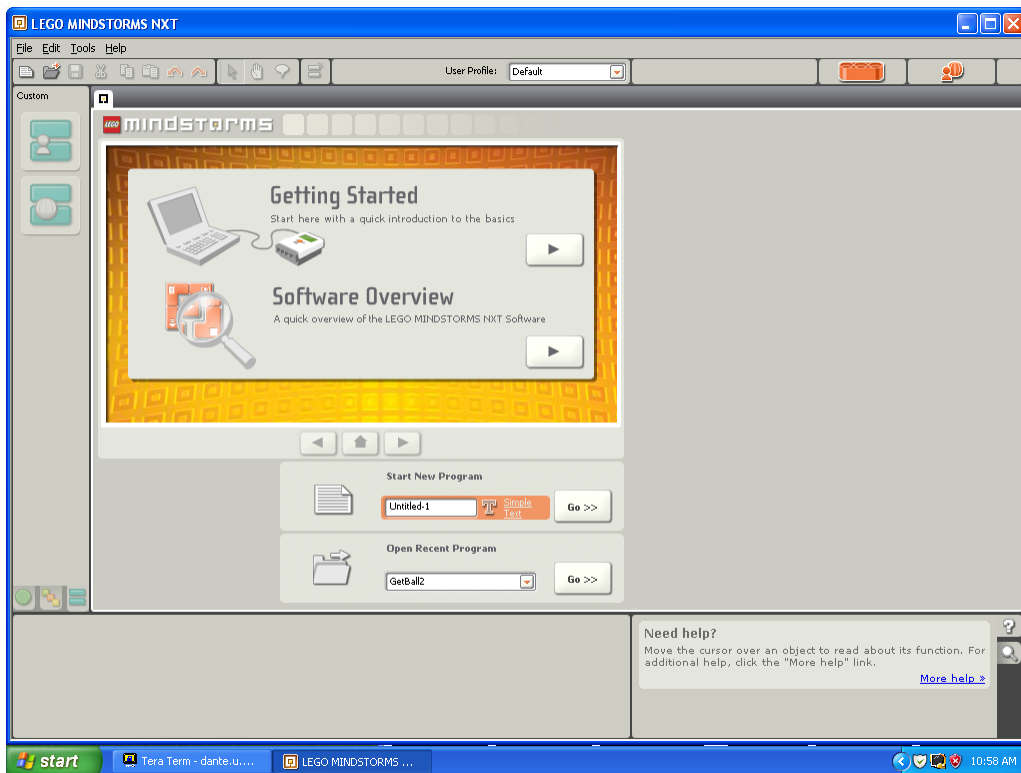
This document is meant to give an general overview of the NXT programming environment and to help people get started programming quickly.

Start Up

Double click this icon to start the NXT Mindstorms programming environment:



This is the start up screen that should appear (or something similar for other operating systems):

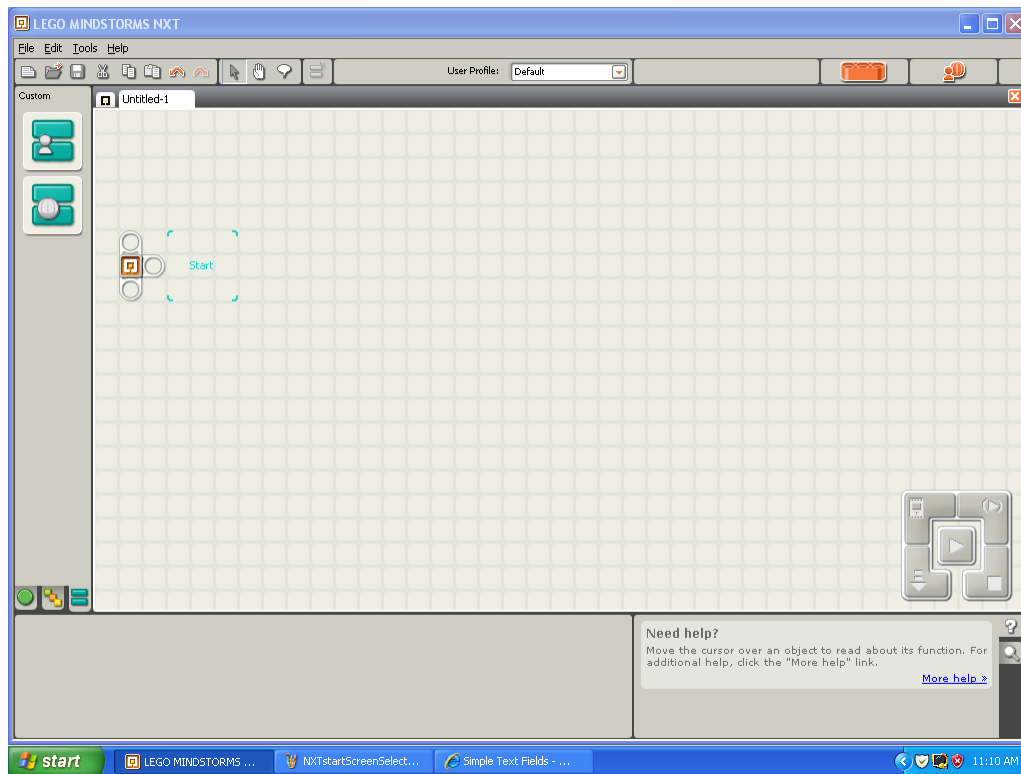


The start up screen gives two options: a tutorial which gives an general tour of the NXT programming environment and and two choices for programming.



The first option is for starting a new program. One simply types a new name or one can just use the default name (if one uses the default name, the program will ask for a file name when saving). The second one is for editing a program that was already created and save from a previous session.

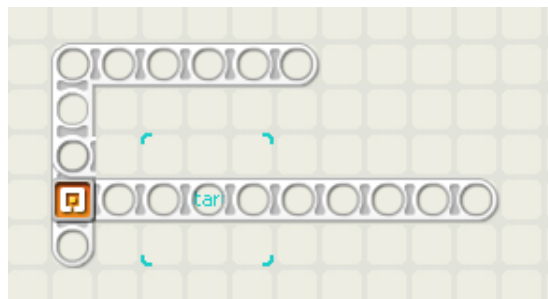
If one decides to create a new program, the following screen appears:



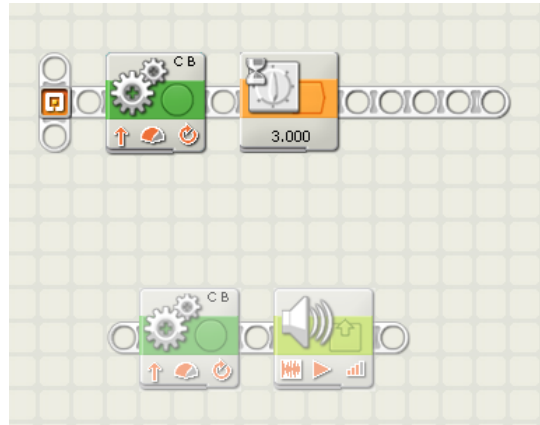
The blank area is used for laying out blocks to construct programs. The blocks for assembling programs come from the **block menu** at the far left (more about this later). Programs start from a **start block** as shown here:



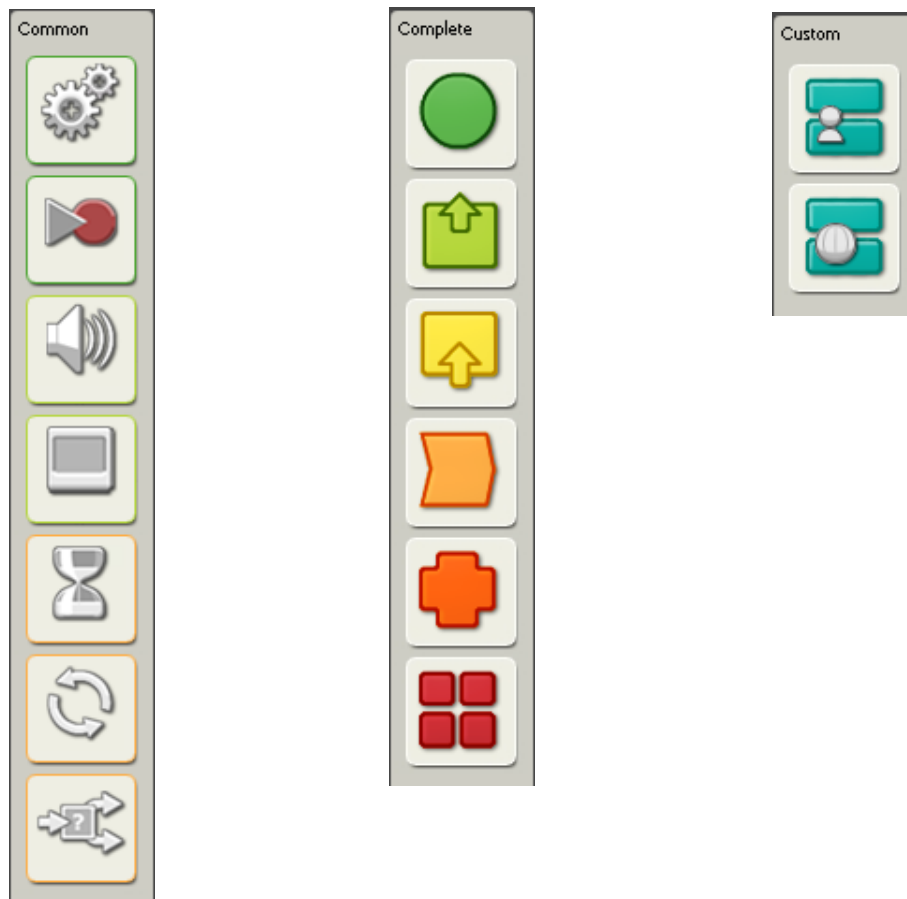
Simply select a block from the block menu and connect it to the start block. The three nodes on a start block allow one to connect three simultaneous sequences of blocks though people don't commonly do this. One can extend these nodes as needed.



All blocks must connect to the start block in order for them to be an active part of a program though they don't need direct connections. They may be connected through other blocks. Block that with out a connection to a start block will appear faded so finding unconnected blocks is easy.



The block menu actually contains three menus called *common*, *complete*, and *custom menus*.

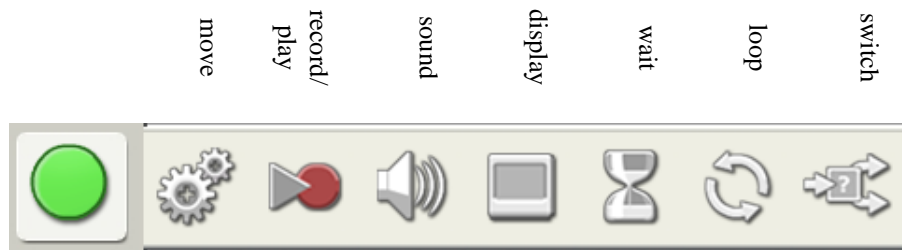


One can switch among the three menus by clicking on the small picture icons below the menu:

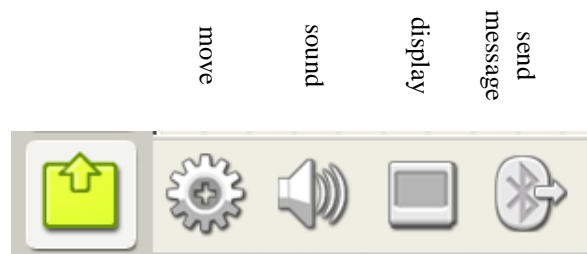


The common menu has commonly used blocks for quick access while the complete menu contains all the blocks built in to the NXT environment including all of the common menu's blocks. Unlike the common menu, the items in the complete menu each have a sub menu with groups of blocks organized as six categories: common, action, sensor, flow, data, and advanced blocks.

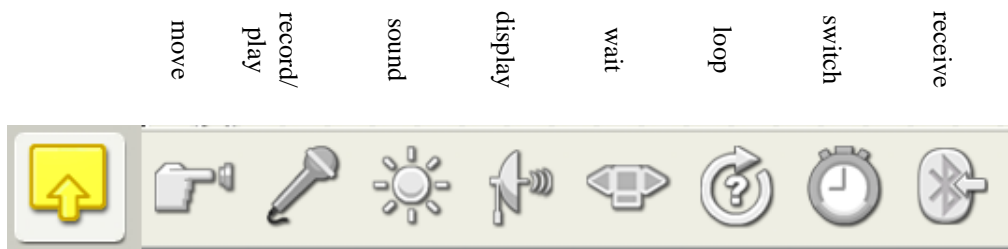
Common blocks: these are the same blocks as the one ones in the common menu.



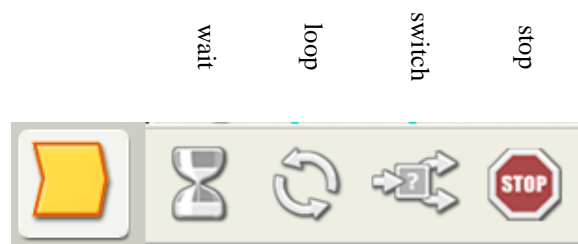
Action blocks: give commands to various parts of a robot



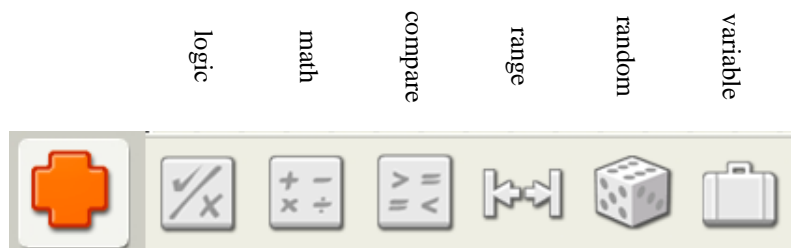
Sensor blocks: these are used to configure how sensors operate and communicate with a robot.



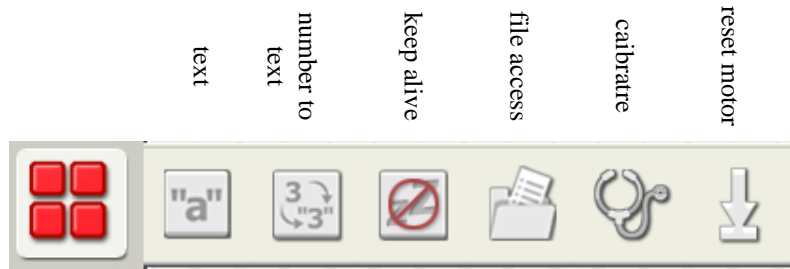
Flow blocks: flow blocks control program flow and operation



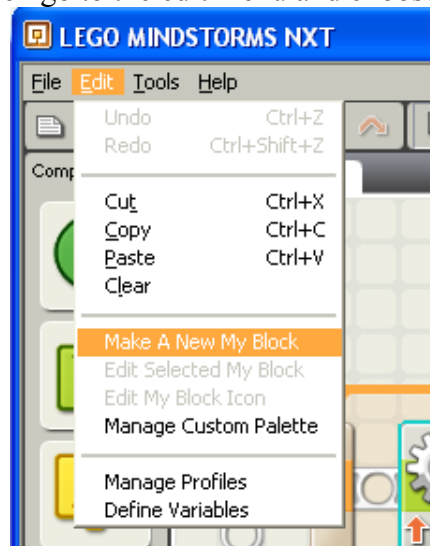
Data blocks: has basic math operators for boolean logic and data processing



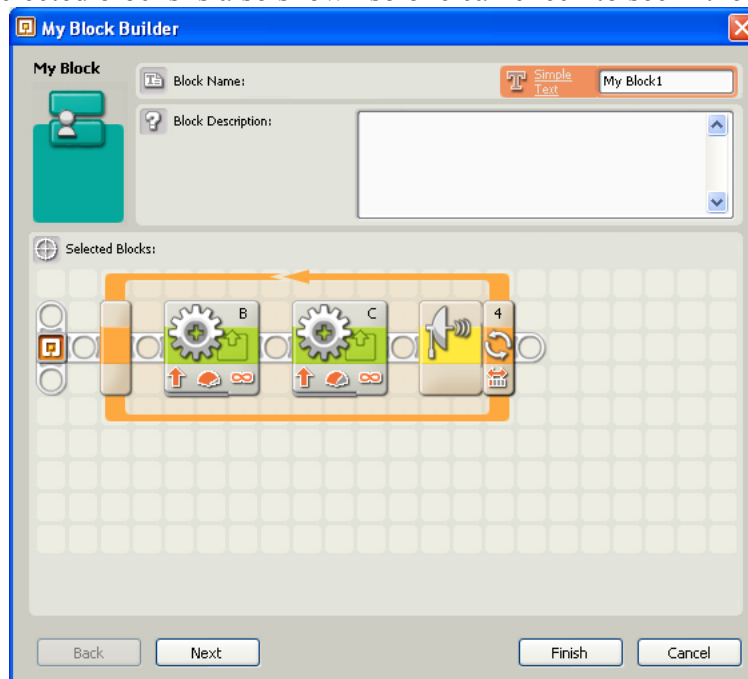
Advanced blocks: for other operations such a storing or displaying data, checking the sensors, etc.



A series of charts is provided at the end of this document that gives a brief description of all the blocks. The last menu is the **custom menu**. This menu has two blocks: my blocks and web downloads. My blocks are created from custom combinations of blocks that act as a group or module. Select an appropriate group of blocks and then go to the edit menu and choose “Make A New My Block”.



Then the My Block Builder interface appears. Here one can give the new block a description and a name. A view of the selected blocks is also shown so one can check to see if the blocks are correct.



Press next to composite an icon for the custom block by dragging images from the given icons.



Finally after pressing finish a new block appears in place of the selected blocks.



and also appears in the custom block menu to be used like any other block.



The web downloads are simply blocks downloaded from the NXT web site. When they are added they appear in the web downloads sub menu.

Another important part of the NXT programming environment is the *controller*.

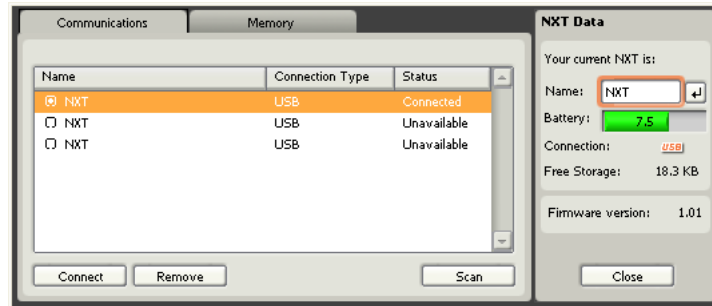


The controller has five buttons. The controller buttons work only if a NXT brick is connected by Bluetooth or a USB cable to a computer running the NXT environment.

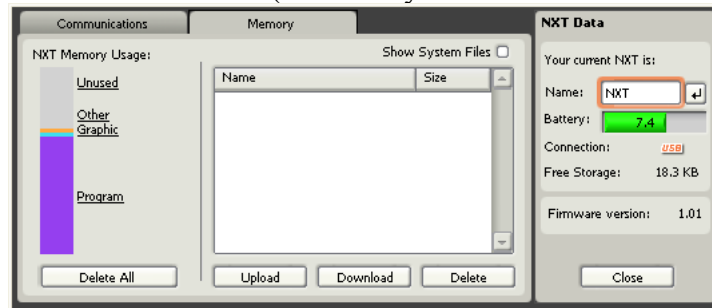
The first button is the **NXT window button** located at the top left corner.



Pressing this button reveals the NXT bricks that are available for connecting and includes both USB and Bluetooth connections and can be used to check connections. Below shows the communications panel.



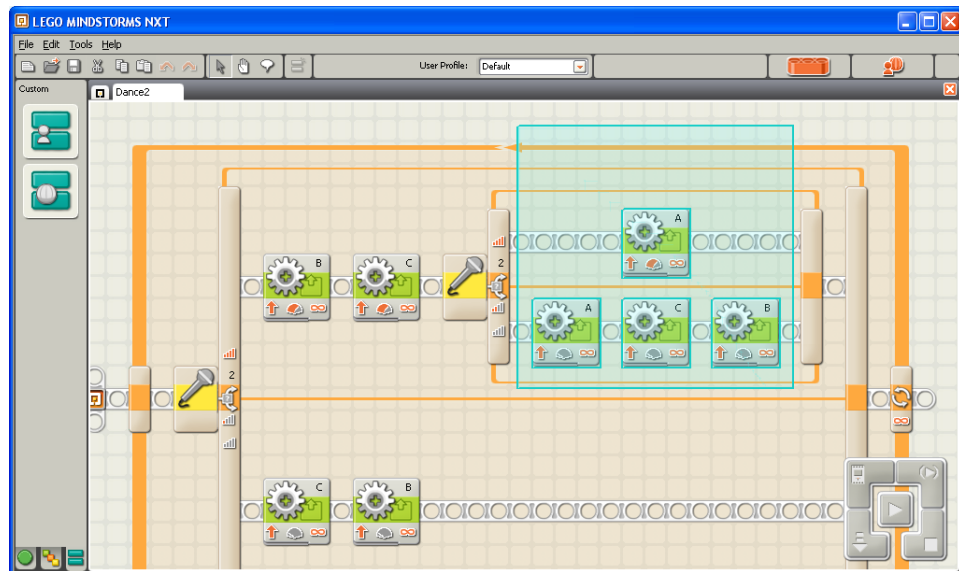
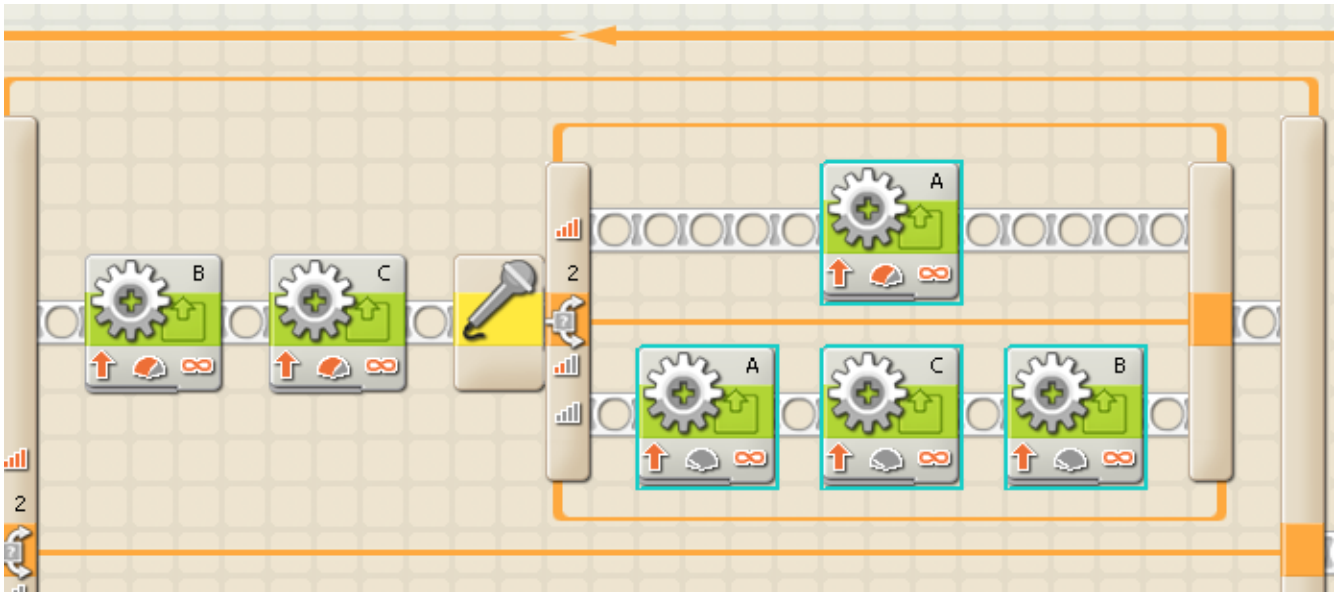
One can also check status of their robot's memory by clicking on the memory tab. This is used for managing programs and one can delete them as necessary to free more memory. The right side displays data about the currently selected NXT brick (or the only connected brick if there is only one).



At the left top corner is the **run selected button** for selecting a part of a program and testing it.



Simply highlight a group of blocks with the arrow tool as shown below and then press this button to see how that group of blocks works. Selected block appear highlighted with blue outlines.



It is very useful for debugging and testing specific program modules and cases without needing to run through an entire program. The next controller button is **download button**. One uses this to compile and download programs without running them to an NXT brick.



In order to stop a program running on a connected NXT brick, press the **stop button**:



It only works for a connected NXT brick. If a robot is not connected, pressing this button will do nothing. The final button in the middle of the controller is the **run button**.



Pressing this button compiles the current program and (if there are not errors) downloads immediately runs it.