

# NXT Quick Start Guide

by Rob Widger



# NXT Quick Start Guide

## Contents

Introduction.....	2
1. A simple approach .....	3
2. Setting up the hardware .....	4
3. Using the NXT without software.....	5
4. The NXT software – writing a simple program.....	8
5. Activities to try out – driving the NXT model.....	11
6. The Programming Blocks.....	13

## Introduction

The intention of this Quick Start Guide is to enable users to program the NXT within minutes of building the model. This can be achieved by programming the NXT with the software on a PC/Mac, or by using the onboard programming utility found in the main menu on the NXT. Don't underestimate the power of the onboard programming tools. They are very effective at introducing the sensors that come with the NXT base set.

The simple approach outlined in section 2 gives new users a format to follow in the first session with the NXT and NXT software.

### A note about sound:

The NXT has the ability to play sounds. This is great when doing some control tasks such as a burglar alarm, police car or level crossing. However, in the early stages (i.e. when using the Quick Start Guide) we recommend that you avoid using sound in your programs in order to avoid having to learn about the memory management system.



**The Sound Block:** Avoid in the early stages of programming but it is great when you know more.



# 1. A Simple Approach

Follow the next 5 steps to start your learning process with the NXT. This is a suggested way of starting off with the NXT and finding success quickly.

## 1. Build the Model

- Standard NXT model – guide found inside the 9797 box; or
- DACTA Simple Model (PDF available from your local dealer. The Simple model, designed by DACTA, can be made with the existing elements from the 9797 base set).

## 2. Explore the Try Me Programs on the NXT

- Try Sound – see the NXT move faster as the noise gets louder
- Try Touch – watch and hear the NXT change when you press the touch sensor.

## 3. Write a Program on the NXT

- Using the NXT program facility on the NXT
- Following the Output/Input/Output/Input/Finish approach

## 4. Write a Simple Program using the NXT Software

- Use the blocks on the Common palette in the software to create a very simple program

## 5. You have now covered the basics and are ready to move on

- Use the Robot Educator to learn about all the programming blocks
- Purchase the DACTA NXT Guide with Curriculum Ideas and programming instructions (English only).

## 2. Setting up the Hardware

### Invest in sorting!

Once you have sorted the elements into the sorting trays, shown opposite, find the building book and start to construct your first model. The NXT is ready from opening the box, the firmware is installed and ready to use.

### Why sort?

Building the models is quicker because LEGO® elements are easier to find.

Now the elements are sorted, start building the LEGO NXT model found in the build guide supplied in the box. Alternatively see the model below.



### A simplified model? You can choose!

The NXT model that is promoted by LEGO Education is superb for all aspects of school work. It is strong and versatile but does take a little time to construct.

There has been a wish expressed by our users to have a very simple model that can be built in less than 15 minutes. Therefore a model has been designed for users to use. A PDF guide is available from your local dealer.

#### The key features of the NXT Simple Model are:

- A model that can be built in under 15 minutes (the basic model just using the ultrasonic sensor and not the full range of sensors)
- The ultrasonic sensor is low to the ground and is more efficient for driving tasks.
- Strong, sturdy construction.
- Limited use of LEGO TECHNIC elements
- Easy to transport and store. Sits comfortably in the box of the 9797 set.

Whichever model you choose you are now ready to explore the features of the NXT.



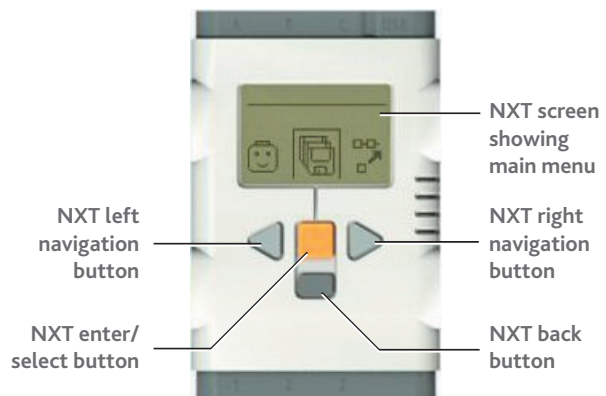
# 3. Using the NXT without software

With a NXT model built it is now time to turn on the NXT and start using it without connecting it to a computer.







- 1 Turn the NXT on by pressing the orange **Enter** button, in the centre of the brick.
- 2 The main menu will be displayed, as shown on the NXT below. Use the left and right arrow keys to explore the various NXT options. Press the orange **Enter** button to select. See the chart at the bottom of the page for an overview.

### How the NXT brick works

The brick works in a similar way to a mobile phone... it is menu driven and files are stored in folders.



### The main menu icons

					
Settings	Try Me	My Files	NXT Program	View	Bluetooth
Change sound levels. Sleep mode and delete files.	A series of sample programs to test out each sensor.	Your programs and sounds are stored	Program simple actions on the NXT using buttons	View all the sensors attached to the NXT	Find and connect to other Bluetooth devices
Probably a section not needed in your early stages.	Great for showing all the sensors in action	Once programs are downloaded they can be run again through here.	A simple program feature to use the NXT without software...	Useful when wanting to use the NXT as a meter. Great for writing programs with sensors	Recommended for confident users. Stick with USB in the early stages.

### Programming without using a computer

Although the NXT was designed to be used with a computer and software, it can be used to carry out some simple programming/cause and effect tasks using the icons below.



Try Me Programs



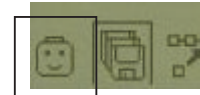
NXT Program

## The Try Me Programs

The **Try Me** section allows pupils to explore how the various NXT sensors work. It is not a programming utility but a 'cause and effect' utility. Each sensor will cause the NXT to do something when that sensor is 'active'. The most impressive is the **Try-Sound** program.

### Trying out the Sound Sensor

- 1 Using the navigation buttons, highlight i.e. move to the centre the **Try Me Programs** icon (the LEGO® face).
- 2 Press the orange **Enter** button
- 3 Try-Touch will be shown; press the right navigation button until Try-Sound appears.
- 4 Press the orange **Enter** button twice to run the program.



STAY REALLY QUIET! Then make a noise. The robot will move! Then experiment with volume to see the robot move faster or slower depending on the volume.

This is a great activity to demonstrate 'cause and effect'; one of the first activities that we show pupils.

### Other TryMe Programs

**Try-Touch** – When pressing the touch sensor attached to Port 1 a face on the NXT screen will say "whoops" when the touch sensor is pressed.

**Try-Ultrasonic** – a warbling sound will play and the tone will rise if an object gets near to the ultrasonic sensor and fall, the further the object moves away...

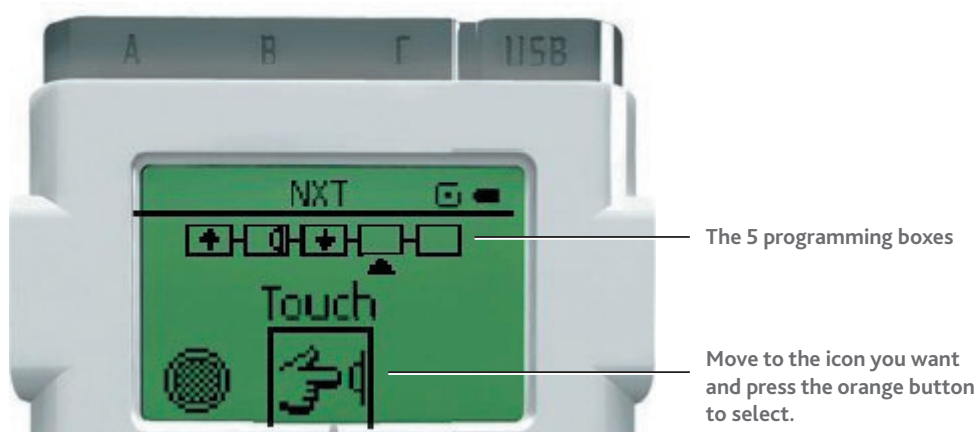
**Try-Light** – The light sensor looks at the colour of the object it is facing and makes a high pitch sound for light objects and a low pitch sound for dark objects.

## The NXT Program Menu

A simple 2 step program can be written following the process of **Output, Input, Output, Input, Next**. For example, drive along until the sound sensor hears a sound. Then reverse until the touch sensor is touched. Stop at this point.

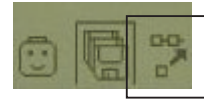
The program is written by making selections in the bottom half on the NXT screen. The choices made by the user are displayed in the 5 boxes at the top of the NXT screen.

In the example below, three of the five boxes have been programmed and the user is about to select a touch sensor to use.



## Accessing the NXT Program Menu

- Go to the **NXT program** screen, as shown opposite, and click the orange **Enter** button. The first screen shows how to connect the sensor then the user is presented with the programming screen, as shown on the previous page.



Below are just a small selection of the programming icons available but give you a taster of what the onboard programming can do.

Outputs		Inputs		Next	
	Forward		Dark		Loop
	Forward 5		Light		Stop
	Right Turn		Touch Sensor		
	Right Turn 2		Wait 2		
	Left Turn		Wait 5		
	Left Turn 2		Wait 10		
	Backward				
	Backward 5				
	Tone 1				

Try out the following programs to see what happens...

Output	Input	Output	Input	Next?	
Forward 	Wait 2 	Turn Right 2 	Empty 	Stop 	The NXT will drive forward and then turn approximately 90 degrees. Now replace the STOP command with the LOOP command. Do this by pressing the back button on the NXT.
Forward 	Light 	Backward 	Wait 2 	Loop 	With the light sensor attached and facing the floor. The NXT will drive forward until the floor becomes light. It will then move back for 2. Then it repeats the same actions.
Turn Right 2 	Empty 	Tone 1 	Wait 2 	Loop 	The NXT turns 90 degrees and then plays a tone. The NXT will then wait 2 before looping.

Simple 2 step programs can be achieved using time and the sensors. These programs can be saved onto the NXT for later use. To edit program users must use the dark grey **Back** button to erase steps. This does mean deleting latter stages of this simple program.

Try out some of your own! Can you...

- Drive the NXT forward until it sees a wall and then reverse for 2?
- Write a program to respond to sound?

# 4. The NXT Software

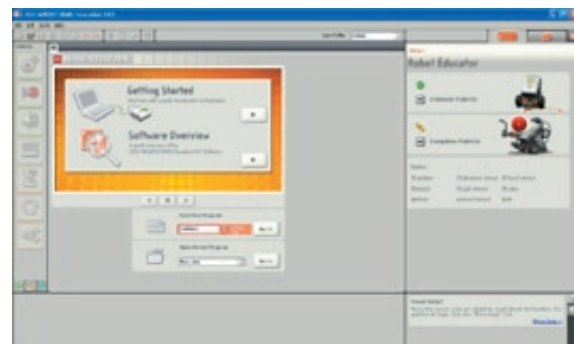
## Before you begin

- 1 If you haven't already done so, you need to install the NXT software onto the computer.
- 2 Double-click on the **NXT** icon to launch the software.
- 3 Plug one end of the USB cable into a USB socket on the computer and plug the other end into the NXT. A window will pop up in the bottom right hand side of the screen saying your NXT is installed and ready to use.

You can now program the software to make the NXT robot move.

## Main Screen Overview – The launch screen

Here you can watch videos and learn more about the software, start a new program, or explore the Robot Educator.



## The Toolbar



Much improved from the previous software, the toolbar now works very similar to the windows toolbars found in standard Windows applications.

## Palette Area

There are three palettes in the software that are used for programming the NXT. They are the Common, Complete and Custom palettes. These palettes are the core of the NXT programming environment.



The Common Palette is all you need for this guide as the Complete Palette is for later, more advanced programming. The Custom Palette allows users to easily create their own blocks.

For all your programming in this guide you will only need to know the following blocks on the common palette:-



Move Block



Wait for Time Block



Wait for Ultrasonic Block



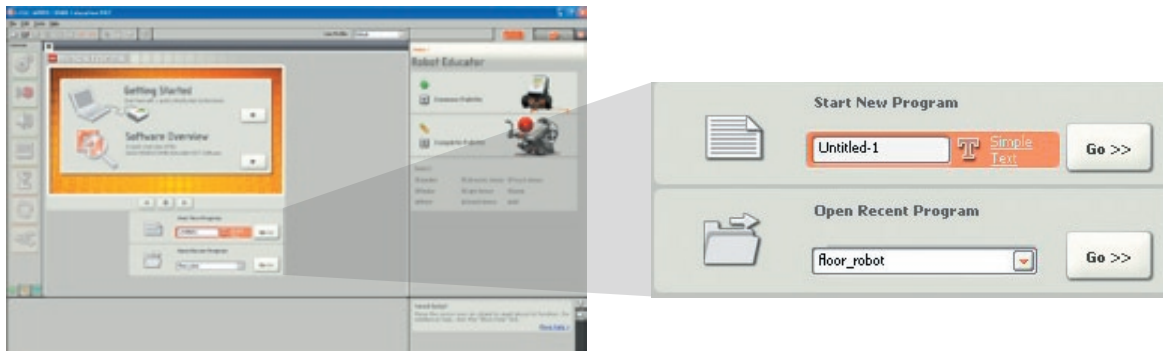
Loop Block



Switch Block



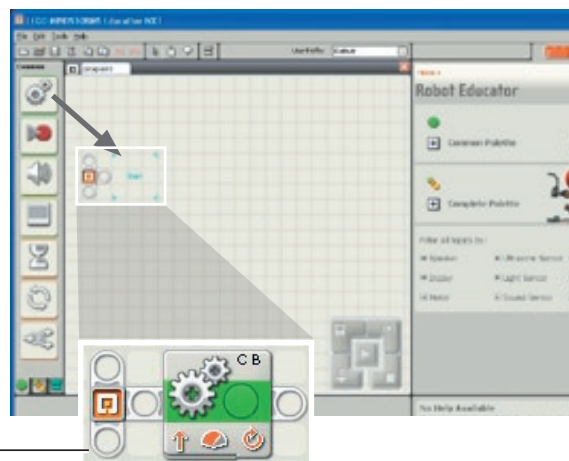
# Writing Your First Program



- 1 To start programming click in the text box under **Start New Program** and enter a file name.
- 2 Press **Go>>** to start.

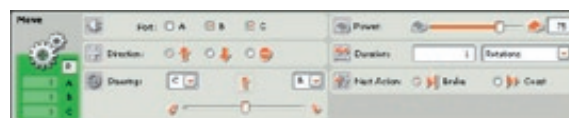
- 3 Programs are made by grabbing blocks off the palette on the left of the screen and dropping them into the main programming window.

The Move block, at the top of the palette, is the best one to start with. Click and drag it to the start space next to the program start symbol, as shown opposite.



The block drops into place and is ready to test.

- 4 When you drop the Move block into the programming window, the Move block menu appears at the bottom of the screen. Here you can change direction, speed and duration of the motors on the NXT robot. Try changing the variables – you can spend ten minutes exploring the different ways of making the NXT move.



- 5 With the Move block in place, click on the **Download and Run** button



**NOTE:** Now you have downloaded your first program the NXT has stored your program in its memory. Pull out the USB cable from the NXT. Notice on the NXT screen is the program you have downloaded to the NXT. Press the orange button to run the program again. This means that you can take the NXT away from the PC and run program. When you are ready to program again simply plug the USB cable back in.

Well done!

## The Robot Educator – a must read for new users

The Robot Educator, on the right of the screen, will teach you all the blocks that you will use. **It is a must for all new users to the software.**

The Robot Educator has 39 activities to teach new users how to program the NXT with the software.

When you open up an activity the Educator provides you with a simple challenge brief (normally in the form of a flash animation or slide show), a building guide (hopefully your model is already built) and a programming guide that takes you through one possible programming solution to the challenge, step by step.



Challenge Brief



Programming Guide



Building Guide

Try out this simple activity found in the Robot Educator (it is driving the NXT in a square):-

- 1 Click on the **+** sign next to Common Palette.
- 2 Select **Activity 8**.
- 3 Watch the Challenge brief.
- 4 Follow the programming guide frame-by-frame as you program.
- 5 Download the program and run.

This is a great way to get started and you can learn at your own pace.

# 5. Activities to try out

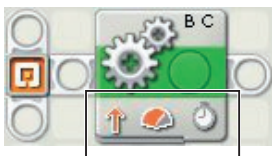
You have managed to make the NXT model move for a limited amount of time. What the NXT can do is only limited by your imagination and programming skills. These two aspects will develop!

The activities in this book are limited to the Common Palette only. They are intended to give the user a brief glimpse of programming using motors with time and/or sensors in the form of a 'Wait For'. The Wait for Blocks are coloured orange and are selected by clicking on the egg timer menu in the Common Palette.

The blocks covered in this guide: **Move, Wait For Time, Wait For Ultrasonic, Loop and Switch.**

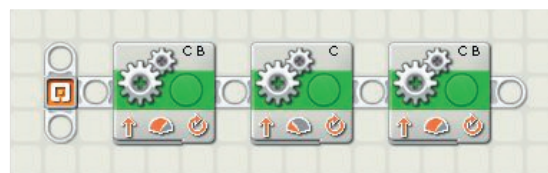
## ACTIVITY 1: Driving, Turning and Stopping

Program the robot to drive in a straight line, then stop followed by turning round and moving back. No sensors are needed for this. For duration keep to using the rotation selection. You can explore the use of other choices later.

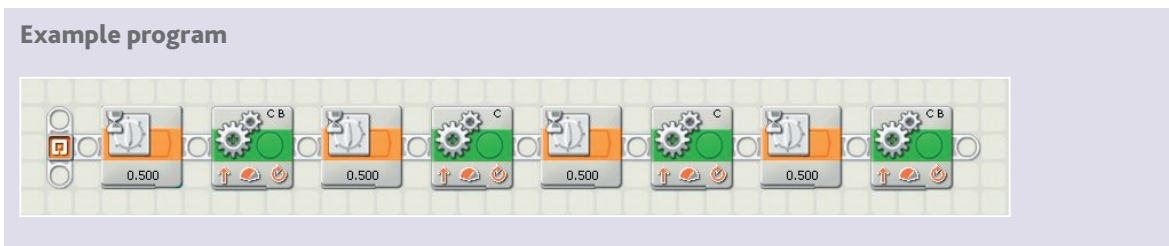


The block in the programming area gives users a basic overview of what the block will execute. Here the NXT will turn on **Outputs B and C** (meaning it will go in straight line) in a **forward** direction, with a **power level of 75%** for period of **time**. The block choices happen at the bottom of the screen.

- 1 Program the NXT to move forward at full power for 1 rotation.
- 2 Turn the NXT approximately 90 degrees. Notice in the example only one output is turned on. This will turn the robot for 1 rotation at 75% power. As only one motor is used the NXT will turn approximately 90 degrees.
- 3 Finally drive the robot forward at 25% power for 1 rotation.

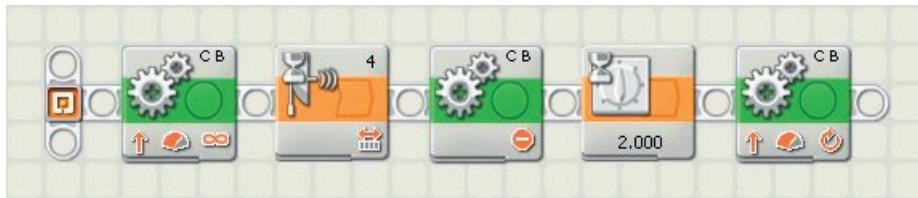


Notice that the NXT does not stop after each block and only stops at the end of the program. For Pauses the Wait For Time block needs to be inserted into the program line as shown in the example below.



## ACTIVITY 2: The Obedient NXT

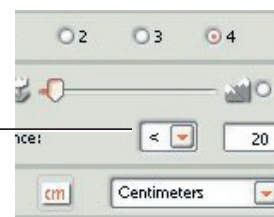
Program the NXT to move forward until it sees an object 20 cm away. The NXT will then stop, wait for 2 seconds and then move forwards again for 1 rotation.



### Notes when programming

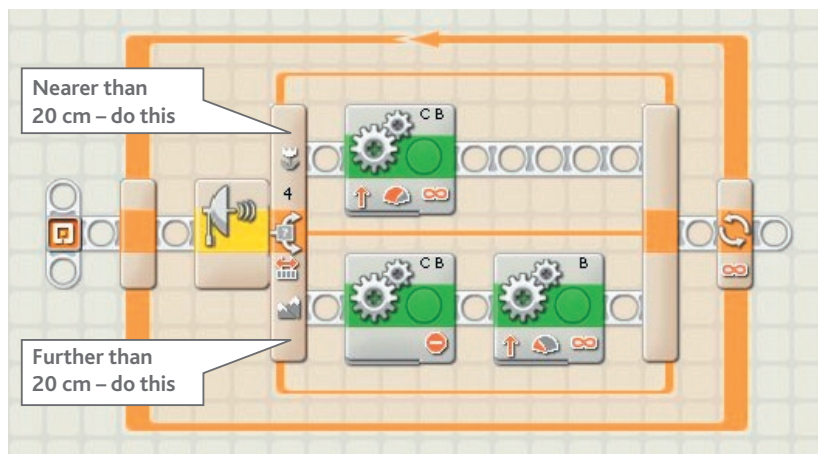
Notice the infinity symbol on the first Move block. Here the duration has been changed from rotation to unlimited. This means it will continue until it receives an input. The input is the Ultrasonic wait for block that has its trigger point set to less than 20 cm.

A second move block is needed with the Stop symbol selected rather than a directional arrow. We then pause for 2 seconds before using a third move block to finish the activity.



## ACTIVITY 3: The NXT Guard Dog

In this activity the ultrasonic sensor is used to guard an area. The NXT will spin very slowly until an object comes into range (less than 20 cm). When the NXT detects an object it will move towards that object. If the object moves out of range then the NXT will start spinning again.



Loop Block



Switch Block

### Notes when programming

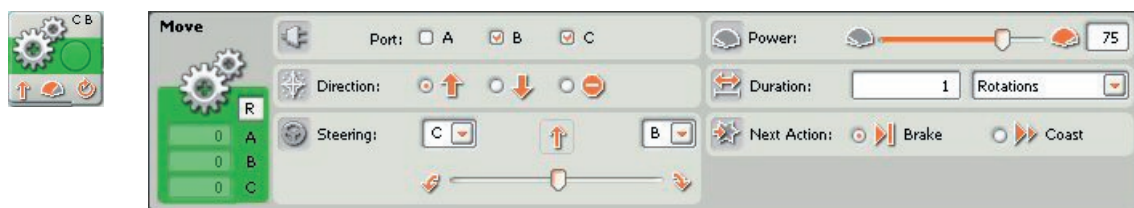
For this activity we introduce two new blocks. The Loop block and Switch block (see above). When writing a program we wish to repeat a loop is the first block we use. Drag one into the programming area and then drag a switch into the loop.

On the switch select the sensor you want to use and set the trigger point. For this example we are selecting the ultrasonic sensor. Once the trigger point is set the top line refers to when it is closer than the trigger point and the bottom line is when it is further away than the trigger point. Standard flower and mountain symbols are used. Now drag the actions you want to happen when the object is near and far.

# 6. The Programming Blocks

The Most important Block in the Software (plus one or two others!)

**The Move Block** *"One block will drive your buggy any way you want it to! It is one of the most powerful blocks in the software!"*



## Summary of Duration

There are four choices to make for the duration of the motor(s) to stay on:-

- **Unlimited** – The motors will keep going and going until they are told to stop by another block, usually a sensor. (see below)
- **Degrees** – The motor will turn its axle for a number of degrees.
- **Rotations** – The motor will turn its axle for a number of complete rotations.
- **Seconds** – The motor will turn for a number of seconds.

## The Loop Block



By default the loop is switched to forever. For anything else the user has to change the settings. Blocks are placed inside the Loop block to create a program. The Loop block is one of the most common blocks used in programming. When the loop menu appears you can change how it loops in the following ways:

**Forever** – repeat continuously, the program will not stop.

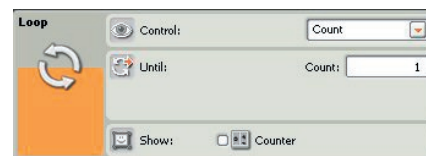
**Sensor** – repeat until a sensor is triggered.

**Time** – repeat for a specific time period only.

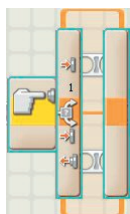
**Count** – perform a specific number of loops before stopping.

**Logic** – Is the data received true or false.

Here is an example menu for counting loops, for example, do 1 loop and then stop.



## The Switch Block



The switch block is a classic 'yes/no' line of programming. Has the touch sensor been pressed or is the light sensor less than 50%? The switch block can be controlled by sensors or values. The sensors available are light, NXT buttons, rotation sensor, sound sensor, timer, touch sensor, ultrasonic sensor, light and temperature sensors. Once the switch block is placed other blocks must be dropped into the upper and lower areas of the switch block for differences to be seen.

## The Wait For Palette

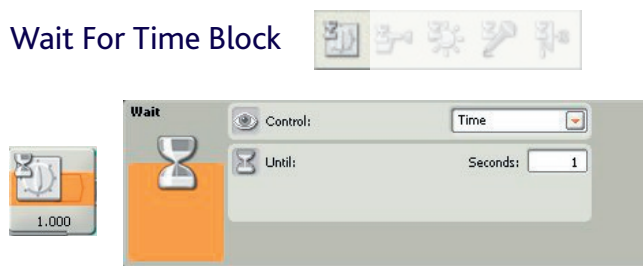


The only block in the Common Palette to have a sub-menu of five blocks. When building the standard model the build guide directs the user to insert the sensors into a default port. These ports are fixed within the NXT software and have to be changed by the user if desired.

### How Wait For Blocks work

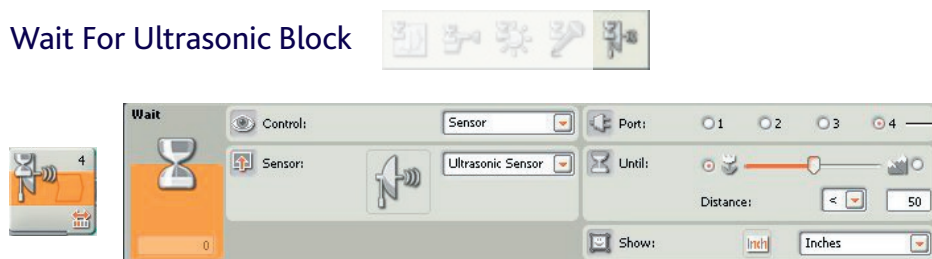
The Wait For block is a pause in the program where the NXT waits for a specific sensor or time to be activated or elapse. For example, drive the buggy in a straight line, wait for the touch sensor to be pressed (when the NXT bumps into a wall) and stop.

## Wait For Time Block



A simple but powerful block to use. Drop the block into the programming area and enter a time in the menu at the bottom of the screen. This will, in effect, create a pause.

## Wait For Ultrasonic Block



### The Ultrasonic Sensor

The ultrasonic sensor, by default, is set to Port 4. Make sure the sensor on the model is connected to port 4 of the NXT.

Change the units of measure to centimetres. The trigger point is set by dragging the slider to the desired value. Is the wait for measuring objects getting closer or moving away... this is selected using the < > less than or greater than symbols.

## Epilogue

I hope you have found this guide useful. You have covered just 0.5% of what MINDSTORMS® can do. There is so, so much more to explore with the NXT software; this is only just the beginning of a superb journey into robotics. I would recommend the Robot Educator as a useful way of learning the blocks and if you want curriculum ideas then I have written a book called NXT User Guide and ICT Curriculum Scheme of Work. This book is based around the UK Curriculum but the ideas could easily be used in any country using the NXT. Good luck, happy programming! Rob Widger, Education Manager.