

Preface

About SunFounder

SunFounder is a technology company focused on Raspberry Pi and Arduino open source community development. Committed to the promotion of open source culture, we strive to bring the fun of electronics making to people all around the world and enable everyone to be a maker. Our products include learning kits, development boards, robots, sensor modules and development tools. In addition to high quality products, SunFounder also offers video tutorials to help you build your own project. If you have interest in open source or making something cool, welcome to join us! Visit www.sunfounder.com for more!

About This Kit

This learning kit focuses on the popular open source platform Arduino. You can learn the knowledge of the Arduino servo by applying this kit.

In this book, we will show you how to build the bionic robot via description, illustrations of physical components, in both hardware and software respects. You may visit our website www.sunfounder.com to download the related code and view the user manual on [LEARN -> Get Tutorials](#) and watch related videos under [VIDEO](#).

Free Support



If you have any **TECHNICAL questions**, add a topic under **FORUM** section on our website and we'll reply as soon as possible.



For **NON-TECH questions** like order and shipment issues, please **send an email to service@sunfounder.com**. You're also welcomed to share your projects on FORUM.

Contents

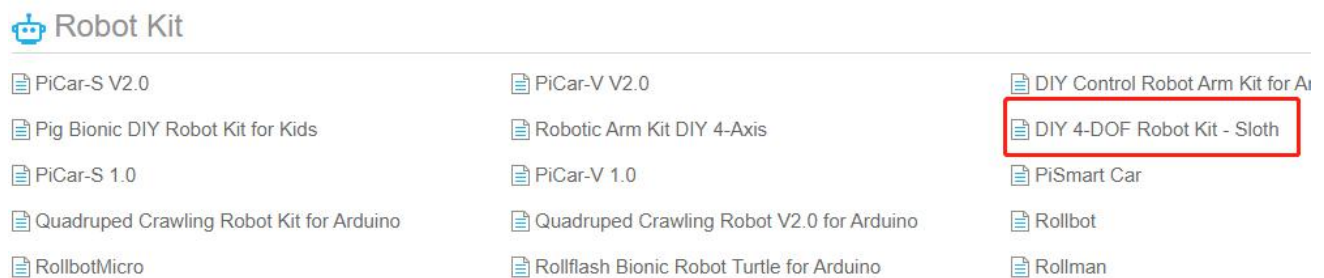
Download the Code.....	3
Get Started with Mixly.....	2
Overview.....	2
Download the Mixly Software.....	2
Open the Software.....	2
Brief Introduction.....	3
Blocks.....	3
Code.....	8
Tools.....	8
Install the Driver.....	10
Add Library.....	11
Lesson 1 Moving Forward.....	12
Lesson 2 Moving backward.....	16
Lesson 3 Moving Forward in S-shape.....	20
Lesson 4 Test Ultrasonic Module.....	24
Lesson 5 Obstacle Avoidance.....	26
Lesson 6 Dance.....	31

Download the Code

Step 1

Go to our official website <https://www.sunfounder.com/> by visiting **LEARN** -> **Get Tutorials** -> **DIY 4-DOF Robot Kit - Sloth**.

Note: The location of the kit will change, but the name will not change.

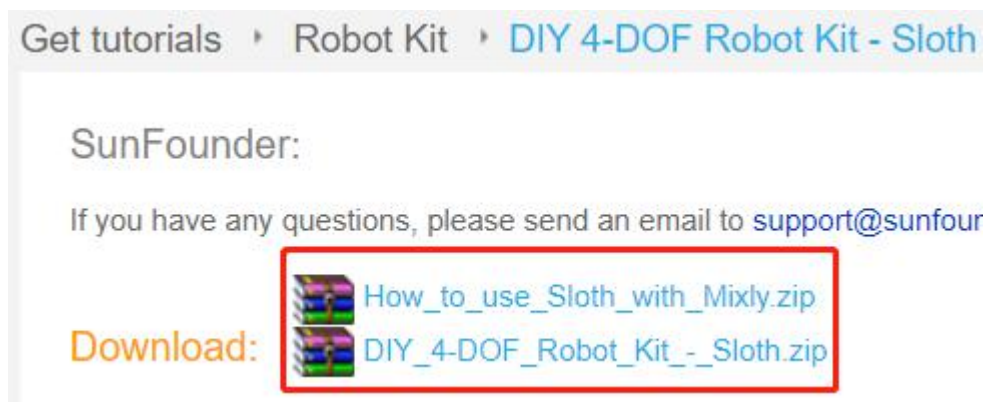


You can also directly download the file by visiting the **following link**:

<https://www.sunfounder.com/learn/category/DIY-4-DOF-Robot-Kit-Sloth.html>

Step 2

After entering the **DIY 4-DOF Robot Kit-Sloth** category, you can see 2 zip files. It is recommended to download **DIY_4-DOF_Robot_Kit_-_Sloth.zip** first, including the Arduino code, drivers, library, schematic and user manual.

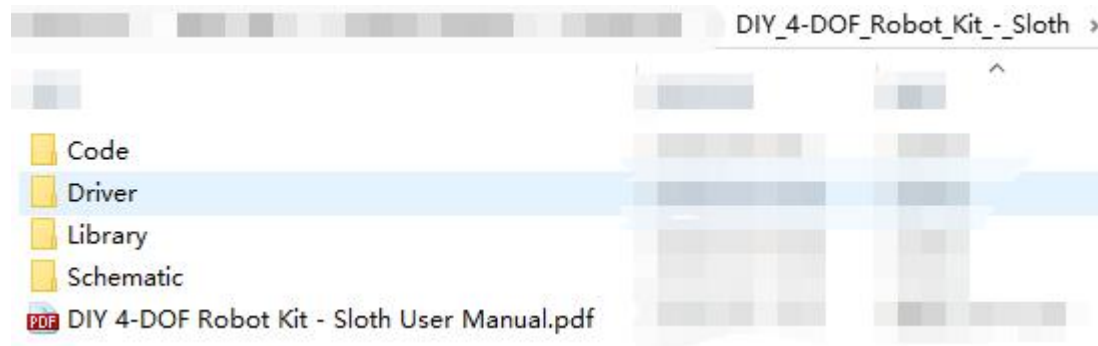


- **How_to_use_Sloth_with_Mixly.zip:** This is an extended use of Sloth via graphical programming software - Mixly to program. The manual contains only a basic introduction to Mixly and how to program in Mixly, but does not include hardware assembly part, you need to download **DIY_4-DOF_Robot_Kit_-_Sloth.zip** first, and after Sloth is completely assembled, you can use Mixly for programming.

Step 3

Unzip the download files.

➤ **DIY_4-DOF_Robot_Kit_-_Sloth.zip**



Code: The Arduino code to make Rollarm work.

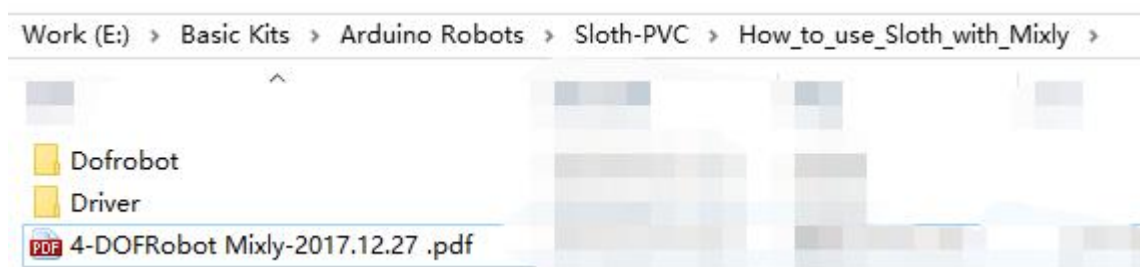
Driver: The Nano board driver software for Windows and Mac OS X system.

Library: The library of Ultrasonic sensor module.

Schematic: The schematic of the Servo Control board.

..User Manual.pdf: Sloth's complete manual, including Arduino installation and basic use, Sloth's hardware assembly and code use.

➤ **How_to_use_Sloth_with_Mixly.zip**



Dofrobot: The library that you need to add in the Mixly.

Driver: The Nano board driver software for Windows and Mac OS X system.

..User_Manual.pdf: The user manual of how to program in Mixly.

Get Started with Mixly

Overview

Mixly is a graphical programming software with some blocks embedded in. You can program by dragging and dropping blocks to combine together like building blocks, thus enabling a complete process of the same sketch in IDE. This simple programming software is aimed to help those who don't want to learn or don't know programming.

Download the Mixly Software

Download the latest Mixly software at:

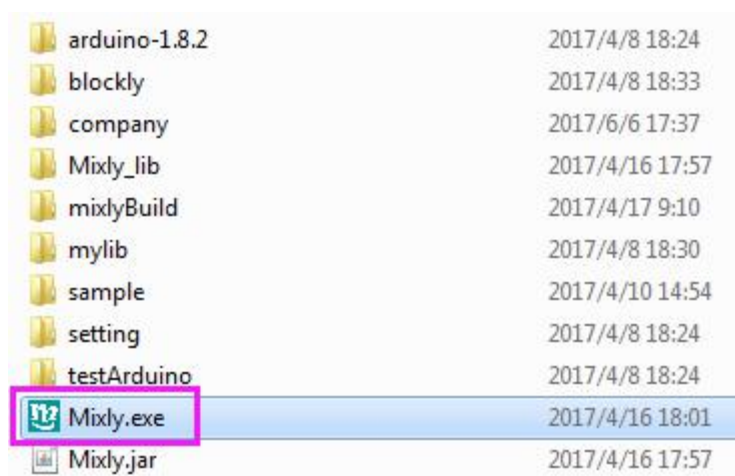
http://wiki.sunfounder.cc/index.php?title=Get_started_with_Mixly

Open the Software

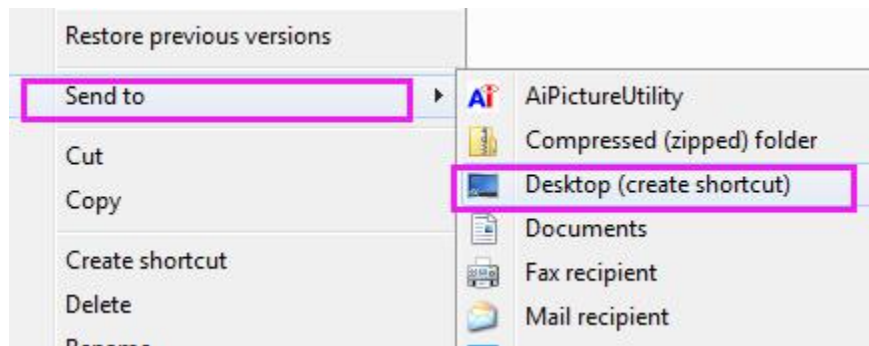
Get into folder Mixly0.995_WIN:



Double click the Mixly icon :



For convenience, we can right-click and select Send to -> Desktop (create shortcut) to create a desktop shortcut.



Brief Introduction

Blocks

The **Mixly0.995_win** version includes In/Out, Control, SerialPort, and some other block categories. Next, let's get through what these categories cover.

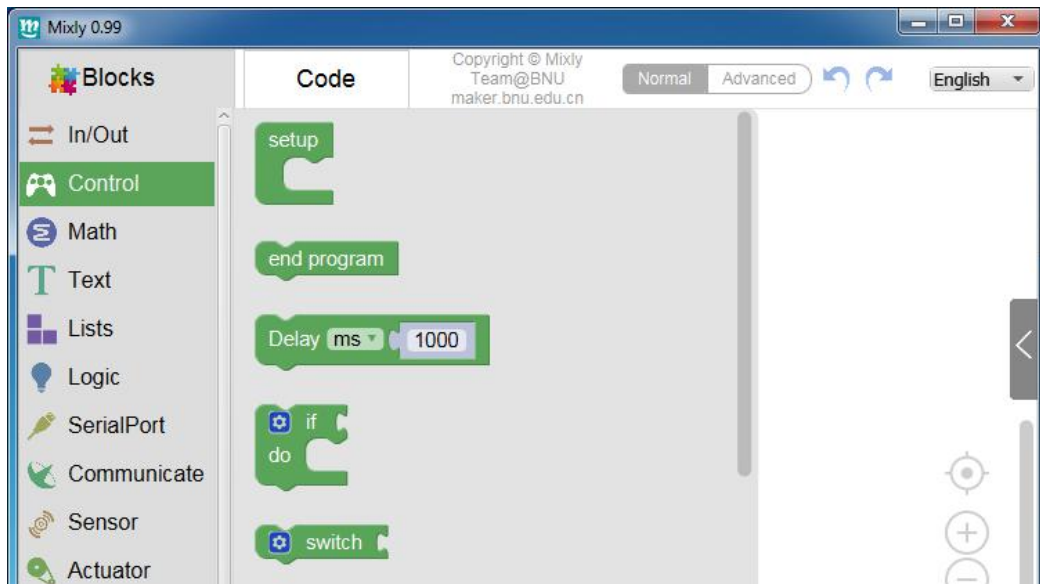
Note: Block categories supports Normal and Advanced views. For more functions, just click **Advanced** to use after you've mastered much.



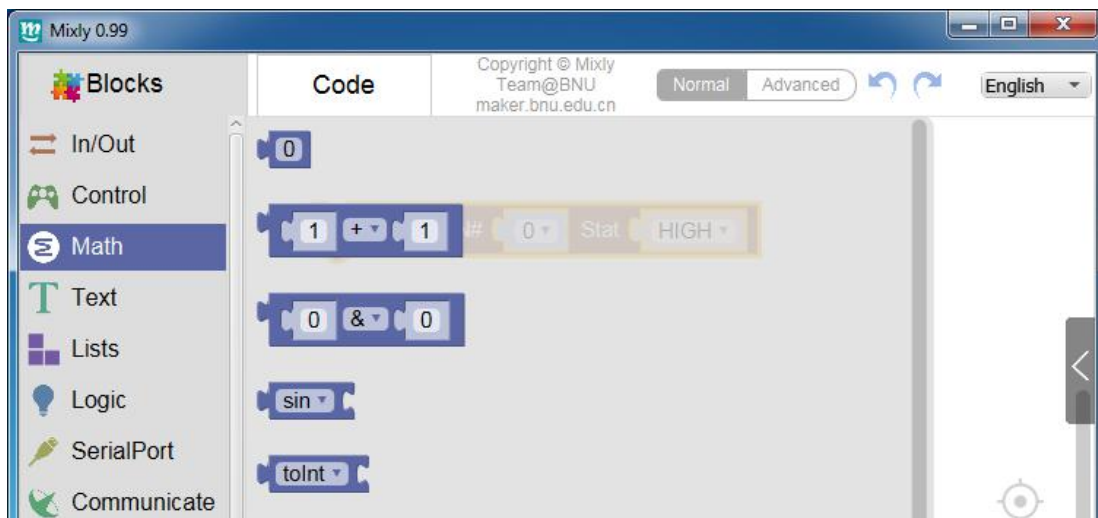
In/Out: DigitalRead, DigitalWrite, AnalogRead, AnalogWrite, attachInterrupt, detachInterrupt, ShiftOut, etc.



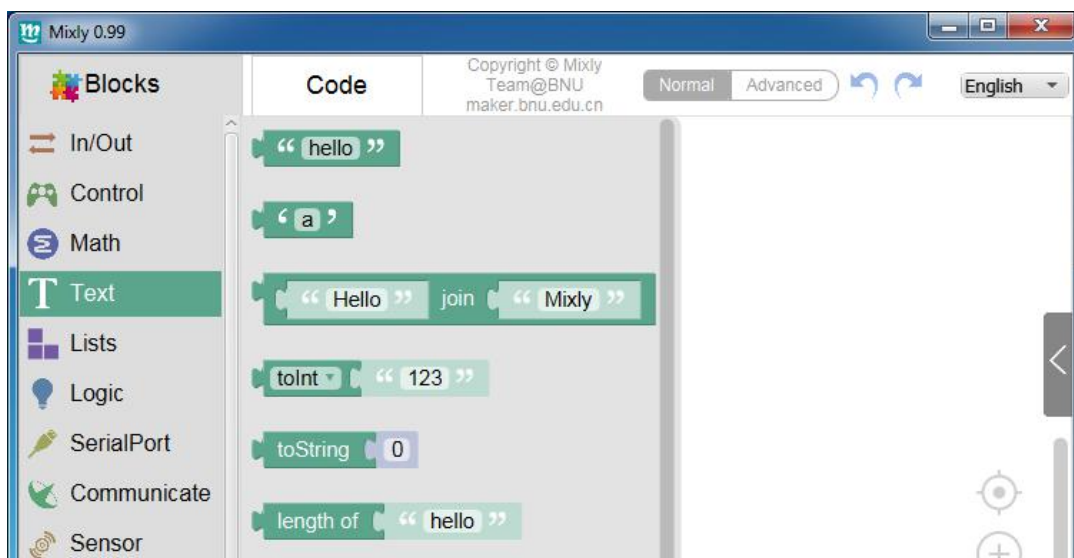
- **Control:** Delay, if...do..., repeat...do..., System running time, setup, etc.



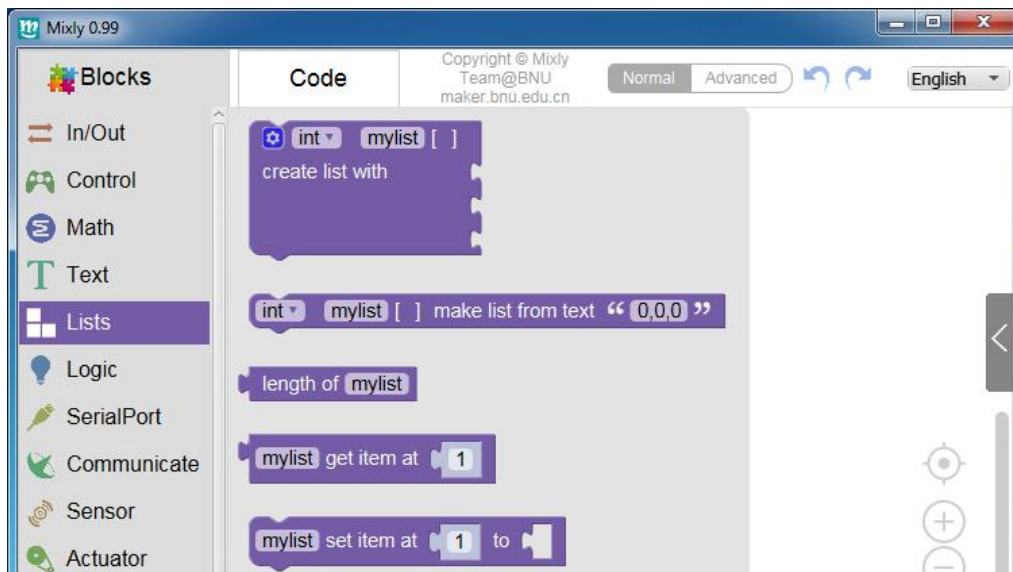
- **Math:** Map, Constrain, operations like +/-, sin/cos, random integer from...to..., etc.



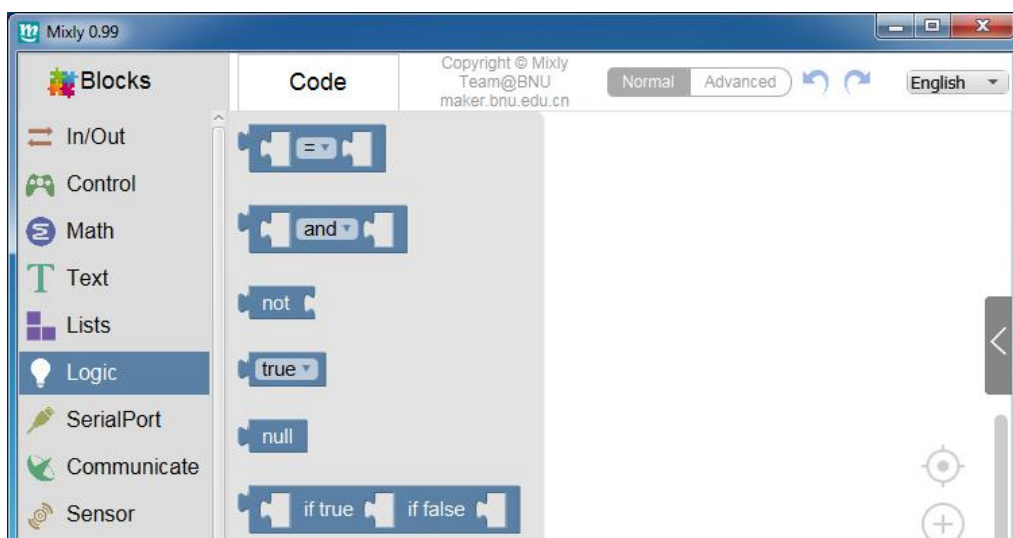
- **Text:** text join, number to string, length of string, compareTo, etc.



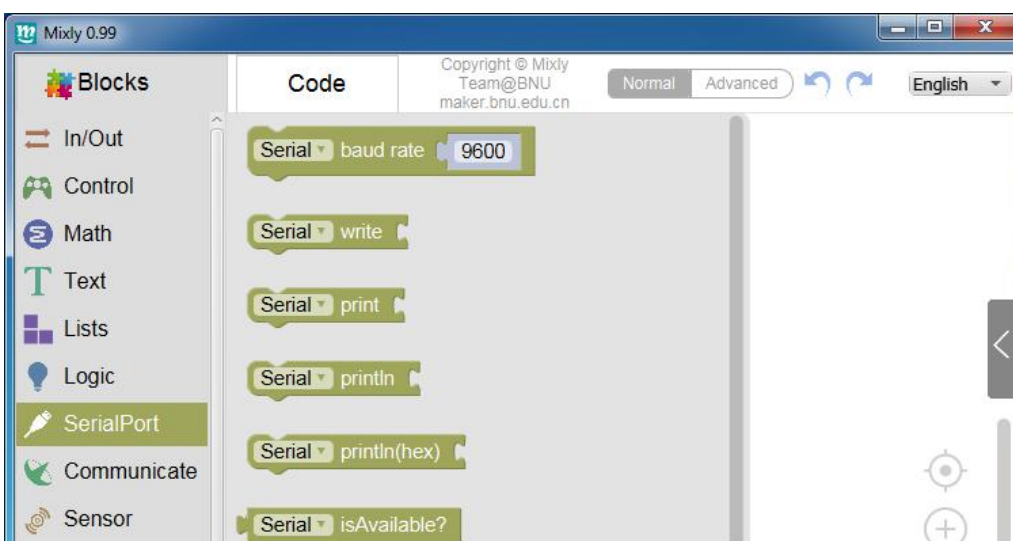
- **Lists:** create list, get item at list, set item at list, etc.



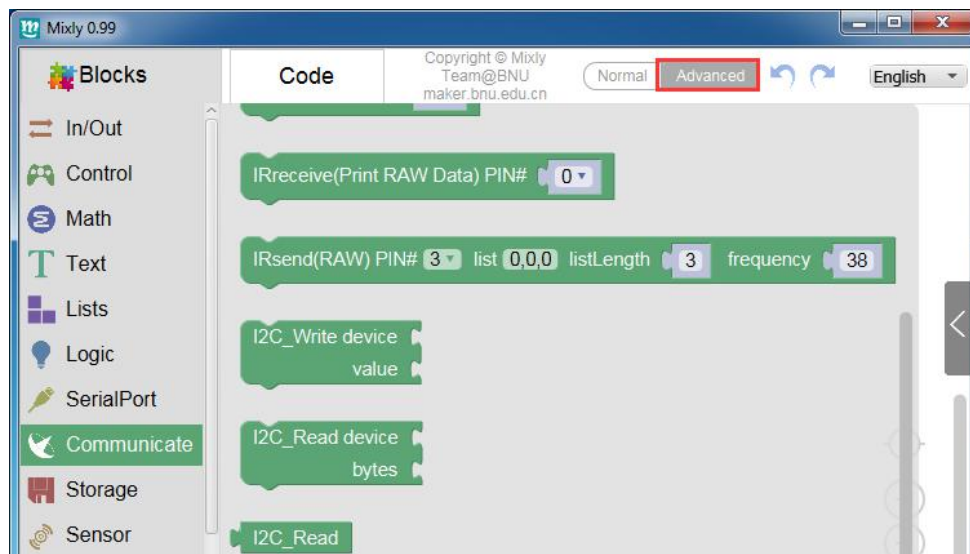
- **Logic:** if conditional statement, logical operations.



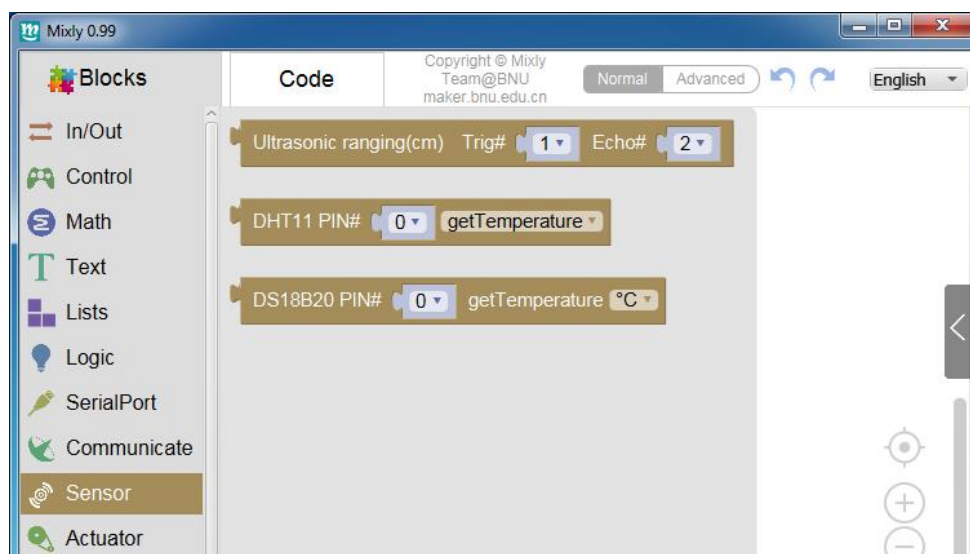
- **SerialPort:** set baud rate, print data, read data.



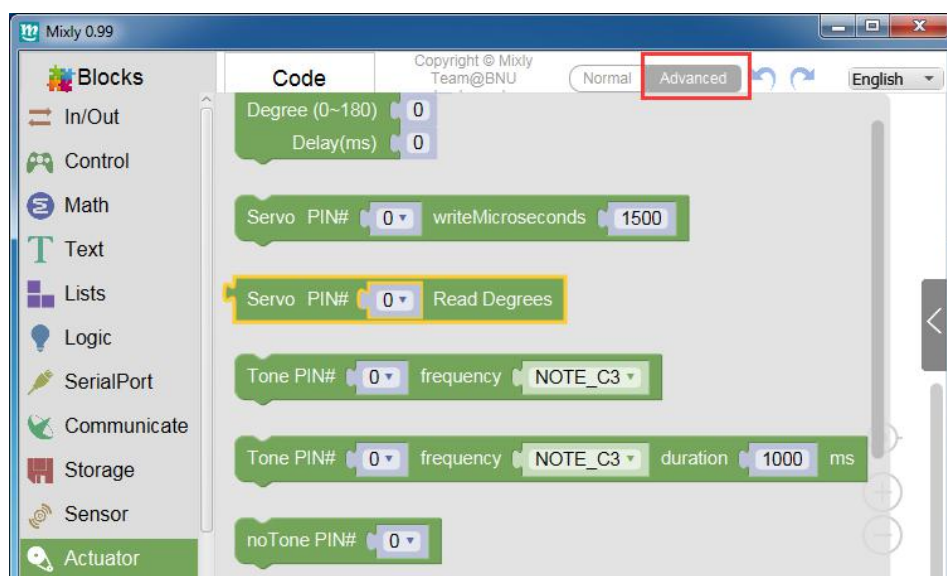
- **Communicate:** IR receive, IR send, I2C read/write device, to SPI, etc.



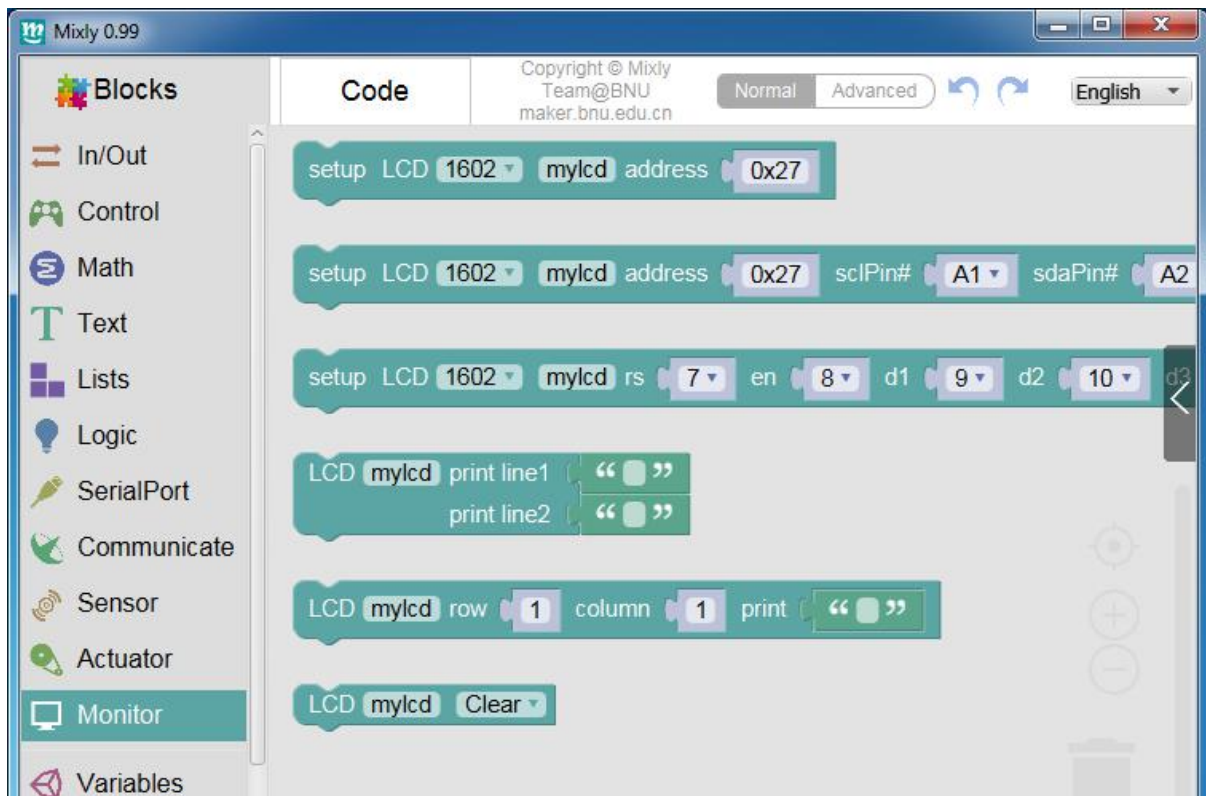
- **Sensor:** ultrasonic, DHT11, DS18B20.



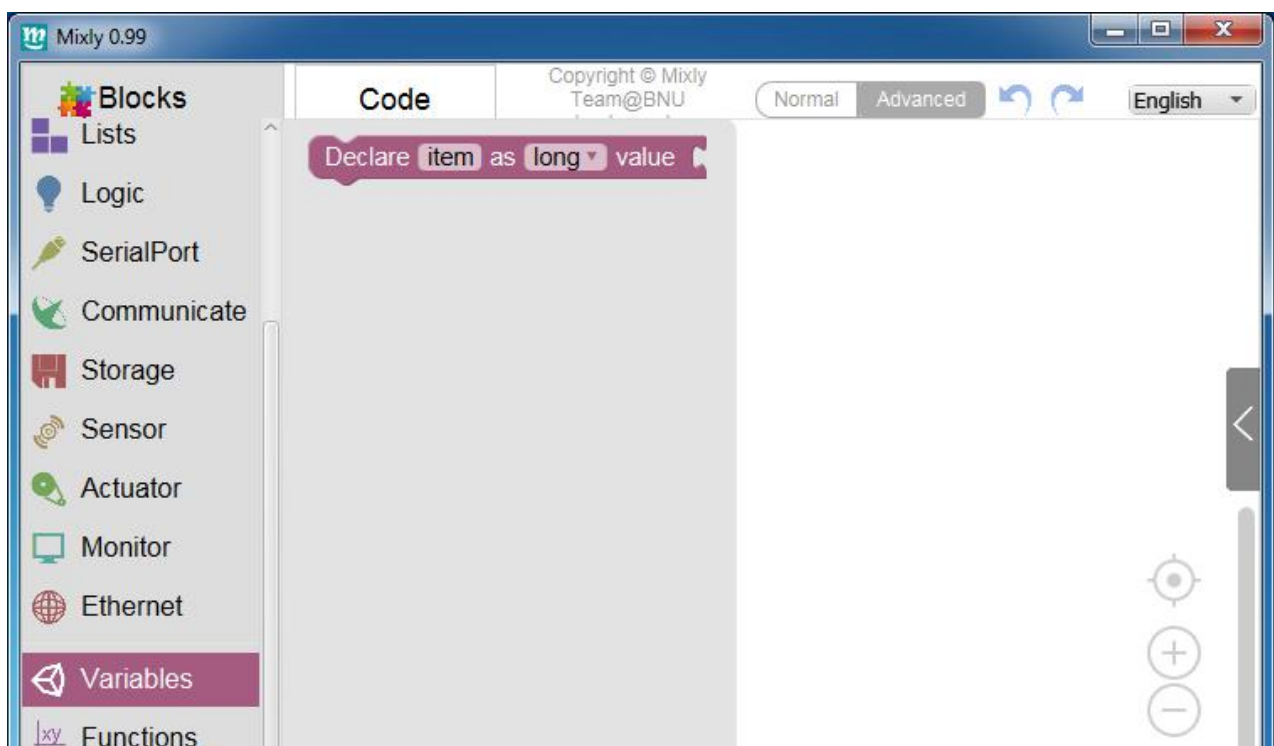
- **Actuator:** Tone control, servo and stepper control.



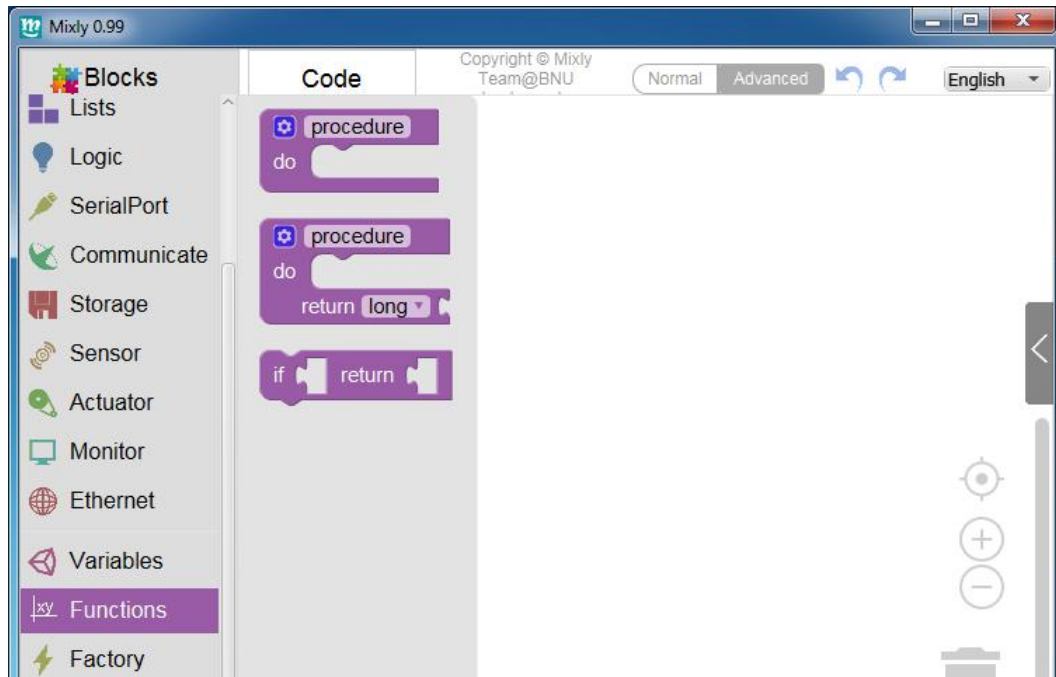
Monitor: setup LCD pin, LCD print data, set LCD I2C address.



Variables: High/Low, True/False, float, integer, Boolean, string variables, etc.



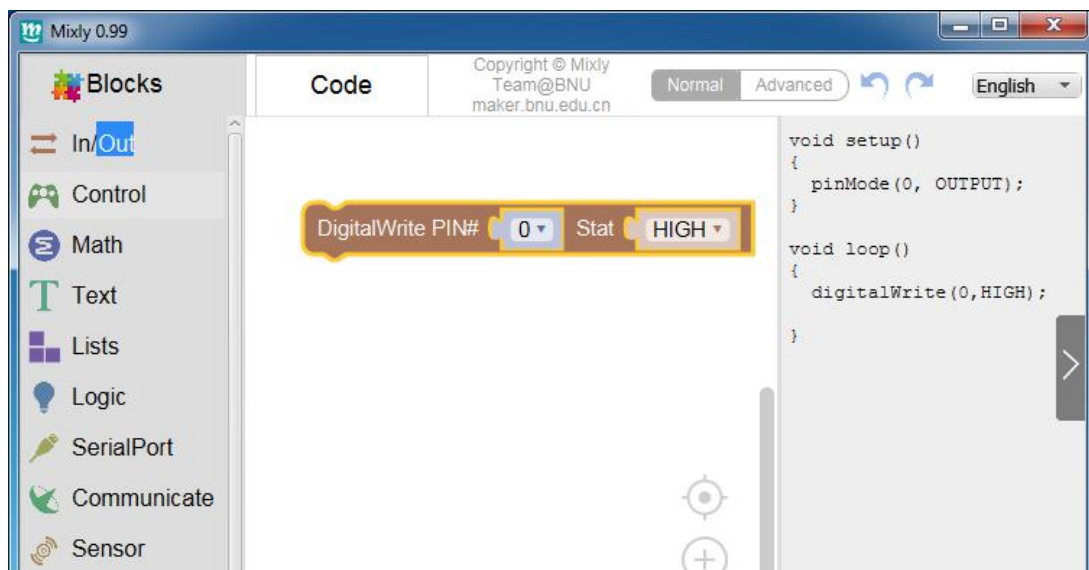
Functions: define function, do procedure with, etc.



Code

You can do graphical programming in this coding area, and the Arduino code will be shown on the right bar for reference.

Note: The code on the right section is just for check and not editable. You can change the blocks on the left coding area to modify the code.



Tools

On the toolbars, you can **Open** or **Save** the code, **Export** or **Import** libraries, **Compile** or **Upload** the code, and select the **Board** and **Port**, and check the **Serial Monitor**.



Select the board and port COMxx: Mixly supports all the official Arduino boards and SunFounder boards (Nano, MARS, and Mercury). Connect the board to the computer, and select the port of the board COMxx as shown in **Device Manager**.



Program Compile and Upload: you can do compile and upload on Mixly, and it will return information, thus you can know whether the code is compiled or uploaded successfully.



Create, Open and Save code: you can create, open and save code.



Import, Export Libraries and Manager: literally, import or export the related libraries.

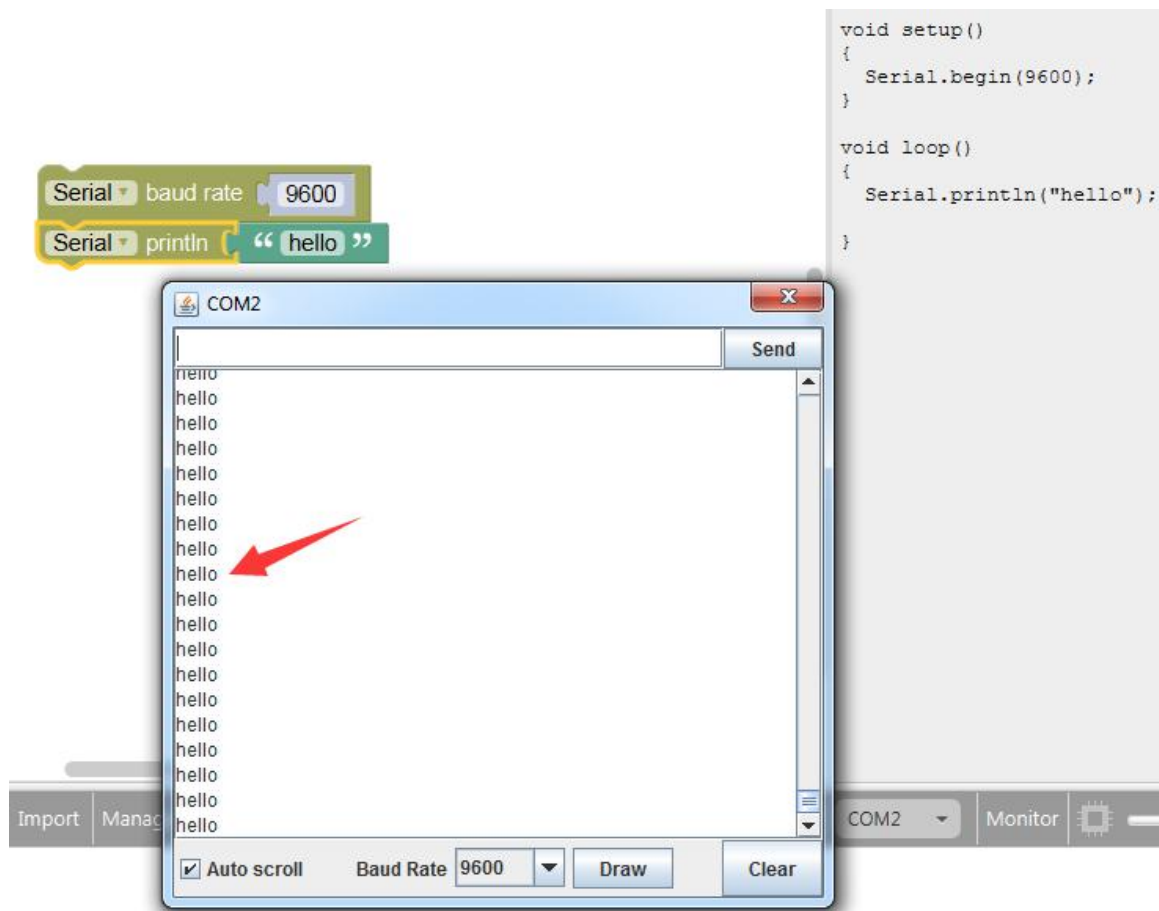


Import is to import a library, **Export** is to export a library, and **Manager** is to rename or delete the imported library which may be used in later part.

Serial Monitor: to print some results or characters you set.



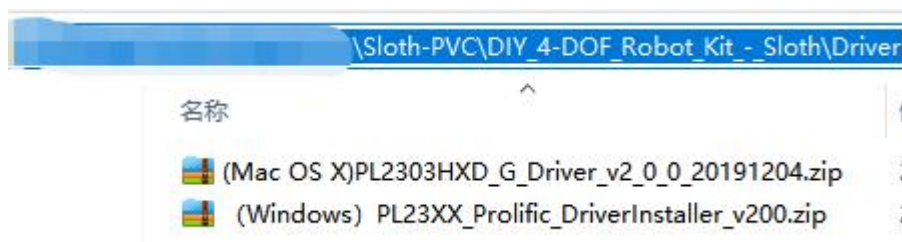
For instance, set print "hello" in the code.



Install the Driver

The USB driver chip of SunFounder Nano board is **PL2303TA**. After installing the Arduino IDE, use Mini USB Cable to connect the SunFounder Nano to your computer. Some Windows computers may install the driver automatically after a while, and the corresponding COMxx pops up. Congratulations, you can use it directly without having to install the driver manually.

For some users who cannot install the Driver automatically, we provide the Driver files for Windows and Mac OS X systems in the path **DIY_4-DOF_Robot_Kit_-_Sloth\Driver**.



Unzip and open the PDF inside and install the Driver according to the **Windows/Mac OS X Driver Installation** section.

Since the PL2303 driver installer is always being updated, the driver provided by us may not be compatible with your computer system. Please go to the link below to download the latest driver.

For Windows users:

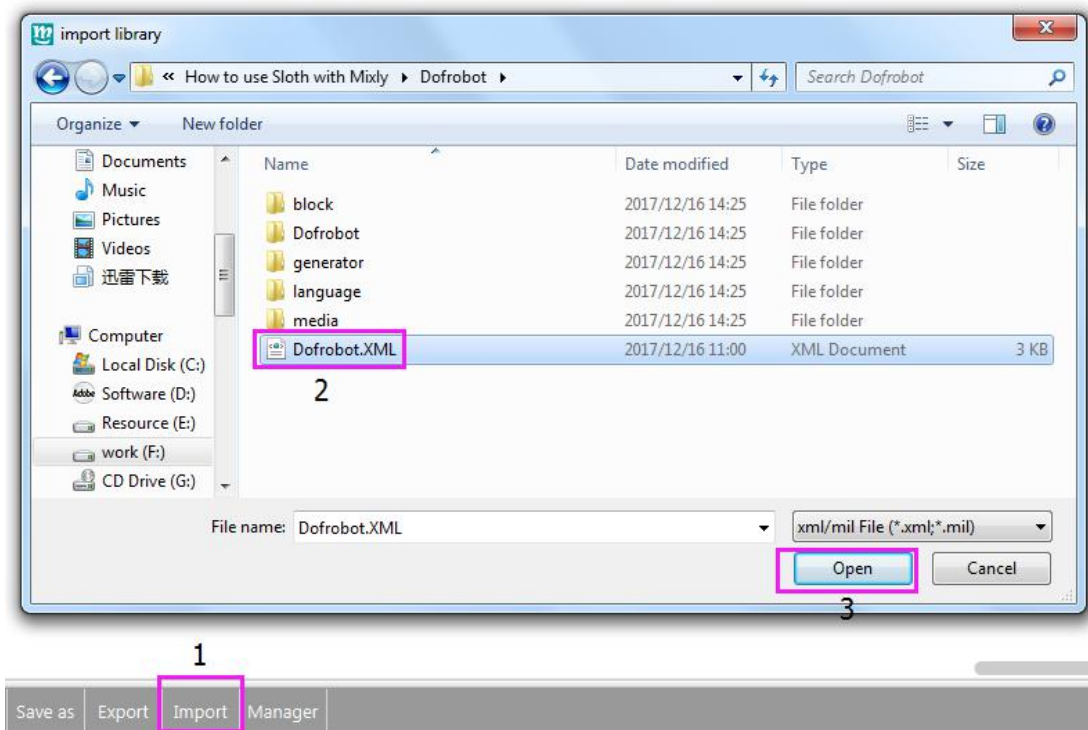
http://www.prolific.com.tw/US/ShowProduct.aspx?p_id=225&pcid=41.

For Mac users:

http://www.prolific.com.tw/US/ShowProduct.aspx?p_id=229&pcid=41

Add Library

Unzip **Dofrobot** package, and click **Import**, to select **Dofrobot.XML** under *How to use Sloth with Mixly\Dofrobot*, and click **open**.




Then you will see the Dofrobot library in the left. If you want to import more libraries without affecting the uploading process, just delete the imported library. Here is how to do: click **Manage**, and select the library you want to remove, and click **delete**. If you want to use this library, just redo Import operations.

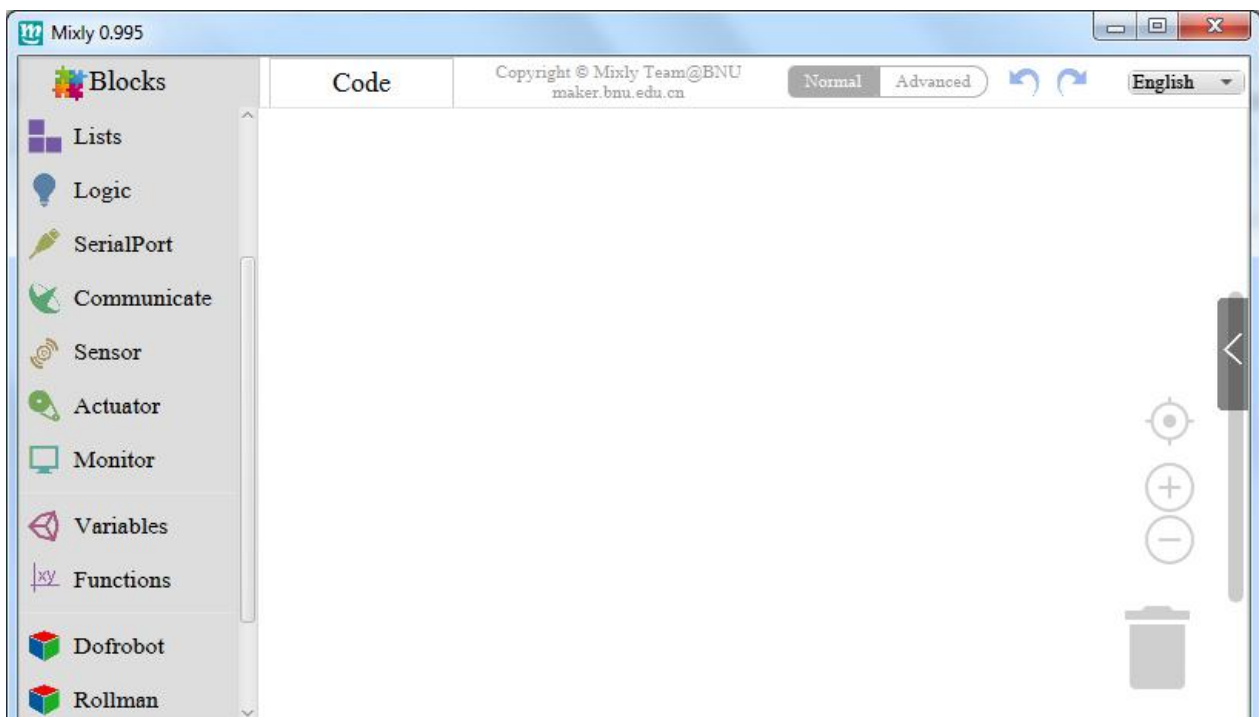
Lesson 1 Moving Forward

Overview

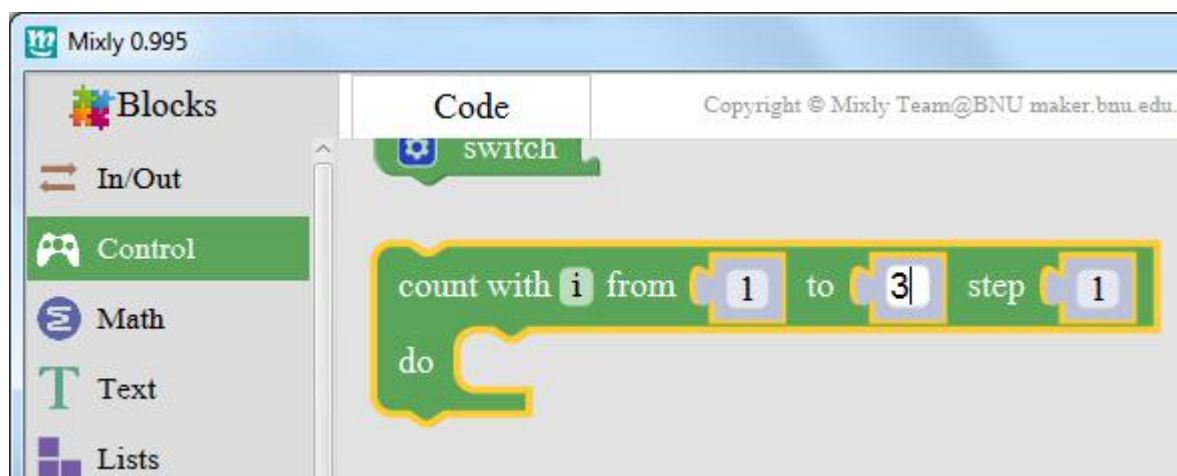
This lesson is how to command the 4-DOF Robot to moves forward and then stop.

Programming

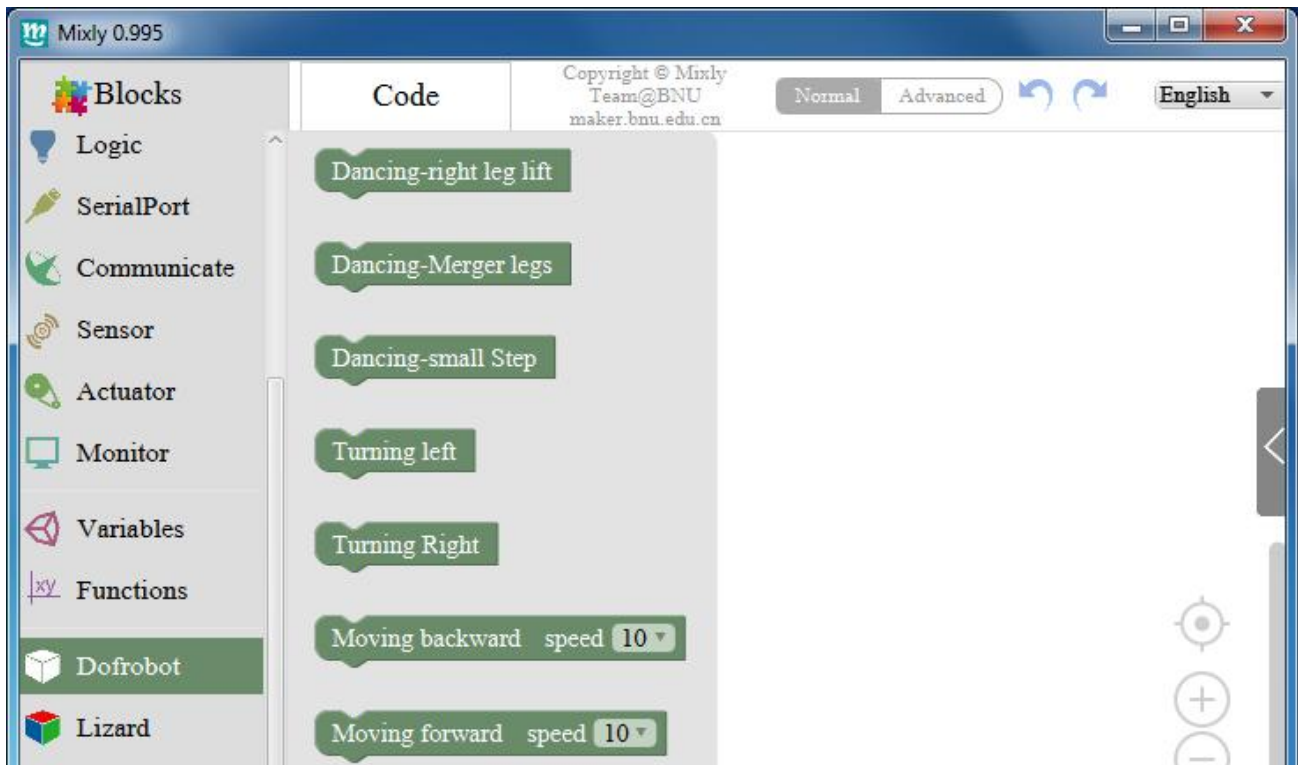
Step 1: Click the icon  on the desktop to open the **Mixly** and you'll be on the homepage:



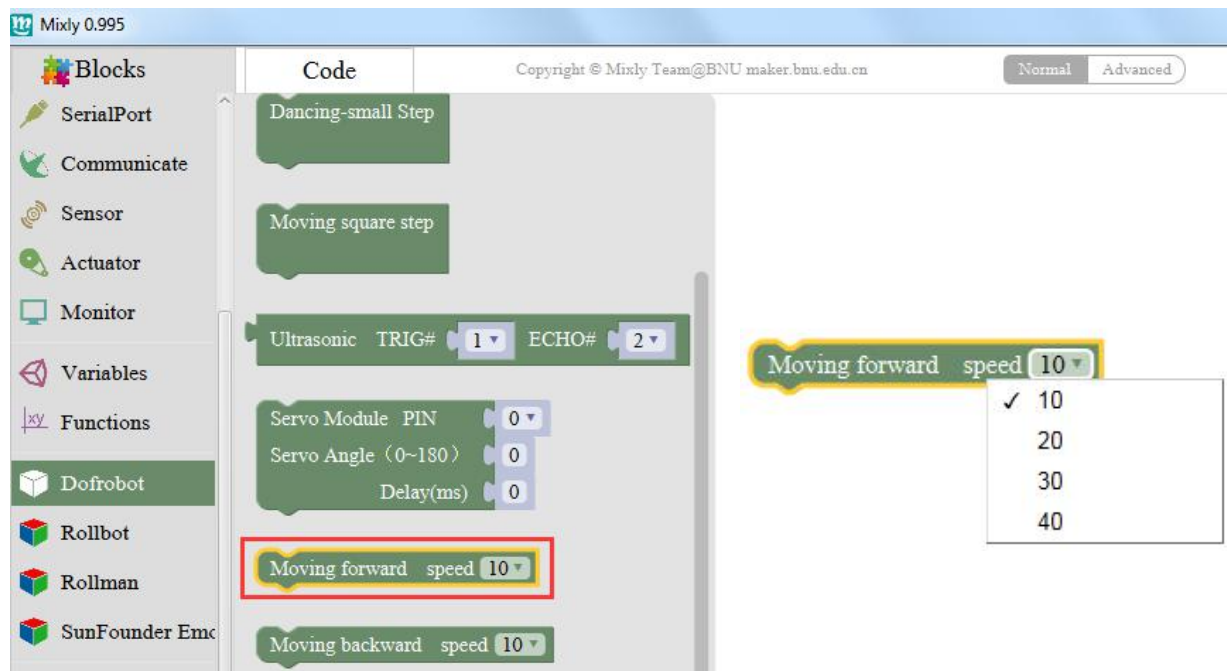
Step 2: Click **Control category**, drag out the **count with block** into the coding area. This is an accumulation block, means that $\text{count} = \text{count} + 1$. When $\text{count} = 3$, then stop.



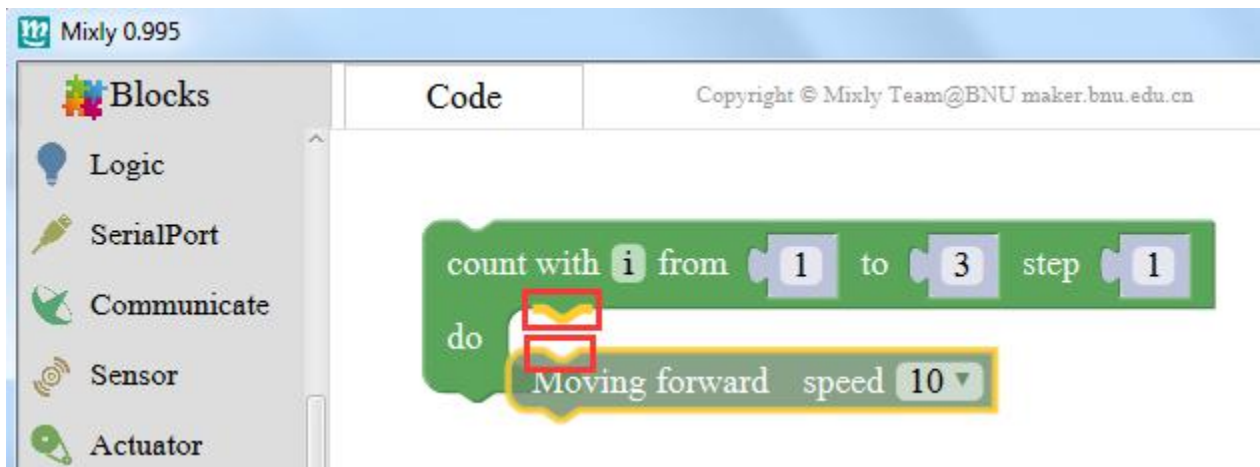
Step 3: Click **Dofrobot** category, then we can see several blocks.



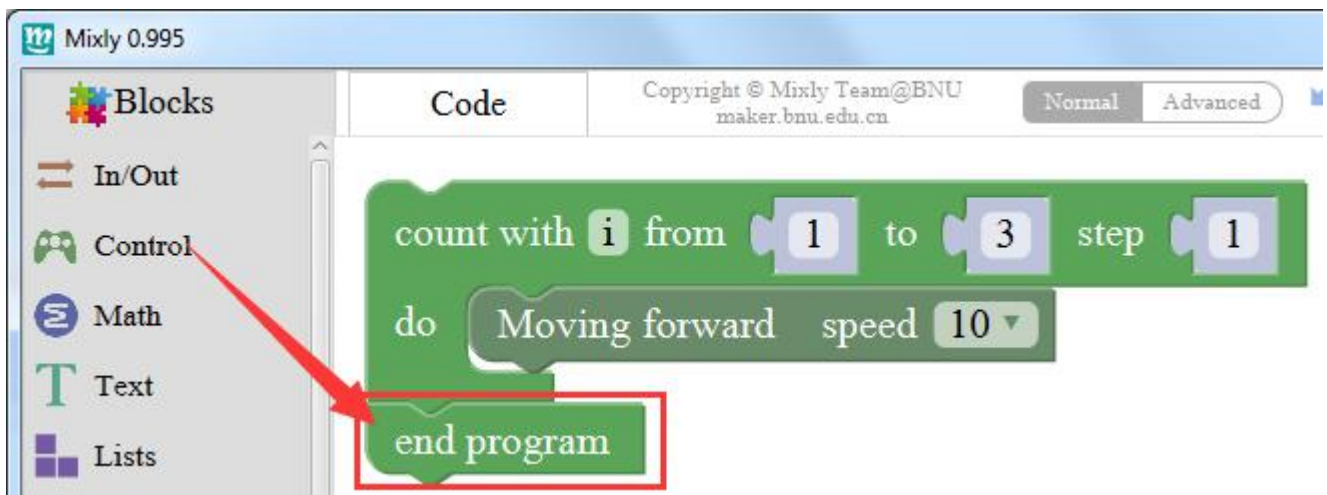
Step 4: Click and drag out **Moving forward block**. This is a block that can controls the robot to go forward. You can set the forward speed as 10, 20, 30, 40.



Step 5: Drag the **Moving forward block** to the **count with block** and fit the bulge and slot of the two to combine them.

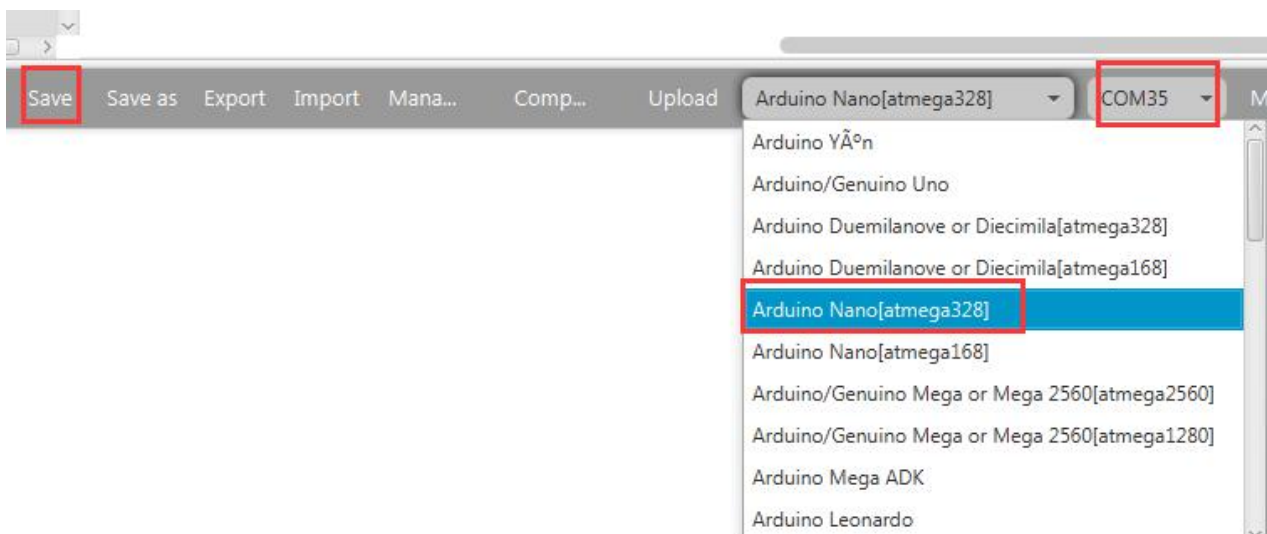


Step 6: You can end the program by the **end program block** from the **Control category**. If you don't use the **end program**, this process will work constantly.

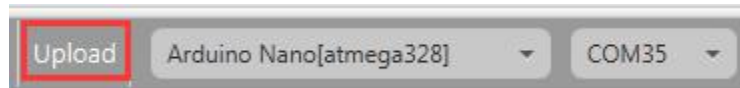


Step 7: Click **Save** after all the programming is done. Select the **board type** and **port** before uploading. For instance, if you use a **Uno** or **Mars** board, just select **Arduino/Genuino Uno**; if you use a **Mega 2560** or **Mercury**, select **Arduino/Genuino Mega** or **Mega2560**.

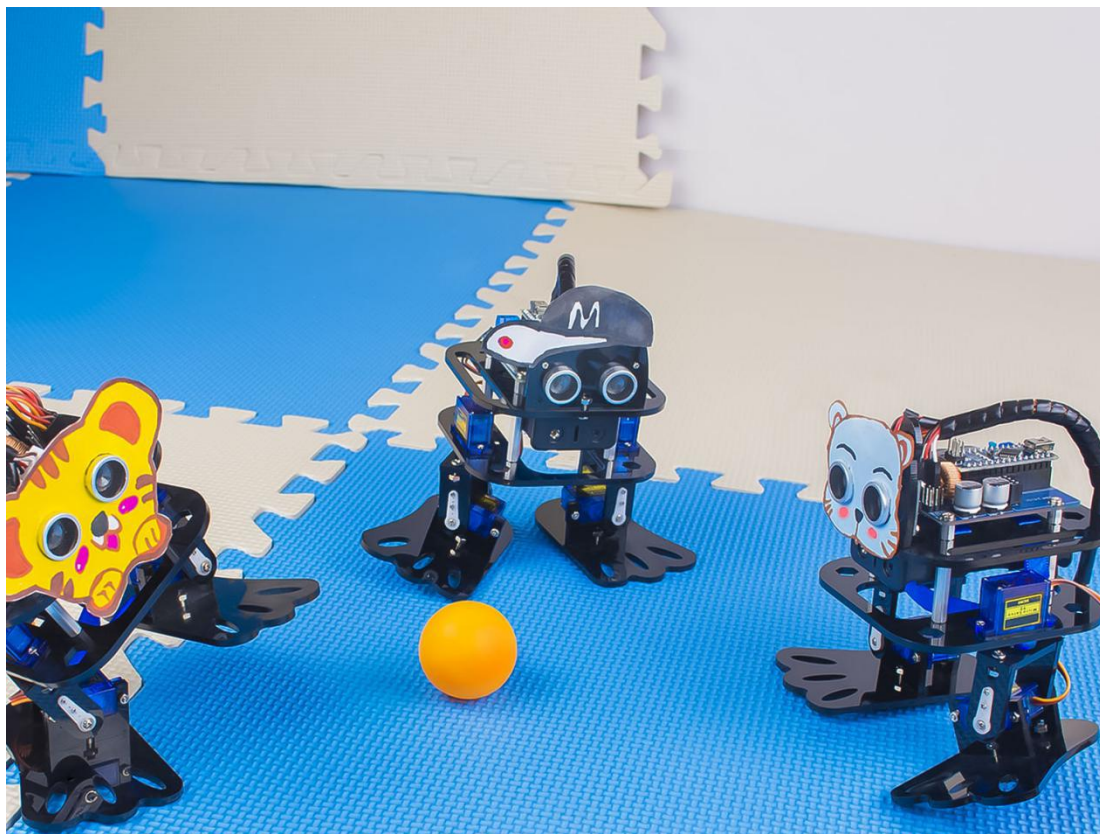
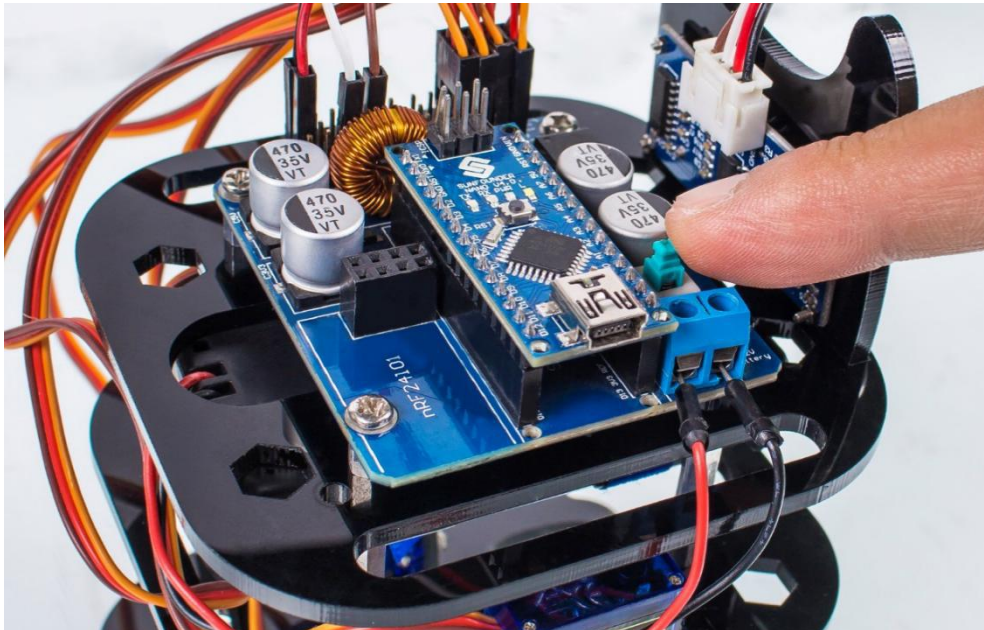
Then select the **port**. You can check the port in **Device Manager** when you connect the board to the computer.



Step 8: Upload the code. If the uploading fails, check and correct the code according to the prompt.



After burning successfully, unplug the USB cable and press the power button on the servo control board. You will see the Sloth moves forward, when two feet all go forward for three steps, it will then stop.




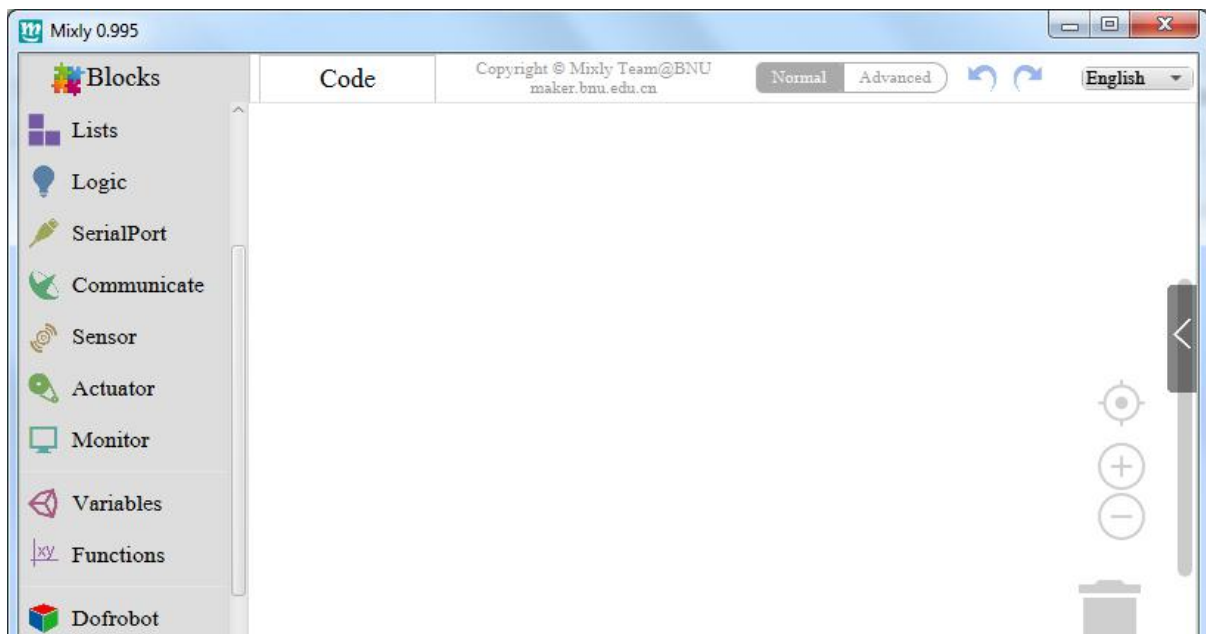
Lesson 2 Moving backward

Overview

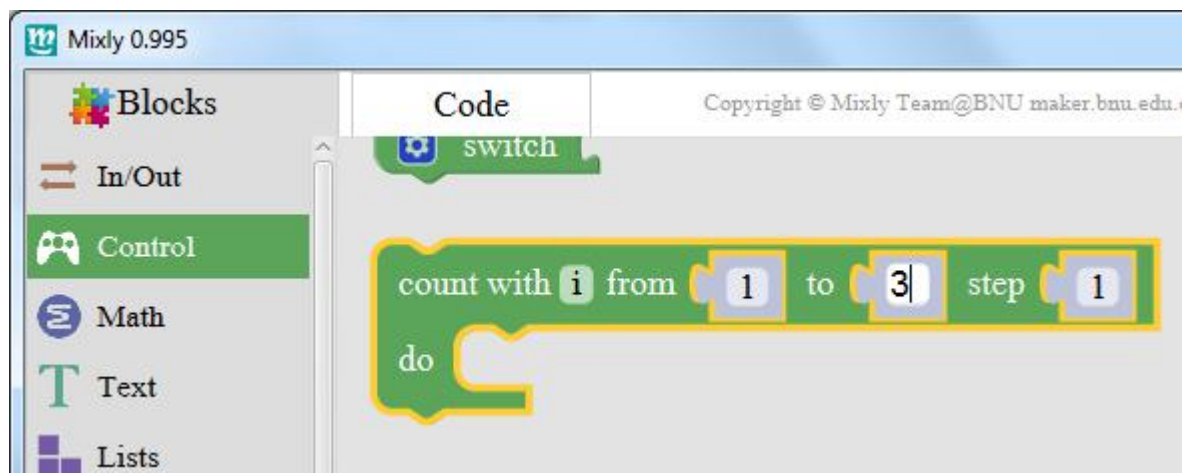
This lesson is to command the 4-DOF Robot to moves backward and then stop.

Programming

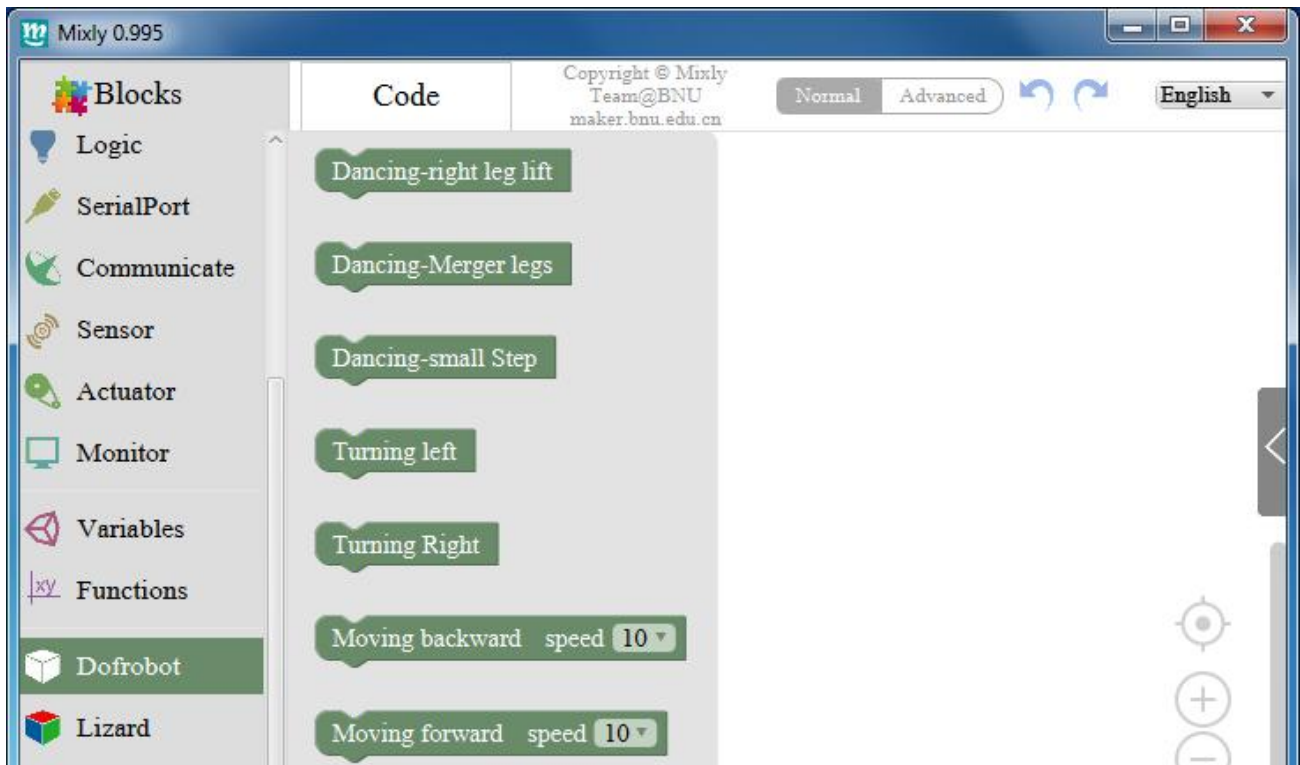
Step 1: Click the icon  on the desktop, you'll be on the homepage as shown below:



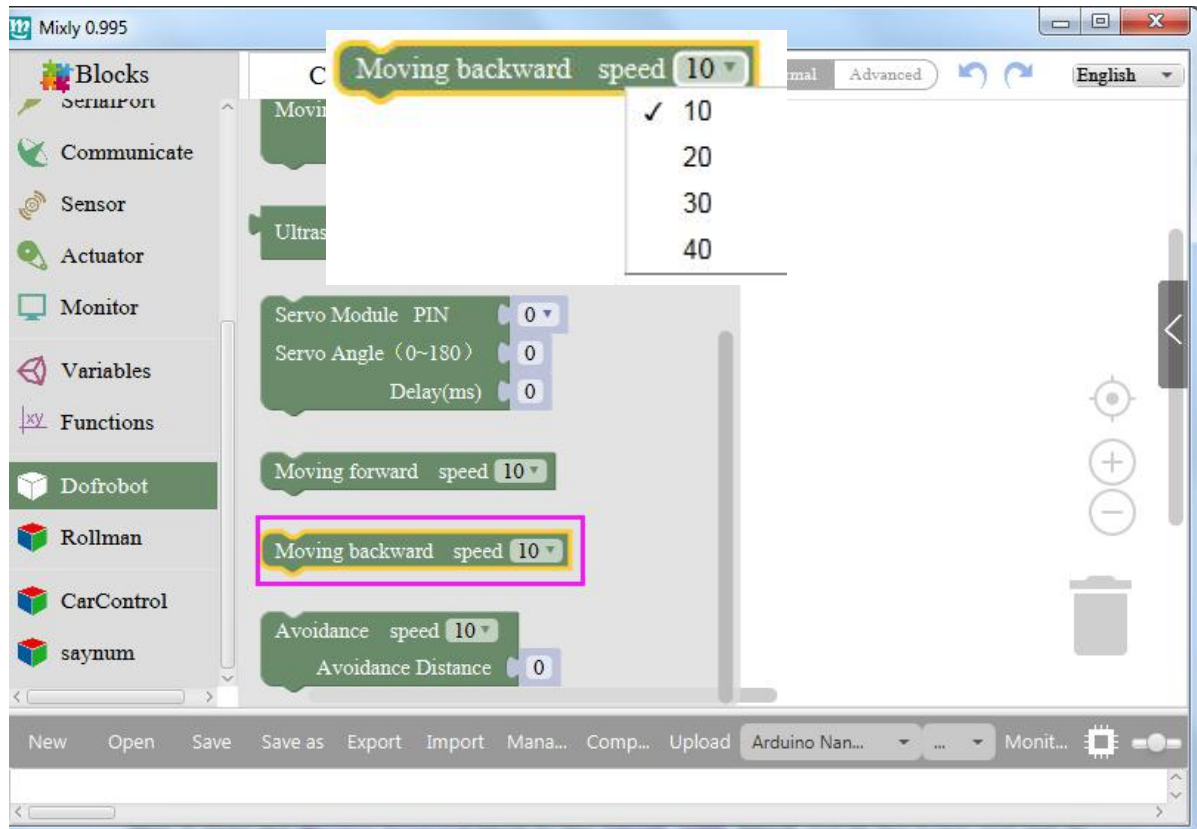
Step 2: Click the **Control category**, drag out the **count with block** into the coding area. This is an accumulation block, means that $\text{count} = \text{count} + 1$. When $\text{count} = 3$, the robot will stop.



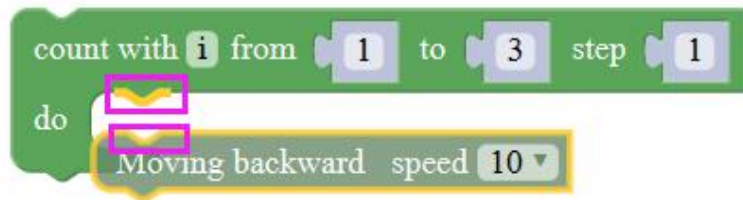
Step 3: Click **Dofrobot** category, then we can see several blocks.



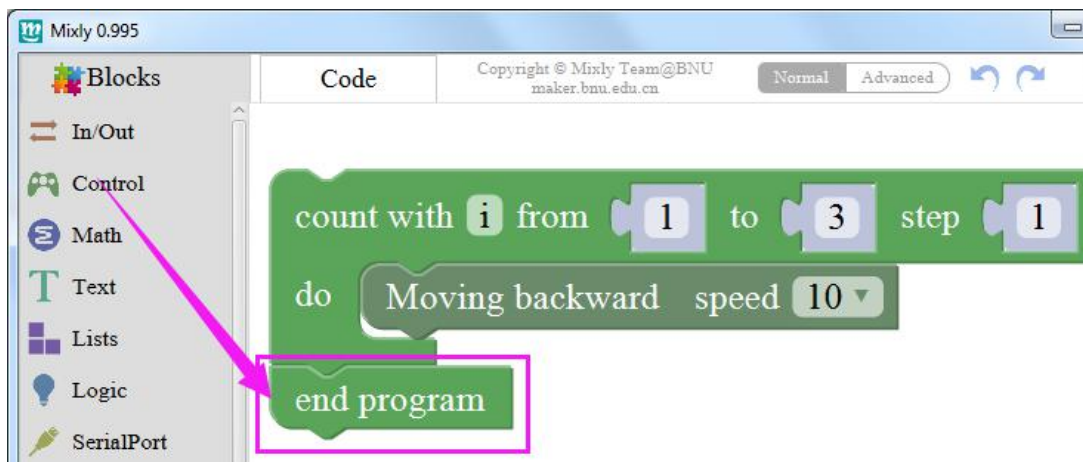
Step 4: Click and drag out **Moving backward block**. This block commands the robot to moves backward. You can set the speed as 10, 20, 30 and 40.



Step 5: Drag the **Moving forward block** to the **count with block** and fit the bulge and slot at the two to combine them.

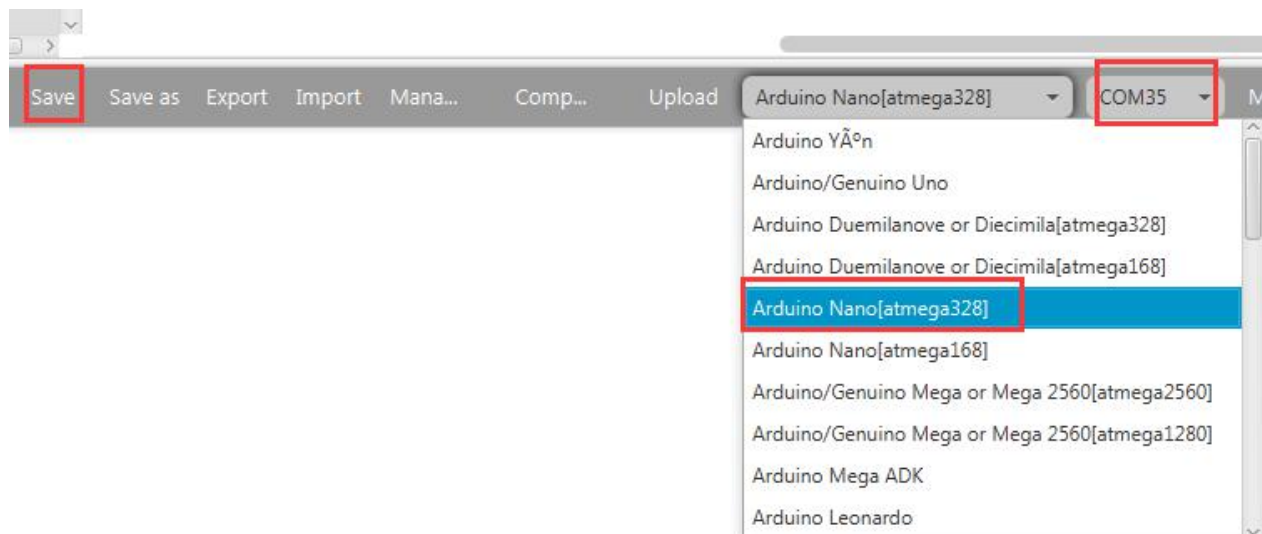


Step 6: You can end the program by the **end program block** from the **Control category**. If you don't use the **end program**, this process will work constantly.

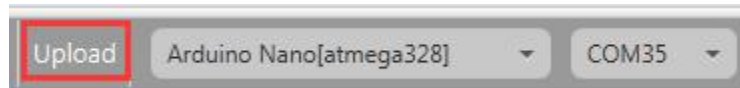


Step 7: Click **Save** after all the programming is done. Select the **board type** and **port** before uploading. For instance, if you use a **Uno** or **Mars** board, just select **Arduino/Genuino Uno**; if you use a **Mega 2560** or **Mercury**, select **Arduino/Genuino Mega** or **Mega2560**.

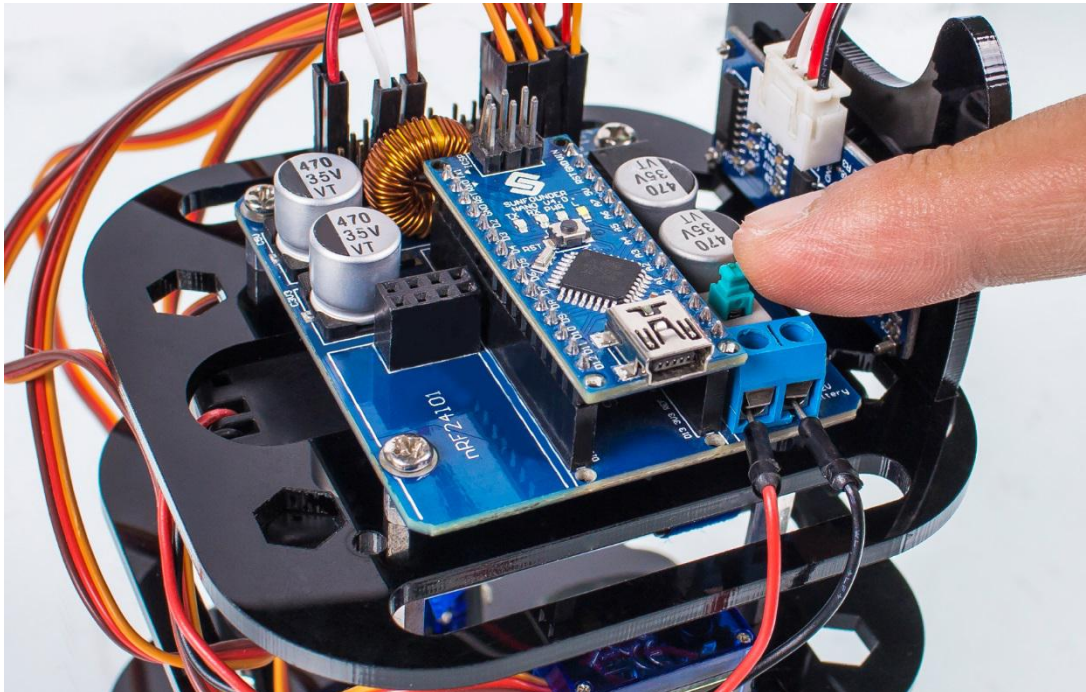
Then select the **port**. You can check the port in **Device Manager** when you connect the board to computer.



Step 8: Upload the code. If the uploading fails, check and correct the code according to the prompt.



After burning successfully, unplug the USB cable and press the power button on the servo control board. You will see the Sloth moves backward for three steps and then stop.



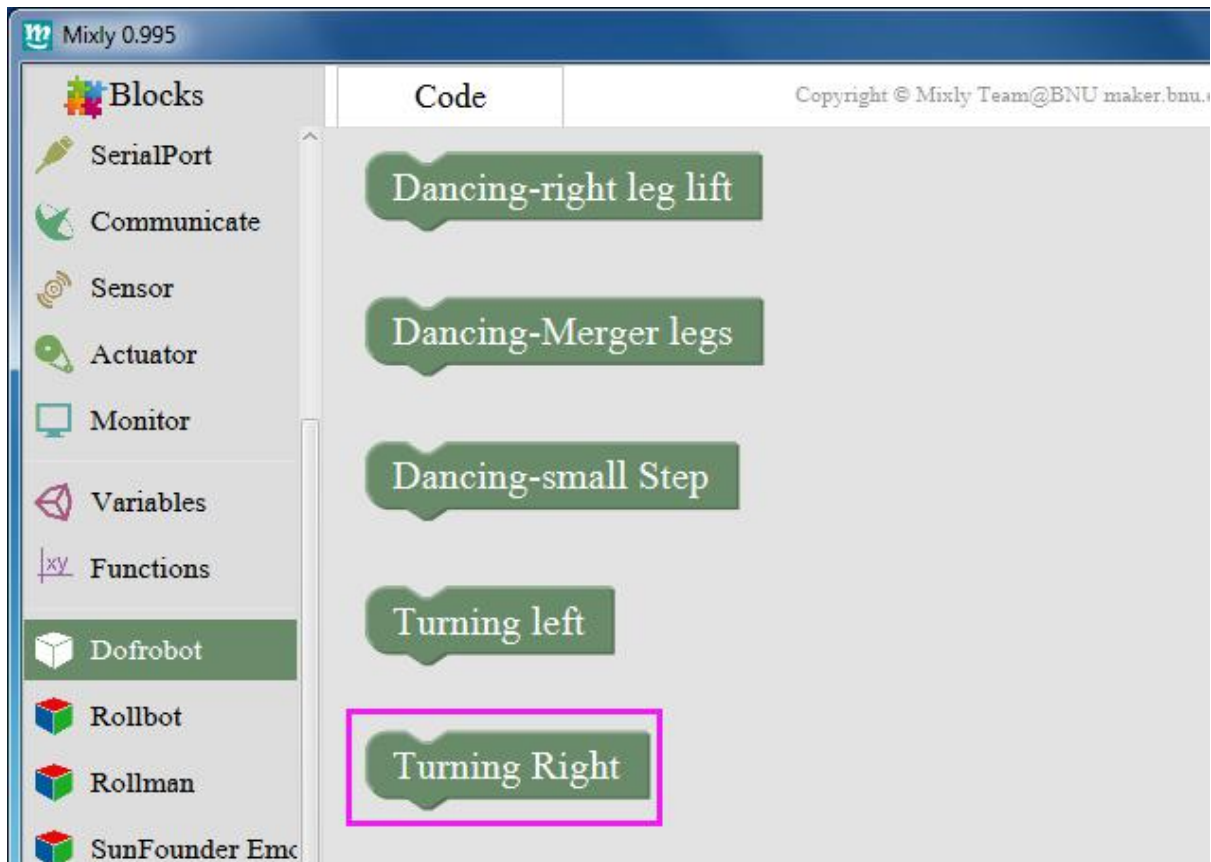
Lesson 3 Moving Forward in S-shape

Overview

This lesson is how to command the Sloth to moves forward in S-shape.

Programming

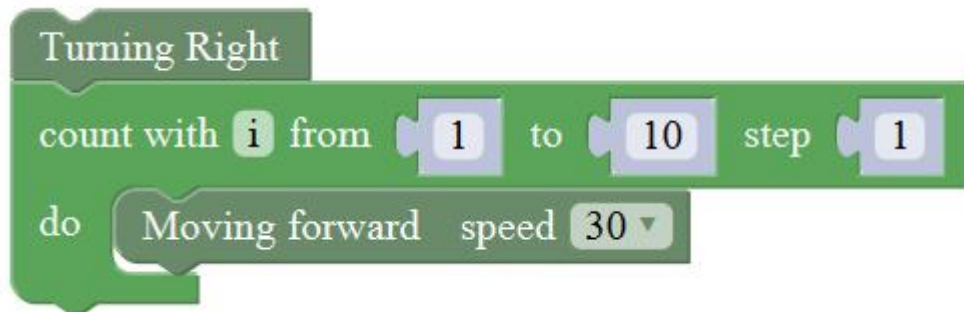
Step 1: Open the **Mixly**, and then drag out the **Turning right block** from **Dofrobot category**.



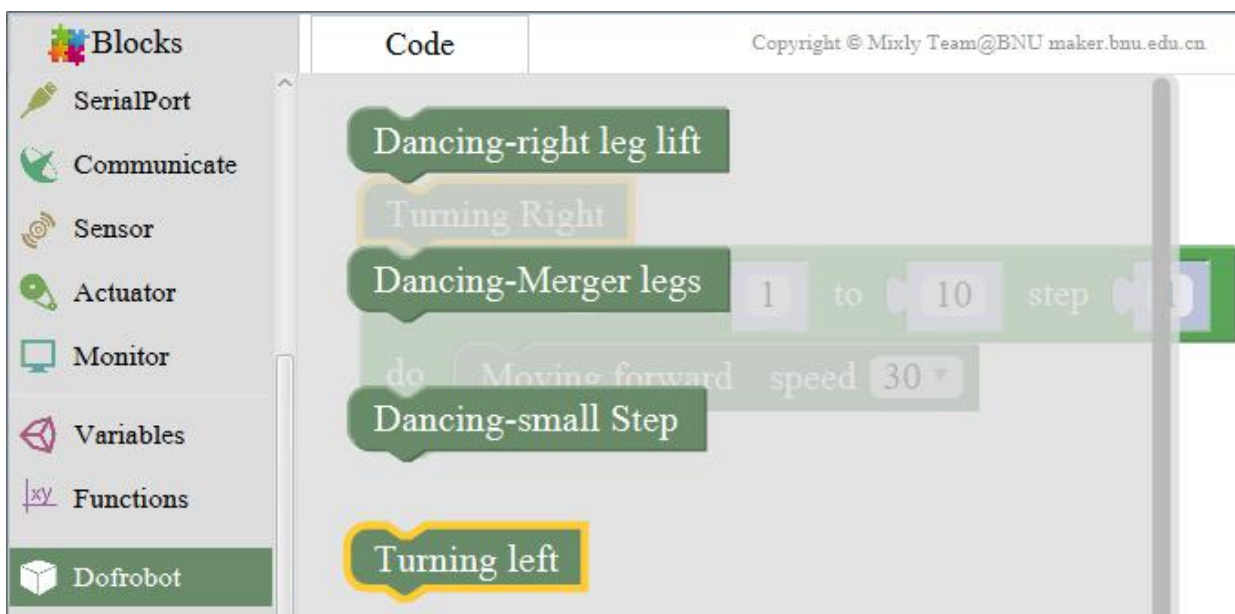
Step 2: Drag out the **count with block** from **Control category**.



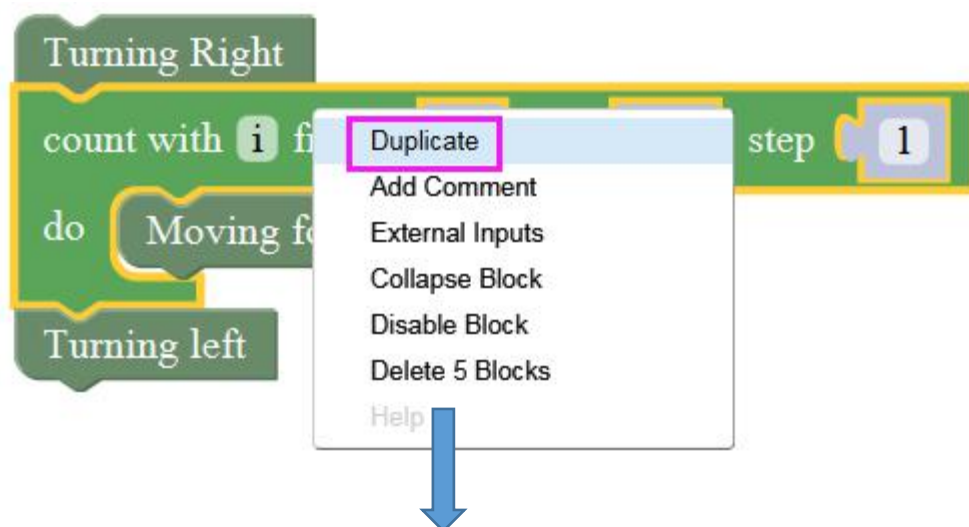
Step 3: Drag **Moving forward block** from **Dofrobot category**, thus the Sloth can turn right and move forward for 3 steps. The speed of moving forward and step can be set by yourself.

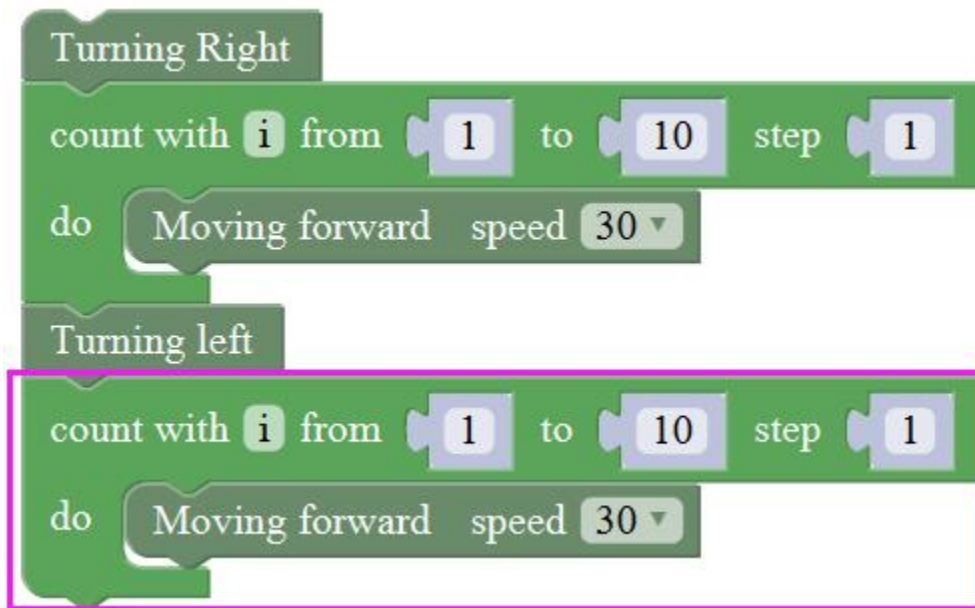


Step 4: Drag the **Turning left block** from **Dofrobot category**.

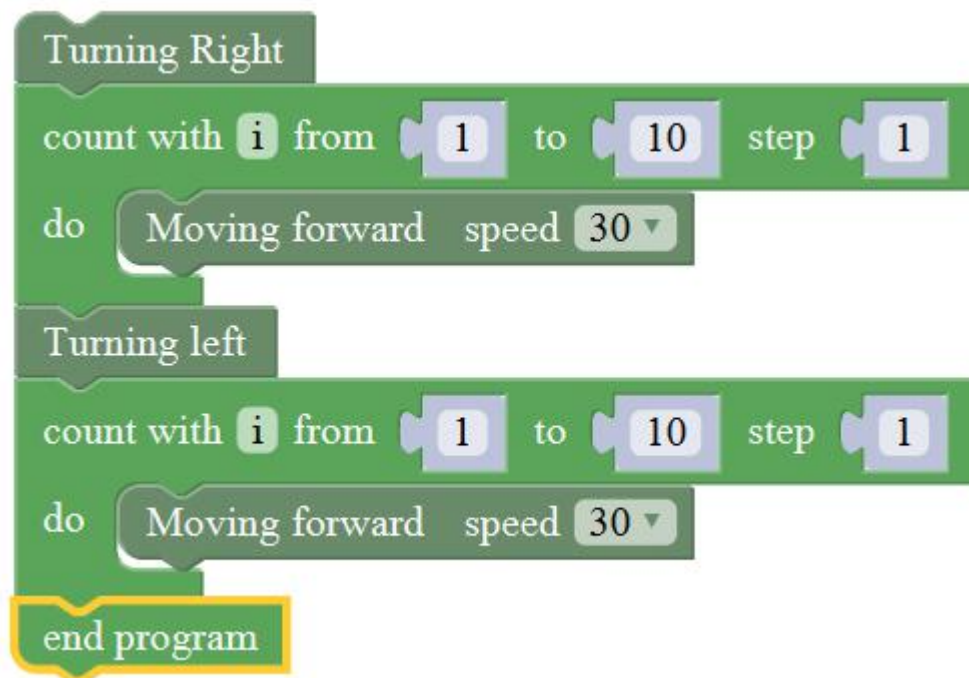


Step 5: Right click the **count with block** to choose the **Duplicate**, then combine with **Turning left block**.



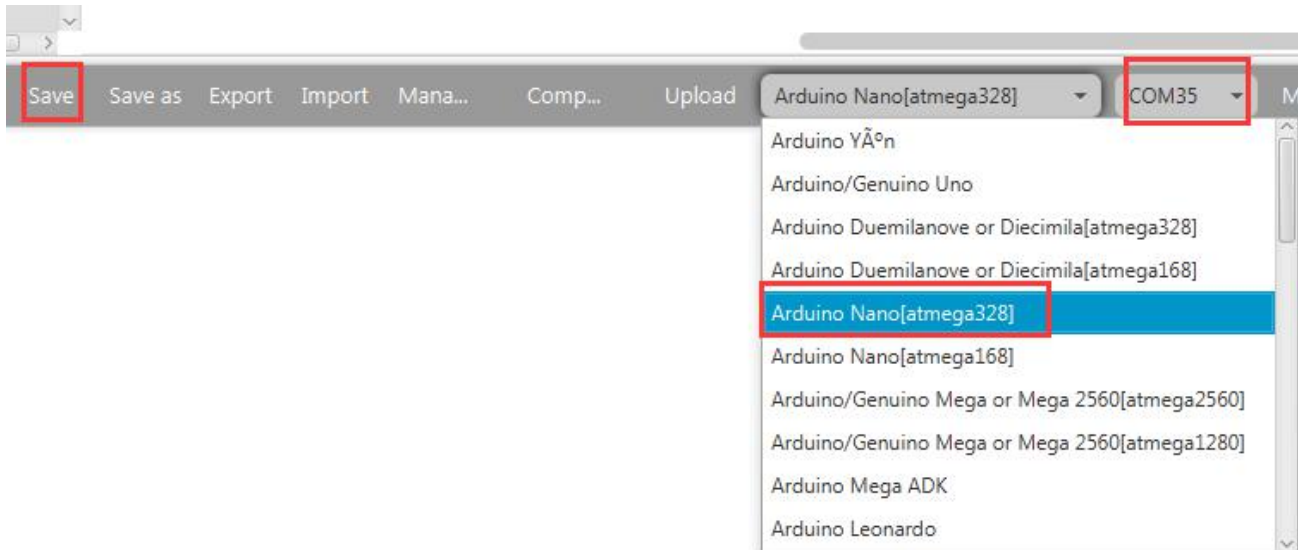


Step 6: Finally end the process by the **end program**.

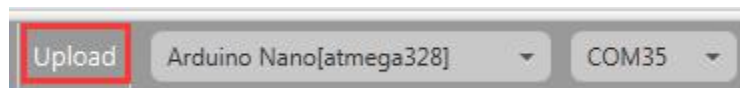


Step 7: Click **Save** after all the programming is done. Select the **board** type and **port** before uploading. For instance, if you use a **Uno** or **Mars** board, just select **Arduino/Genuino Uno**; if you use a **Mega 2560** or **Mercury**, select **Arduino/Genuino Mega** or **Mega2560**.

Then select the **port**. You can check the port in **Device Manager** when you connect the board to computer.



Step 8: Upload the code. If the uploading fails, check and correct the code according to the prompt.



After burning successfully, unplug the USB cable and press the power button of the servo control board. You'll see the Sloth turn left and move forward for 3 steps, then turn right and move forward for 3 steps.

Note: Different surfaces have different friction rate, thus the Sloth turning angle may not too big in some kind of abrasive surface.

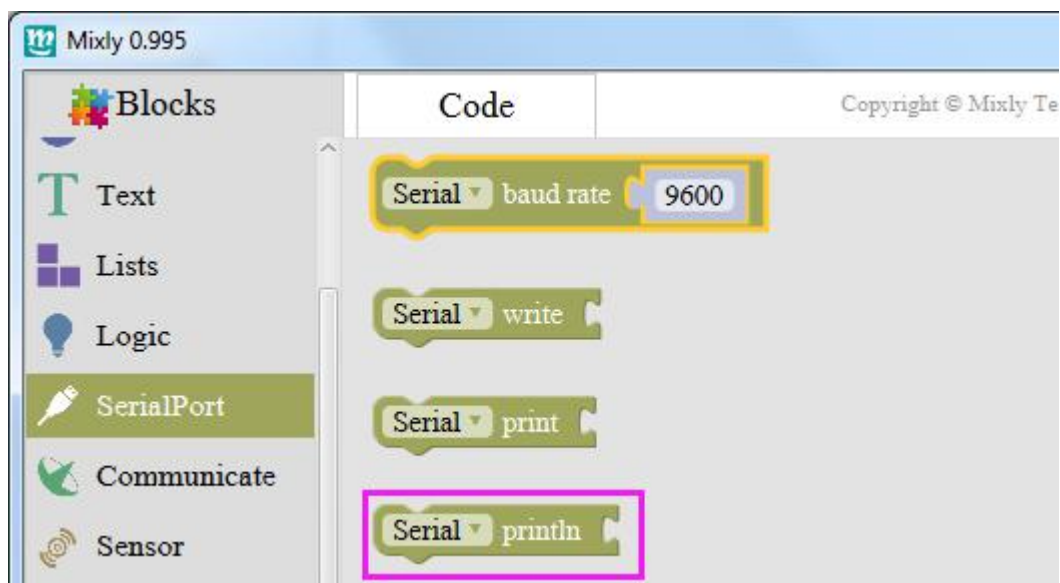
Lesson 4 Test Ultrasonic Module

Overview

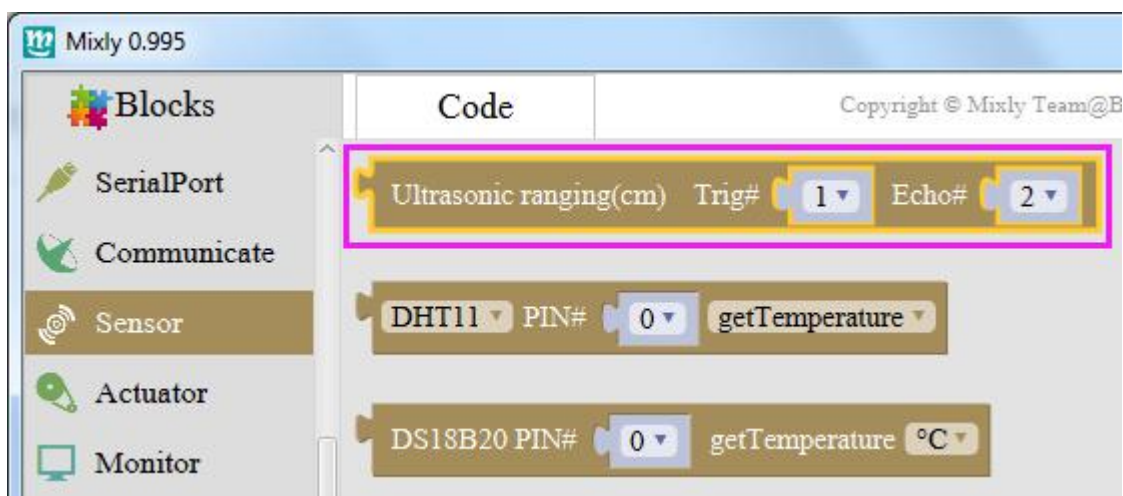
Sloth also has an ultrasonic module as its eyes to avoid obstacles. Now let's test the ultrasonic module.

Programming

Step 1: Click the **SerialPort** category, drag the **Serial.println** to the coding area.



Step 2: Click **Sensor**, and select the **Ultrasonic block**, then drag to the back of **Serial.println**.



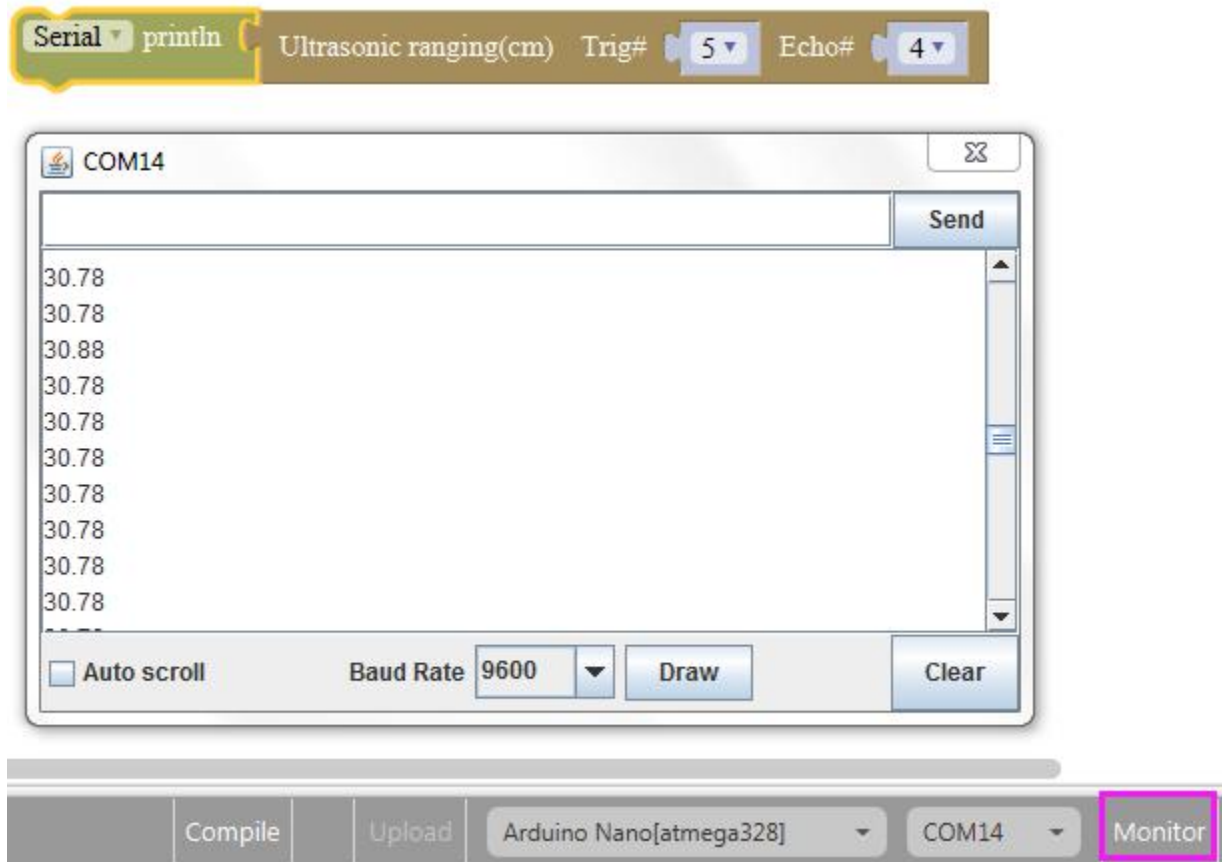
Step 3: Choose the pins for TRIG and ECHO, if the TRIG connect to 4, and ECHO connect to 3, then when you set the numbers, each should plus 1, which means TRIG is 5 and ECHO is 4.



Step 4: Select the **board** type and **port** before uploading.

Step 5: Click **Upload** to run the code. After it, press the power button on the Sloth and keep USB connect with computer.

Step 6: Click the **Monitor** button, you will see the distance appear on the **Serial Monitor**.



Lesson 5 Obstacle Avoidance

Overview

This lesson is to command the Sloth to turn left when obstacles appear in 5cm range.

Programing

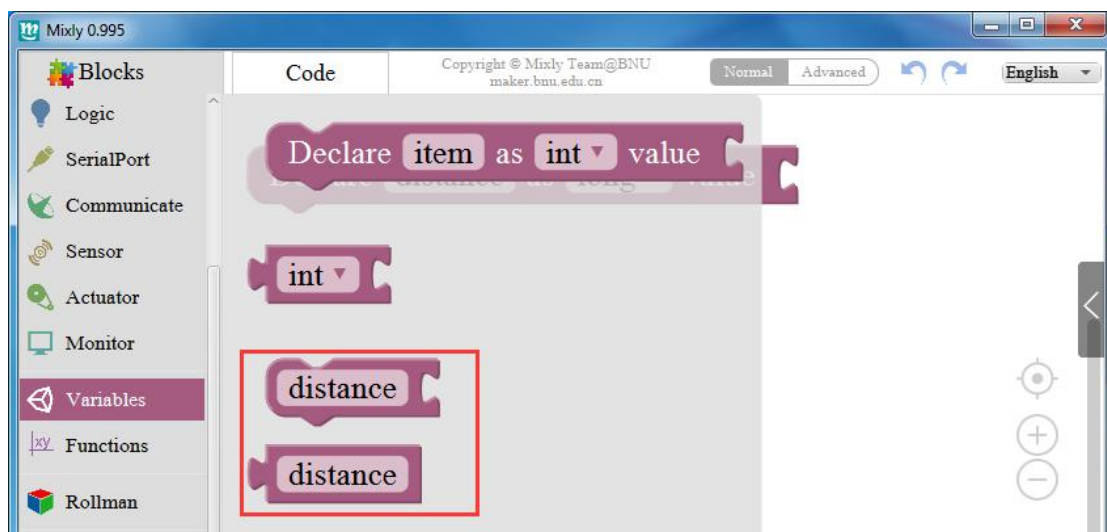
Step 1: Declare a variable **distance** to store the detected distance data.



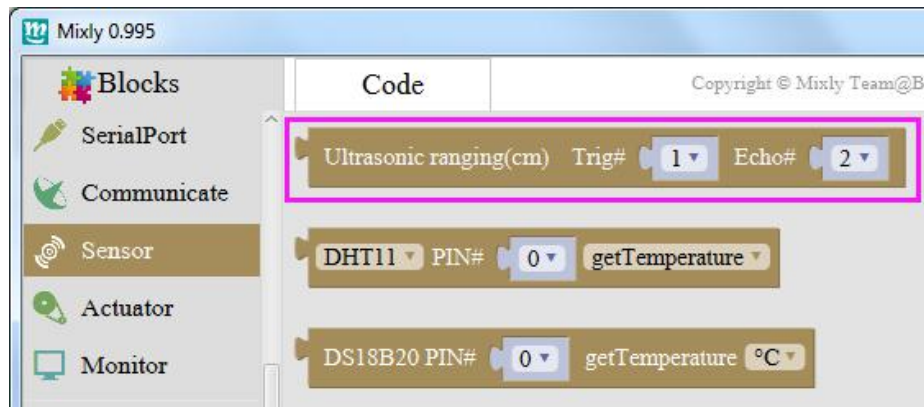
Change **item** into **distance** in the slot:



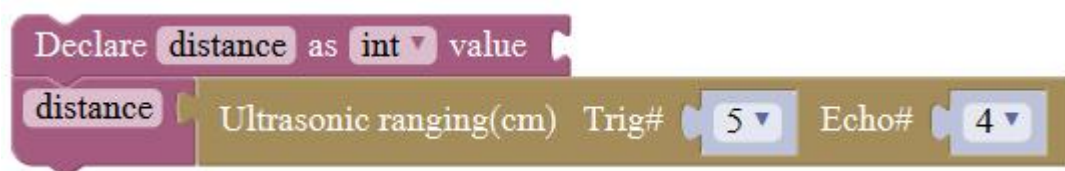
Step 2: Click **Variables** again and you can see two **distance blocks** with different ending patterns: input and output, with slot and bulge respectively.



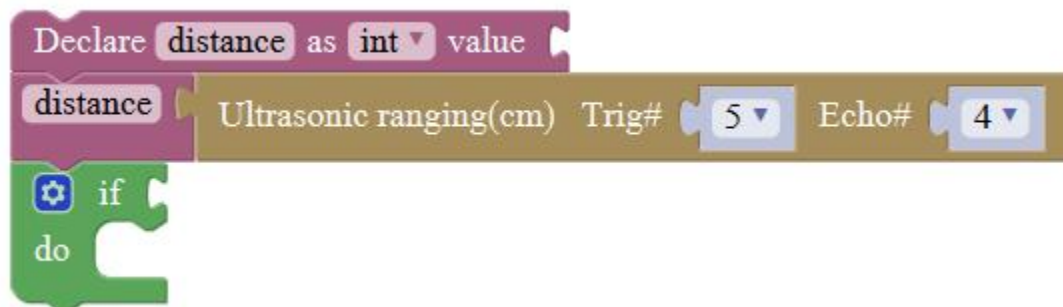
Step 3: Click **Sensor**, and select the **Ultrasonic block**.




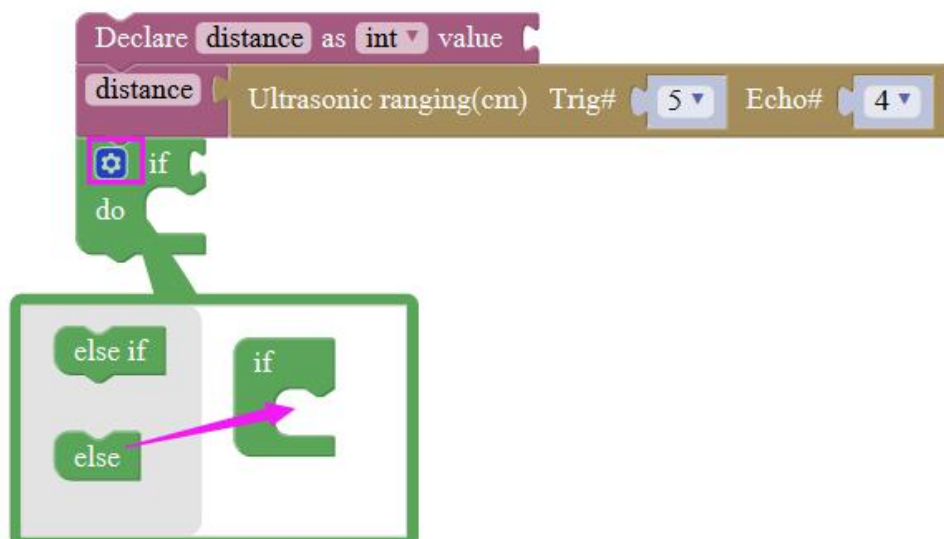
Step 4: Drag it behind the **distance variable** to combine and store the detected distance, then correct the TRIG and ECHO.




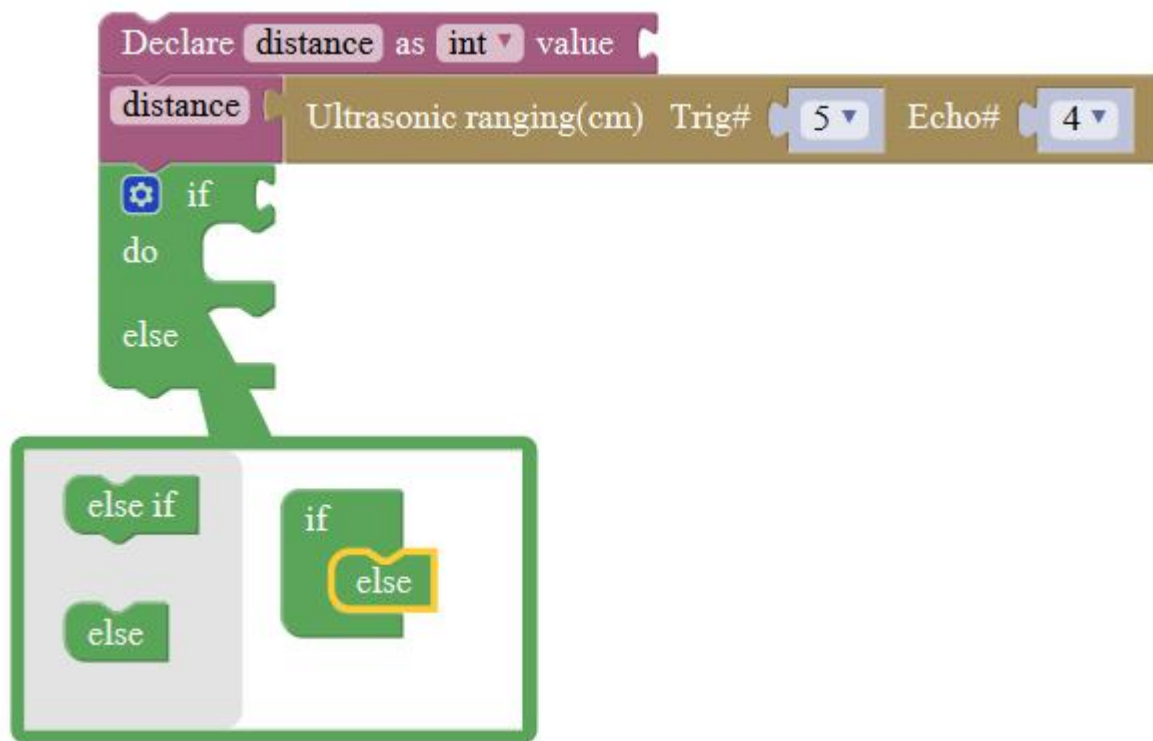
Step 5: To judge whether the detected distance is less than 5cm, click **Control**, drag out an **if_do** block and drop it under **distance category**.



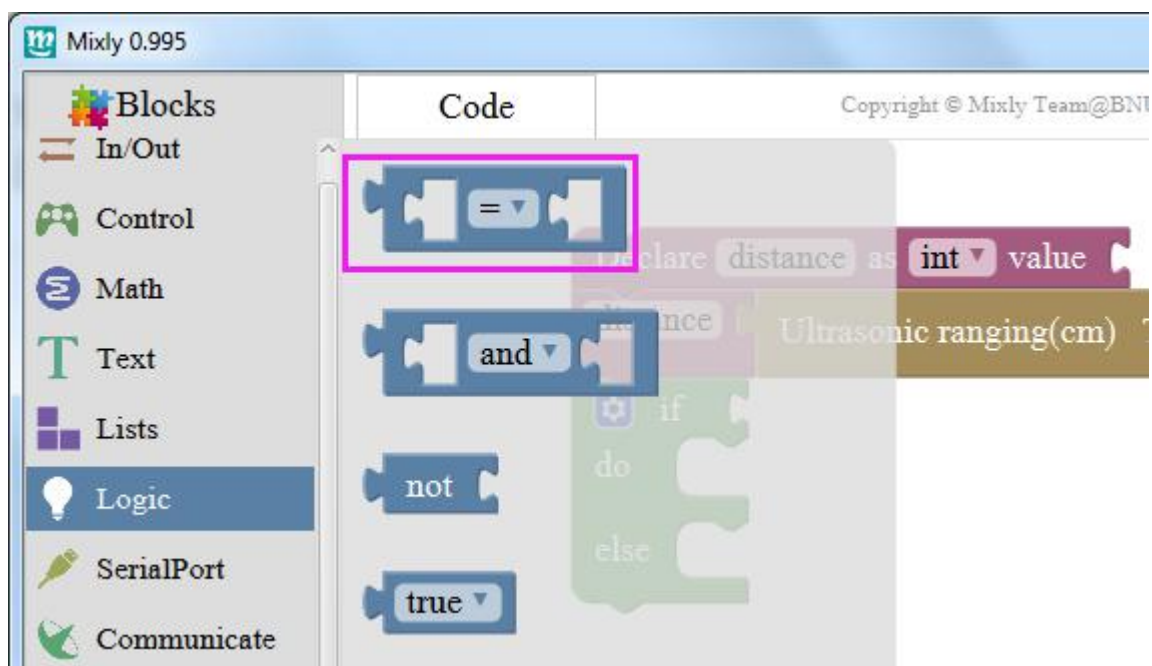
Also add a reverse condition **else**, for what to do if the detected distance is or more than 20cm. Click the setting icon  beside **if**, and drag **else** into the right **if** block.



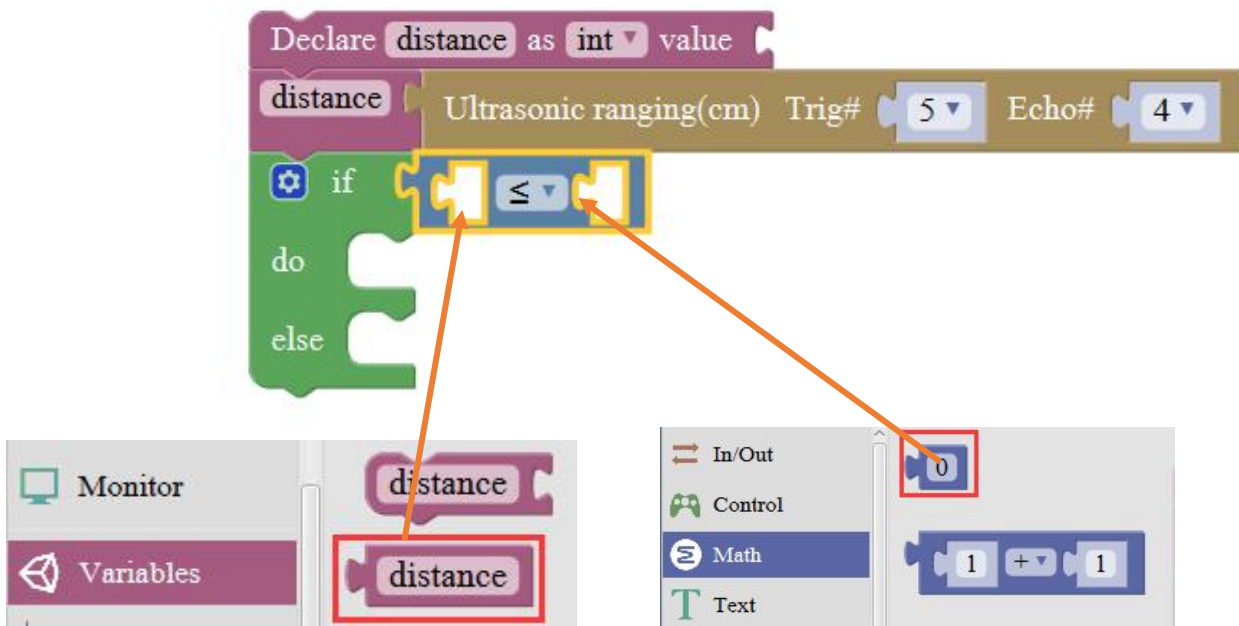
Then it becomes **if_do_else** block, click the  icon again to fold the pop-up.



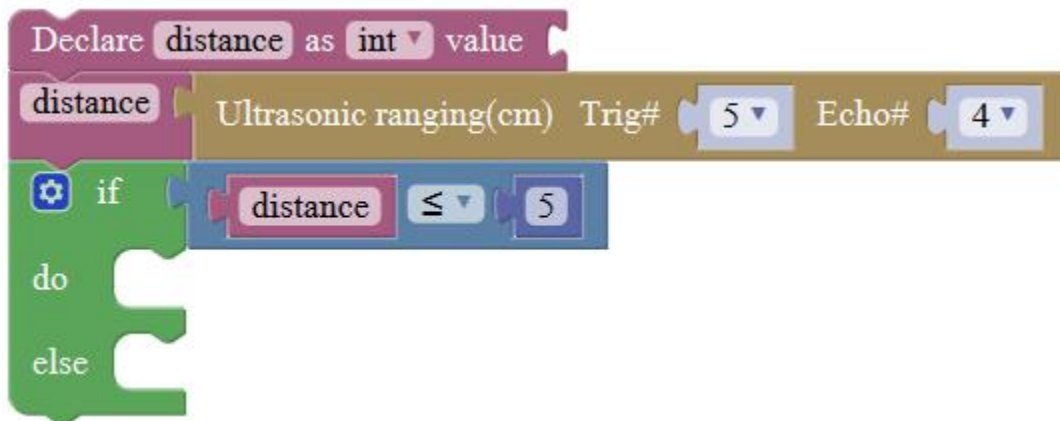
Step 6: Now, let's edit the less than 5 judgement, click **Logic**, and drag the first block behind the **if** block to combine.



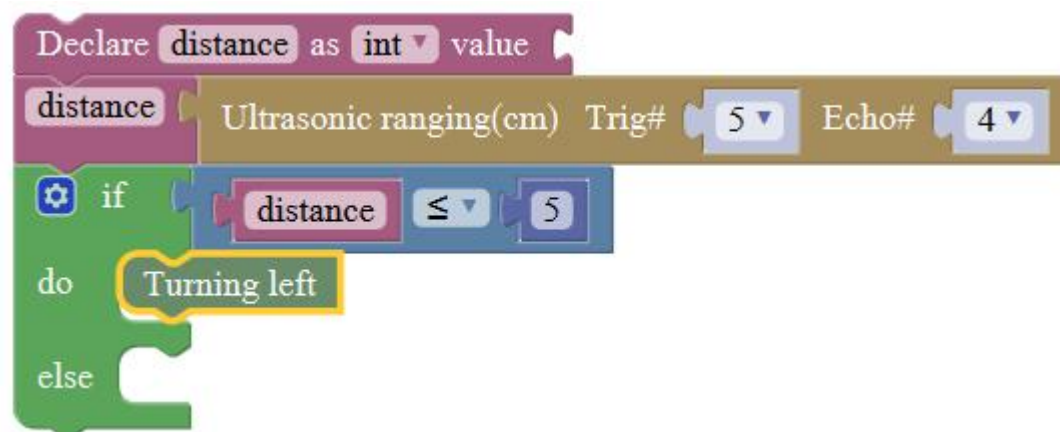
Step 7: Click the triangle icon next `=` to select `<` on the list, drag an output/bulge **distance** in **Variables** to the first slot, and then a **constant** in **Math** to the second.



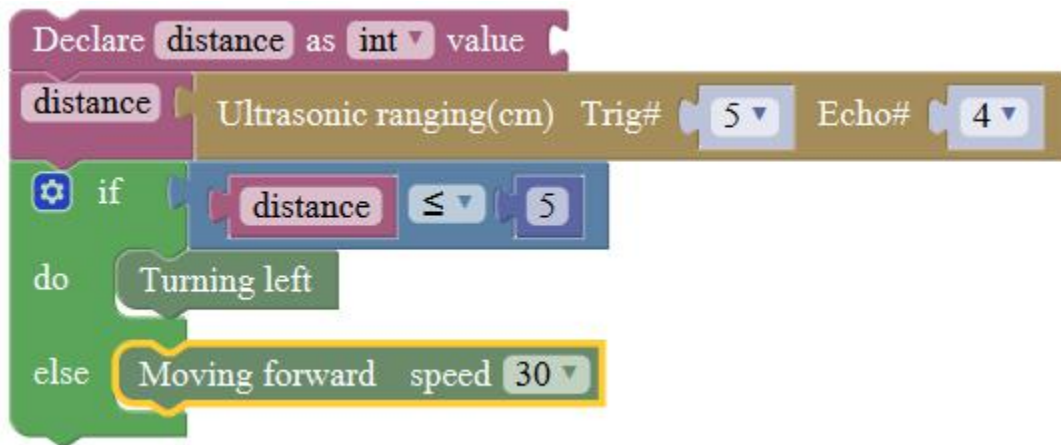
Change **0** into **5** in the slot as shown below:



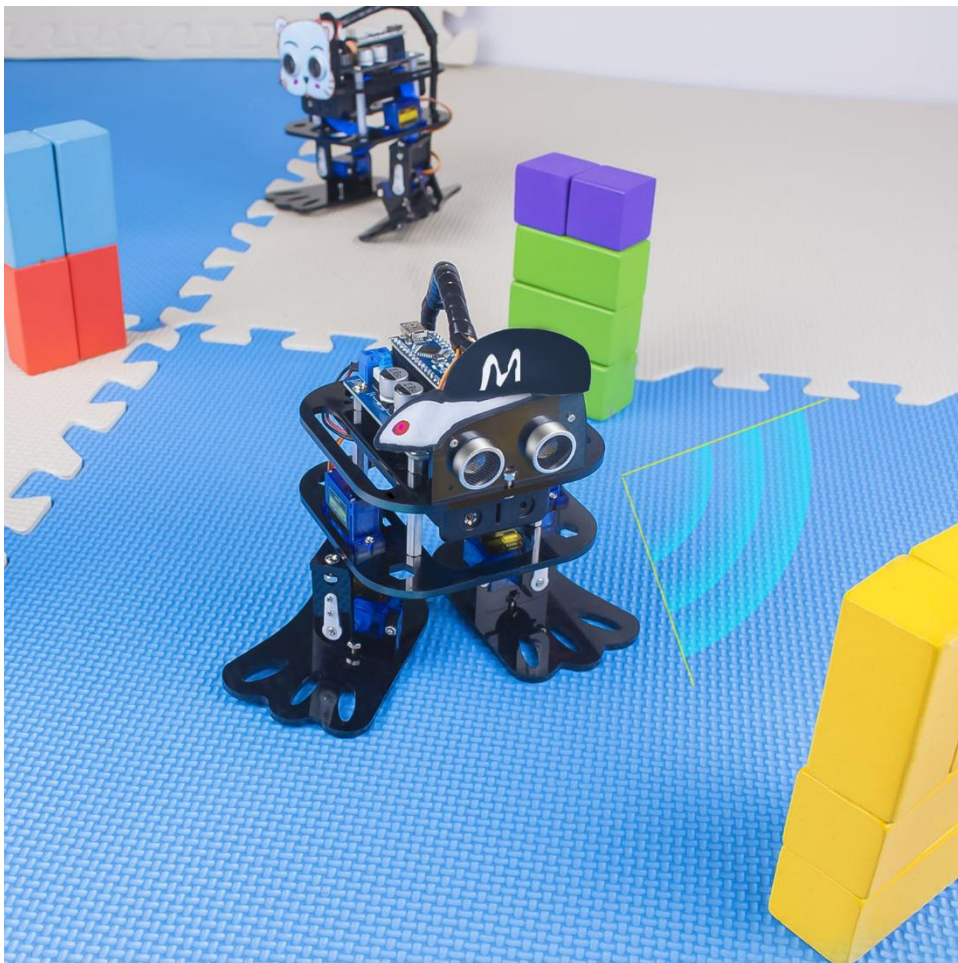
Step 7: Command the sloth to turn left when distance<=5, drag the **turning left block** under the **Dofrobot category**. You can also command it to turn right by **turning right block** if meet obstacles.



Step 8: The **else** is the reverse judgement: command the sloth to move forward when distance>5. Drag the **Moving Forward block**, end the program with **end program block**.



Step 9: Click **Save** after all the programming is done, Select the **board** type and **port** before uploading. Click Upload to run the code. After the code is running successfully, press the power button of the Sloth. When the Sloth is moving forward and there has obstacle forward in 5cm range or you put your hand before it, the Sloth will turn left and then moves forward.



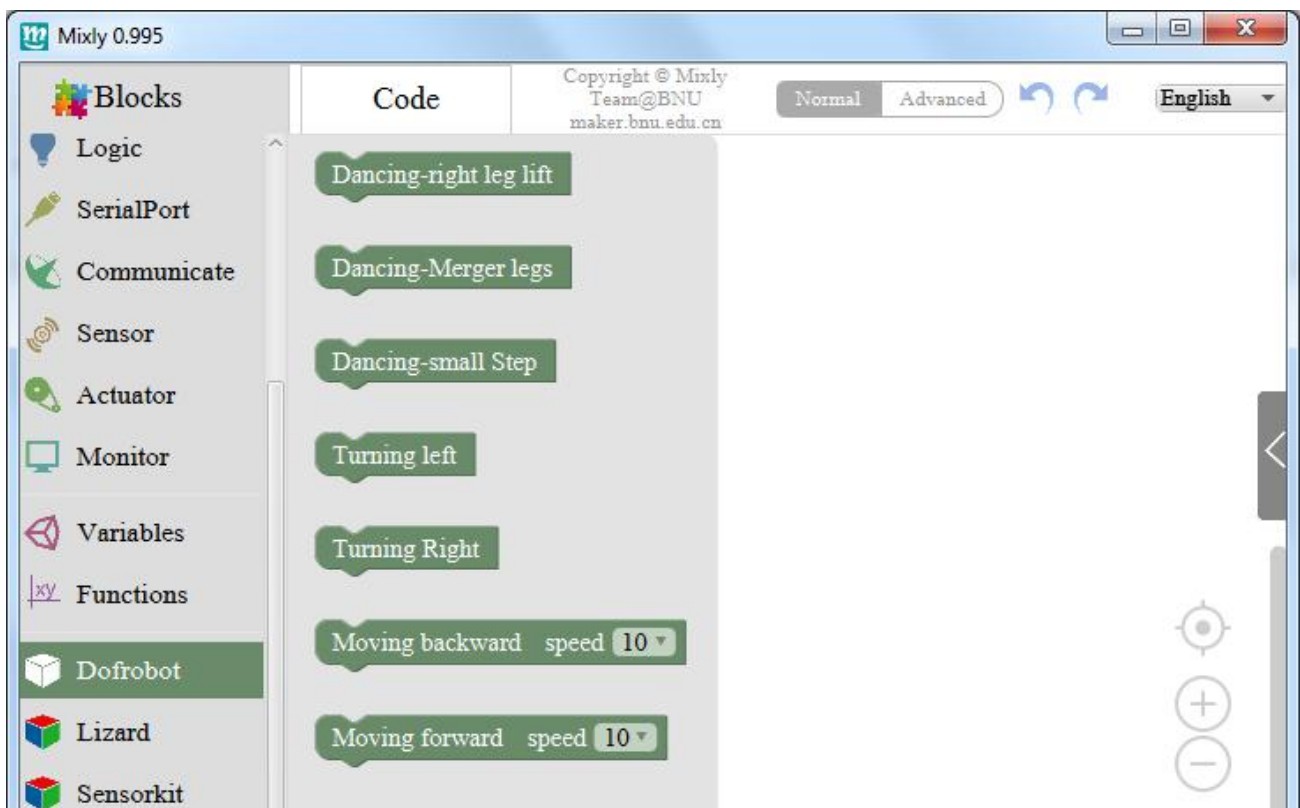
Lesson 6 Dance

Overview

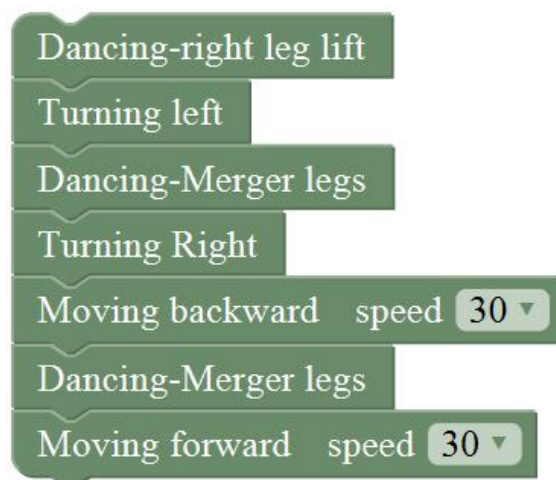
In the **Dofrobot category**, there have many different blocks wrote to command the Sloth to do different actions. The Sloth can dance if combining these blocks together.

Programing

Step 1: Click the **Dofrobot category**, drug out these blocks if you want.



Step 2: Combine them together.



Step 3: Click **Save** after all the programming is done, Select the **board** type and **port** before uploading. Click Upload to run the code. After the code running successfully, press the power button on the Sloth. You will see the Sloth starts to dance.

