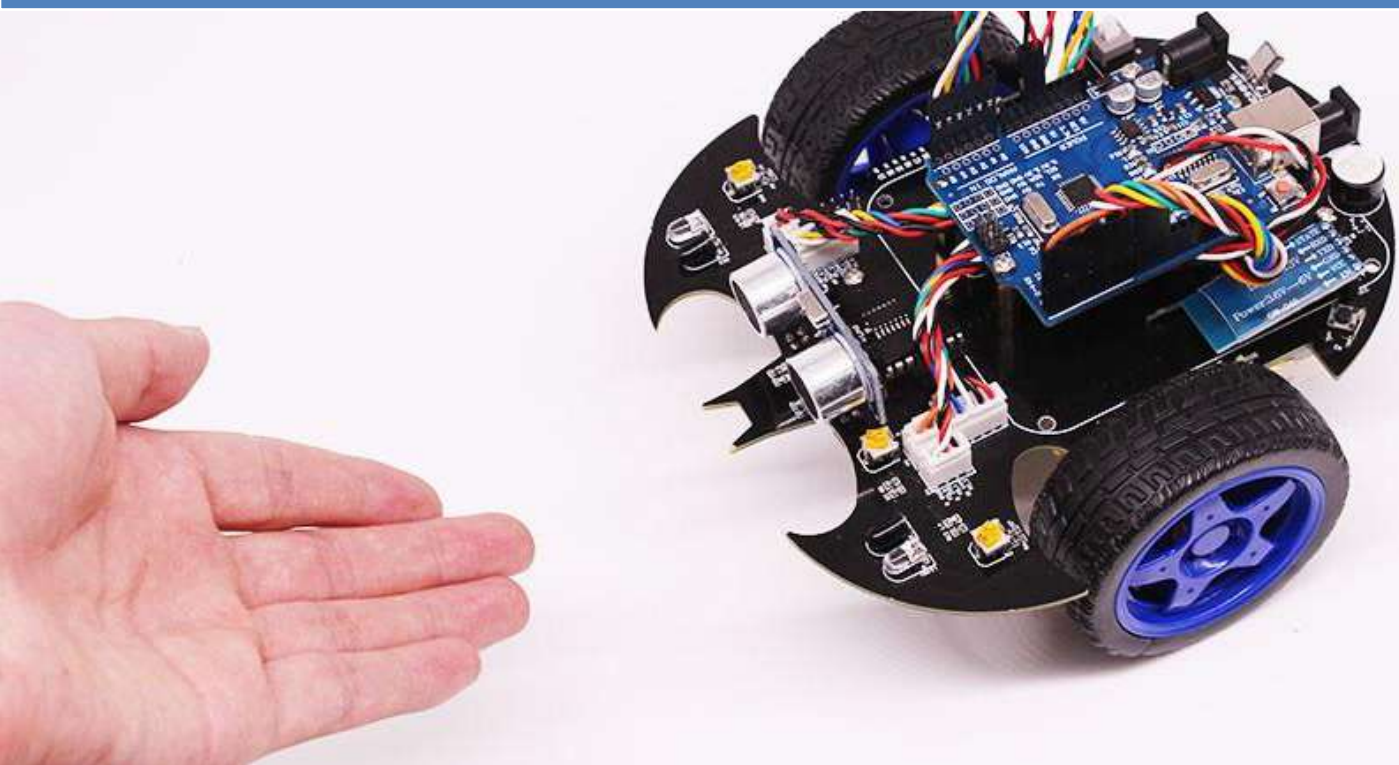


2019

Yahboom Arduino Batmobile

Graphical Programming Tutorials



Graphical programming

1. LED Blinks	2
2. Buzzer sounds	5
3. Buzzer sounds at Different frequencies	10
4. Button control LED	14
5. Ultrasonic distance measurement	17
6. Go ahead	21
7. Button start car	23
8. Control Speed	26
9. Line walking	29
10. Infrared obstacle avoidance	33
11. Tracking	37

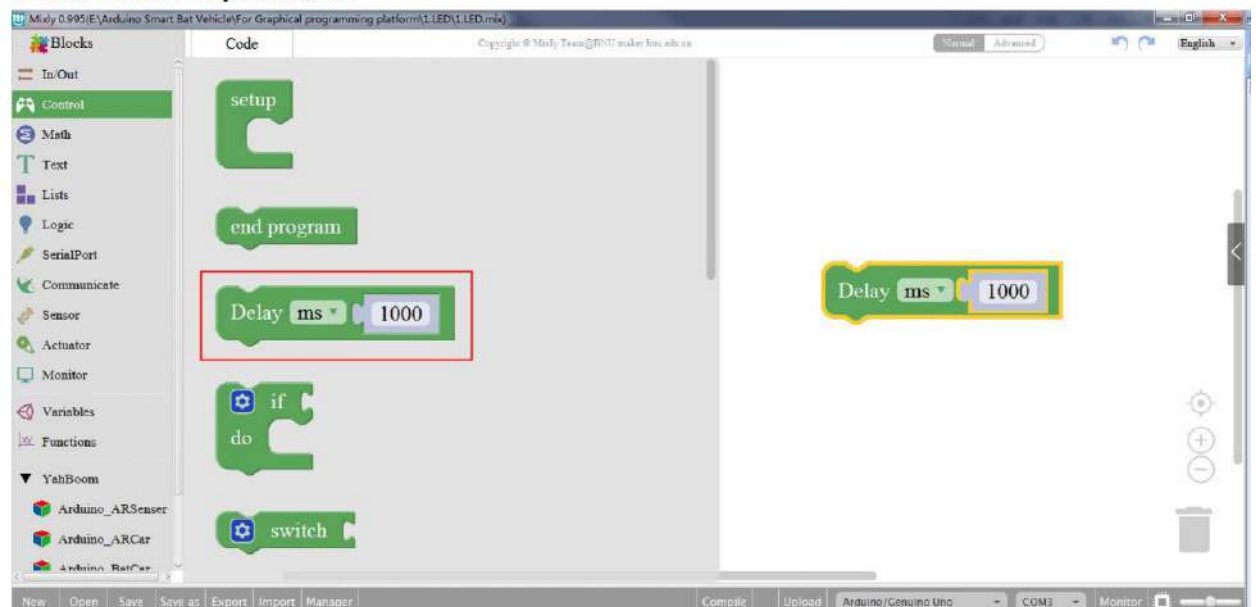
1- LED Blinks

1.LED Blinks

Follow this picture to connect the modules:



1.Modules of experiment:





2. Experiment



(1) Complete the connection of modules and click the button “Compile”. You can see “Compile success” in the dialog window until compile completed. If you see “Compile failed” means that there is something wrong with the connection of modules. Check and fix it.



(2) Choose “Arduino/Genuino Uno” and correct serial port as shown.



(3) Now that configuration is complete, you can click the button “Upload”.

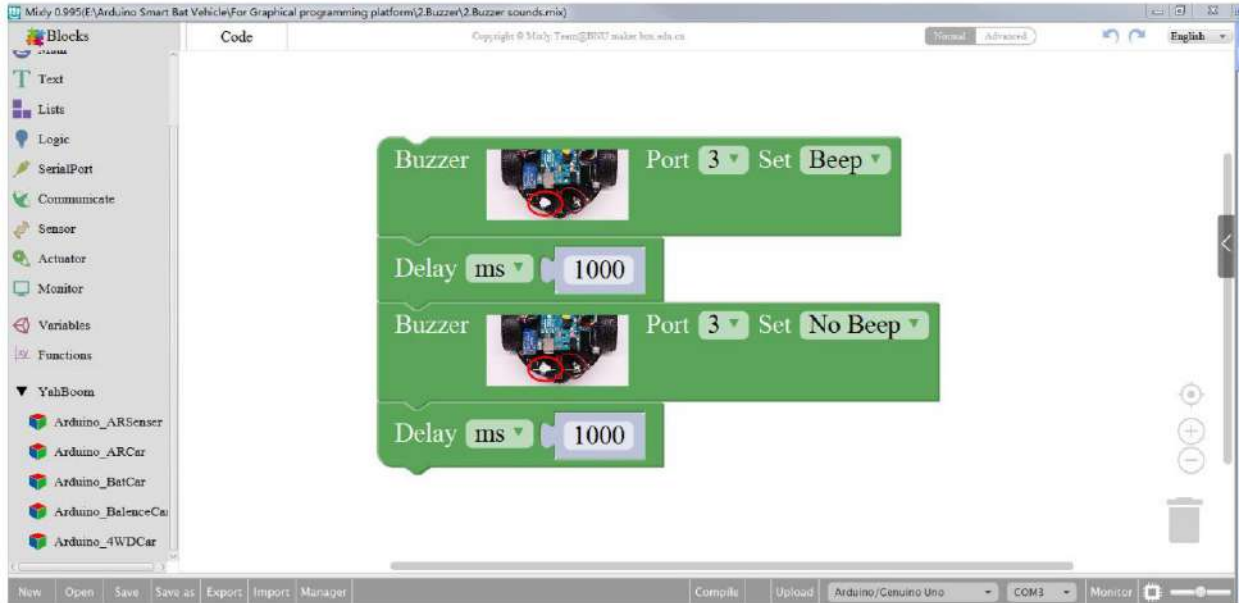


(4) Finally, It's done if you see "Upload success" in the dialog window and the LED blinks for 1s. Let's try different delay time and look at the difference.

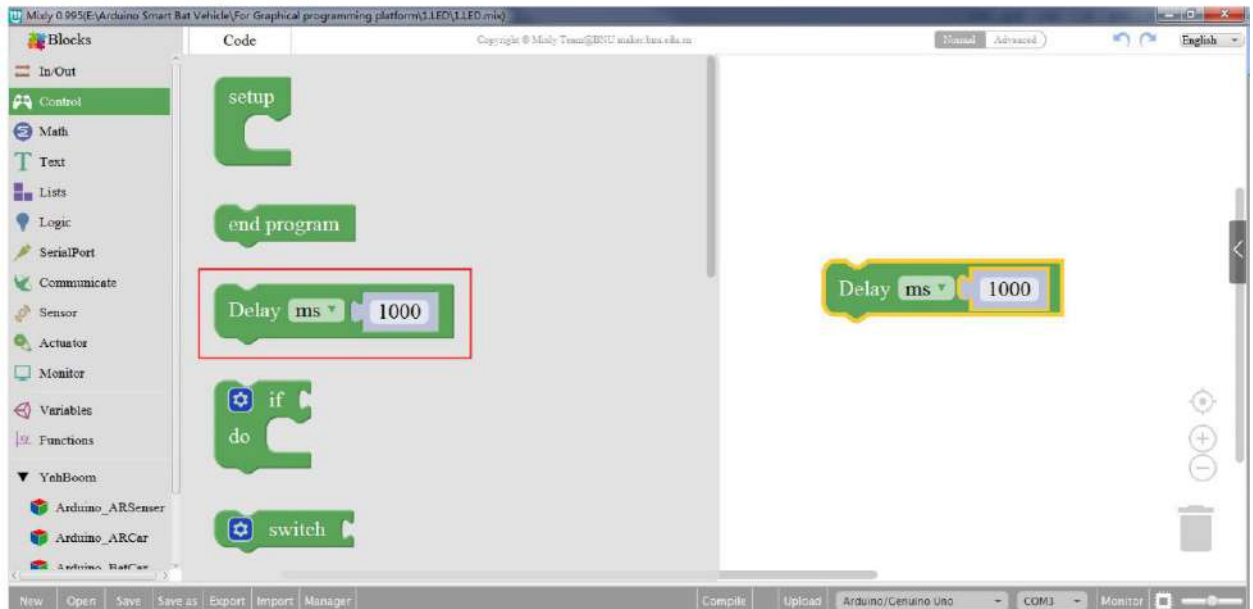
2- Buzzer sounds

2.Buzzer sounds

Follow this picture to connect the modules:

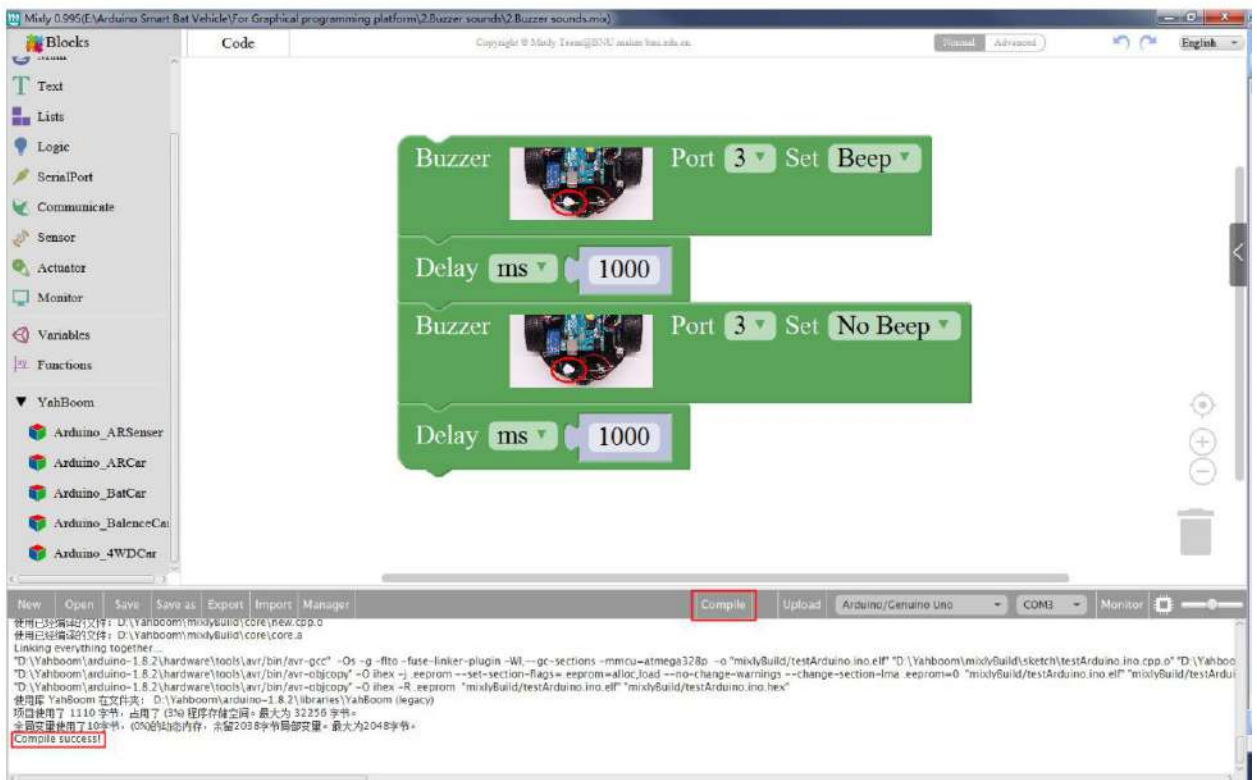


1.Modules of experiment:

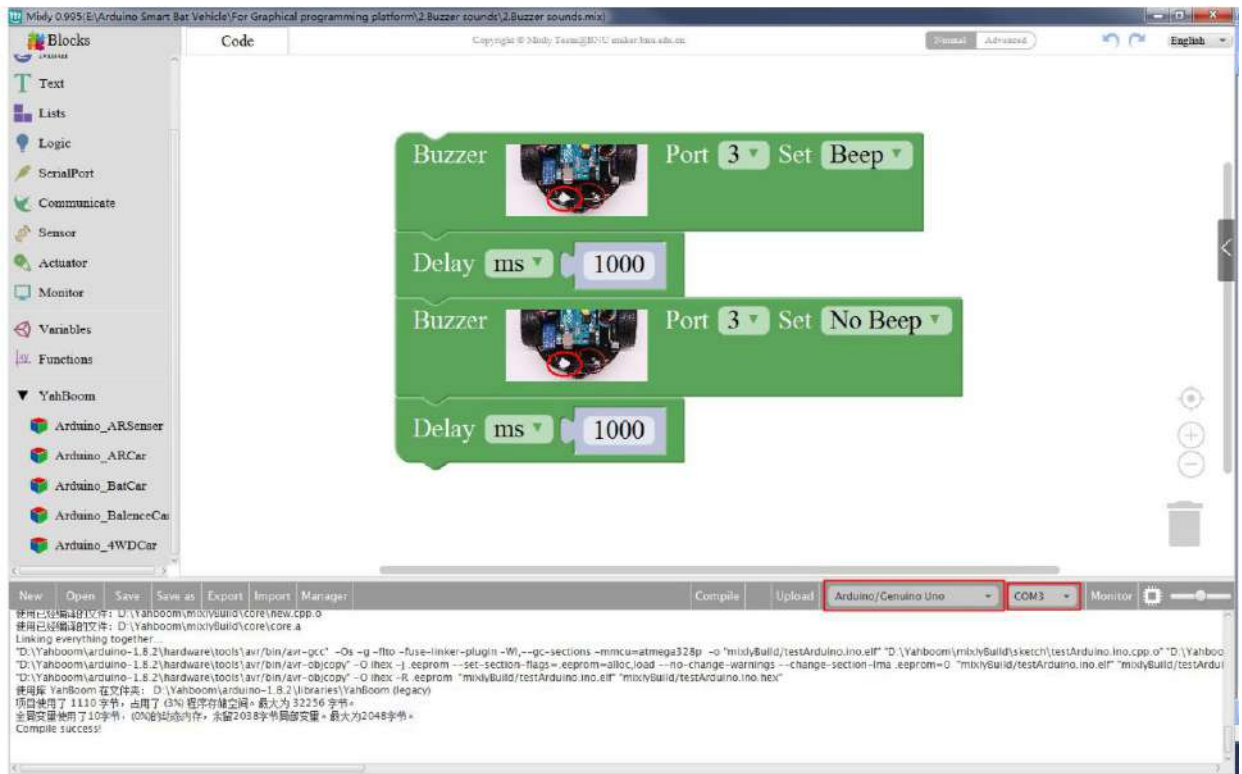




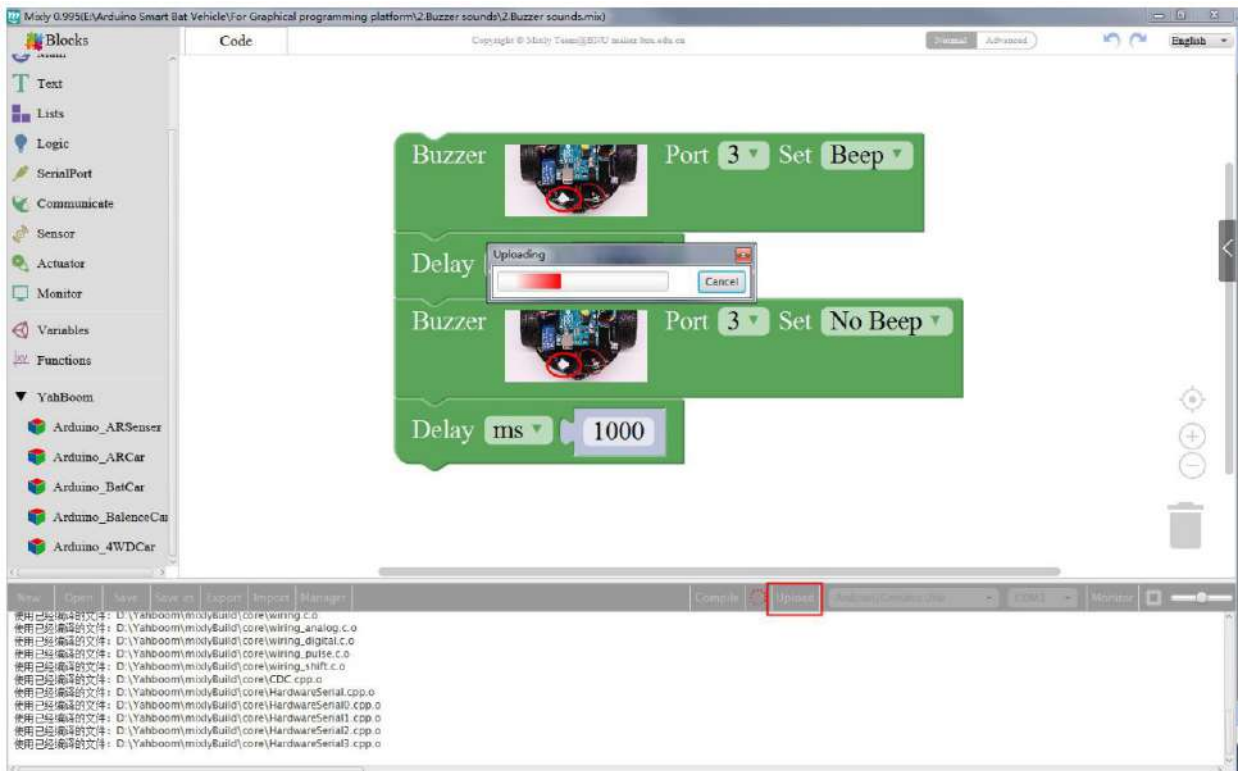
2.Experiment



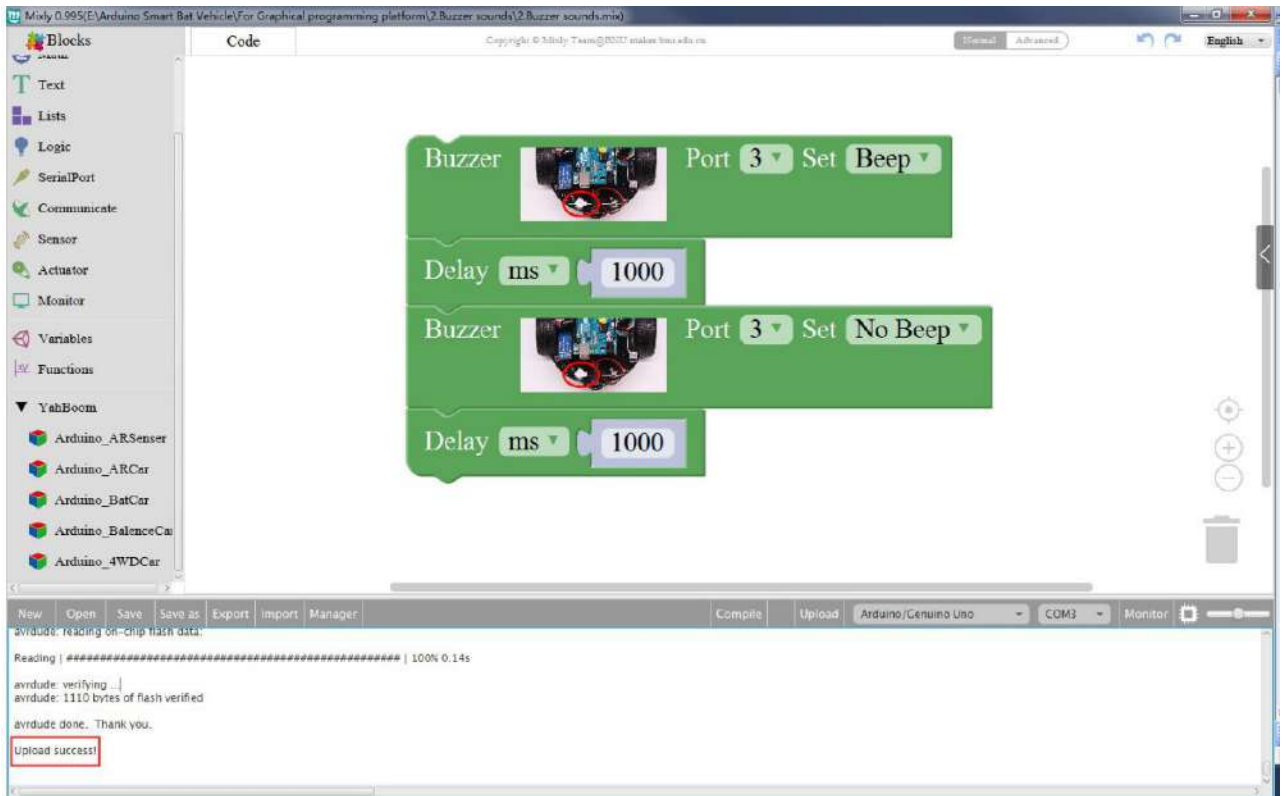
(1) Complete the connection of modules and click the button “Compile”. You can see “Compile success” in the dialog window until compile completed. If you see “Compile failed” means that there is something wrong with the connection of modules. Check and fix it.



(2) Choose “Arduino/Genuino Uno” and correct serial port as shown.



(3) Now that configuration is complete, you can click the button “Upload”.



(4) Finally, It's done if you see "Upload success" in the dialog window and the buzzer sounds for 1s. Let's try different delay time and look at the difference.

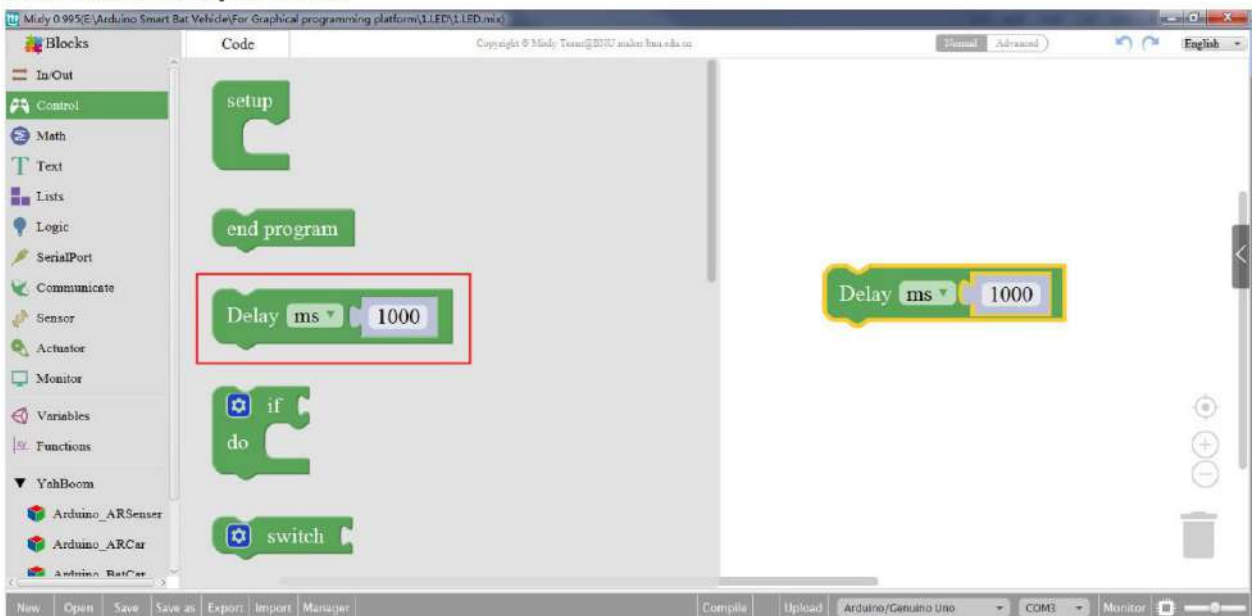
3- Buzzer sounds at Different frequencies

3. Buzzer sounds at Different frequencies

Follow this picture to connect the modules:

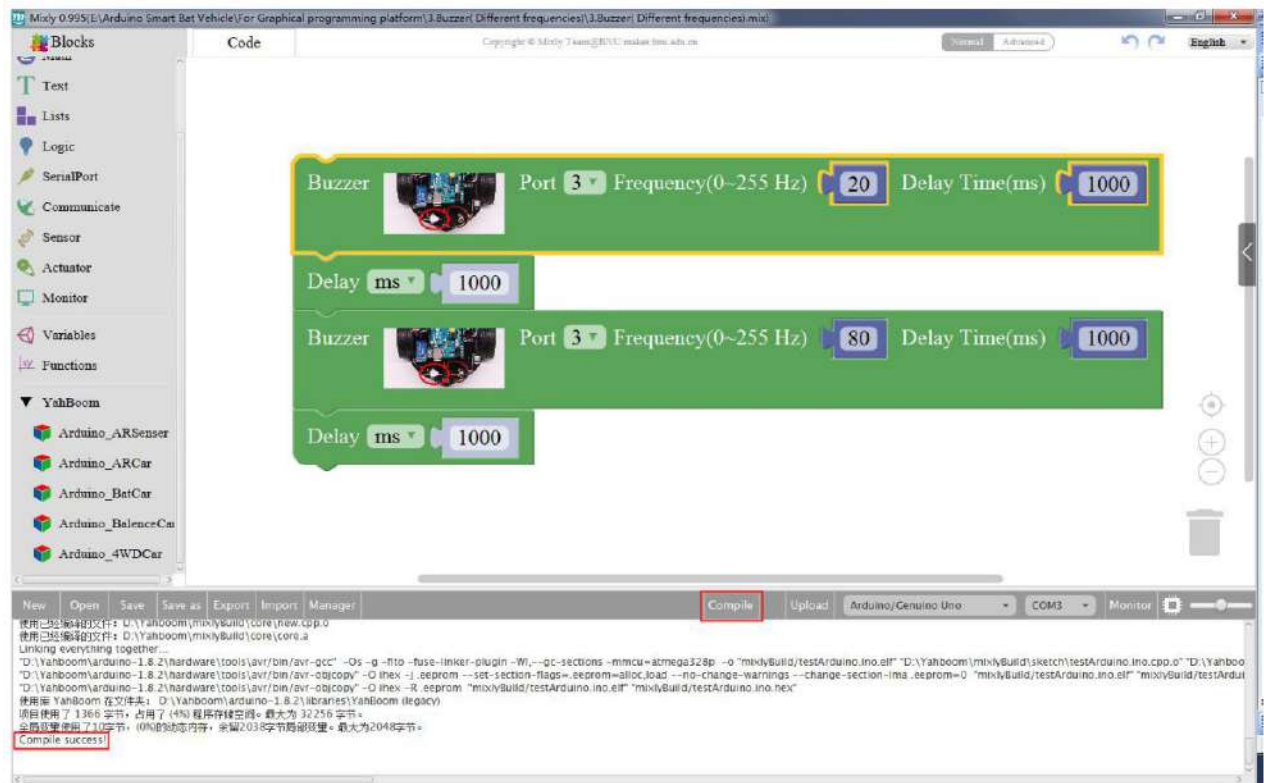


1. Modules of experiment:

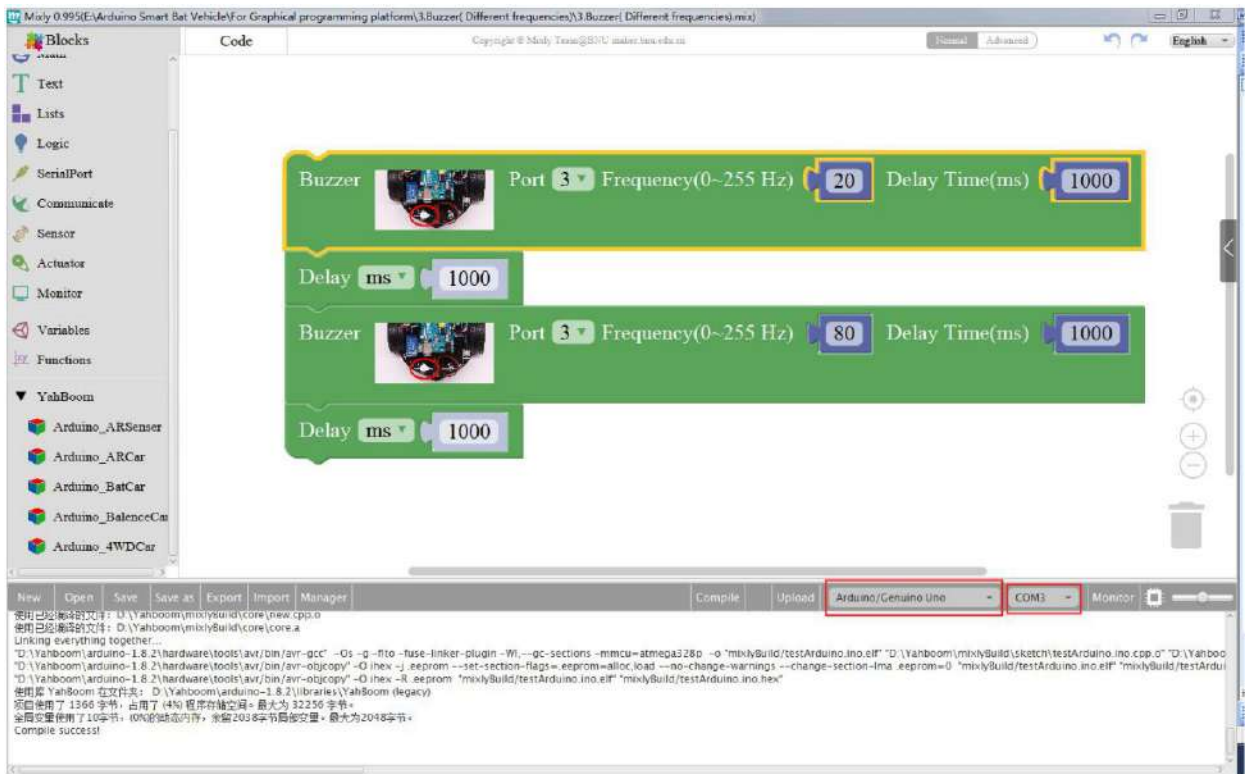




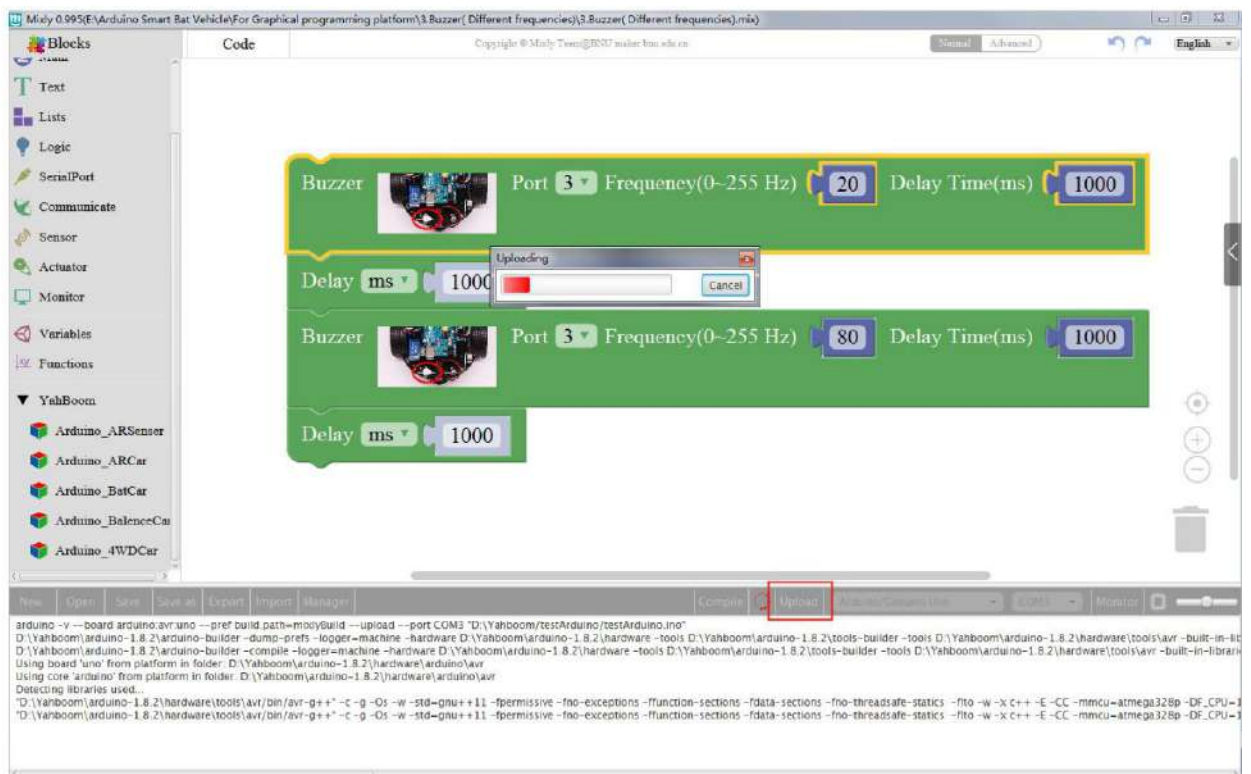
2 .Experiment



(1) Complete the connection of modules and click the button “Compile”. You can see “Compile success” in the dialog window until compile completed. If you see “Compile failed” means that there is something wrong with the connection of modules. Check and fix it.



(2) Choose “Arduino/Genuino Uno” and correct serial port as shown.



(3) Now that configuration is complete, you can click the button “Upload”.

The screenshot displays the Arduino IDE interface with a Scratch-style block-based program. The program consists of four blocks: a Buzzer block (Port 3, Frequency 20 Hz, Delay 1000 ms), a Delay block (1000 ms), another Buzzer block (Port 3, Frequency 80 Hz, Delay 1000 ms), and a final Delay block (1000 ms). The interface includes a left sidebar with 'Blocks' and 'YahBoom' categories, a top menu bar, and a bottom console window showing the upload process.

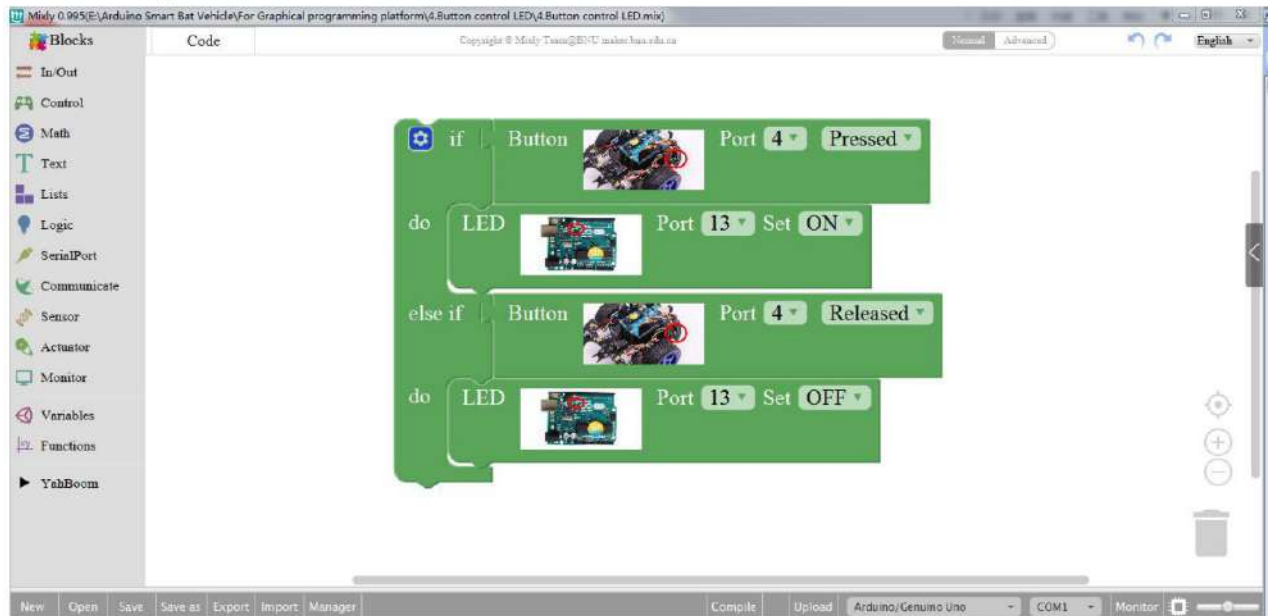
```
avrdude: reading on-chip flash data...
Reading | ##### | 100% 0.17s
avrdude: verifying ...
avrdude: 1366 bytes of flash verified
avrdude done. Thank you.
Upload success!
```

(4) Finally, It's done if you see "Upload success" in the dialog window and the buzzer sounds different from class "2.Buzzer sounds". Let's try different frequencies and look at the difference.

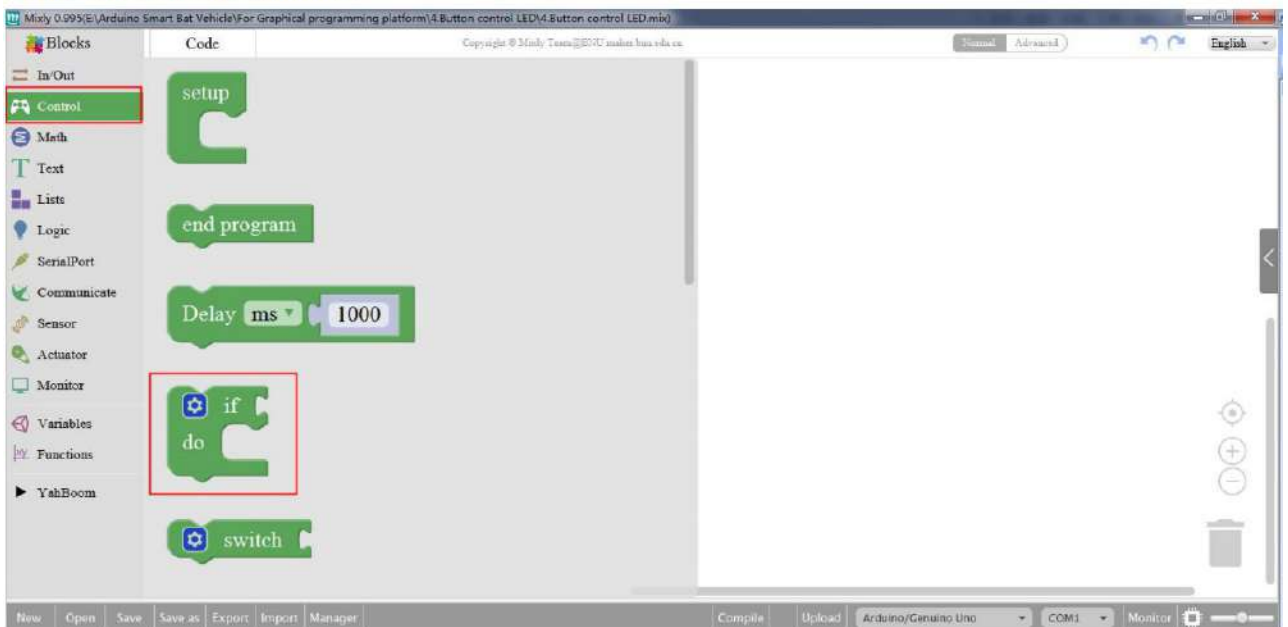
4- Button control LED

4. Button control LED

Follow this picture to connect the modules:

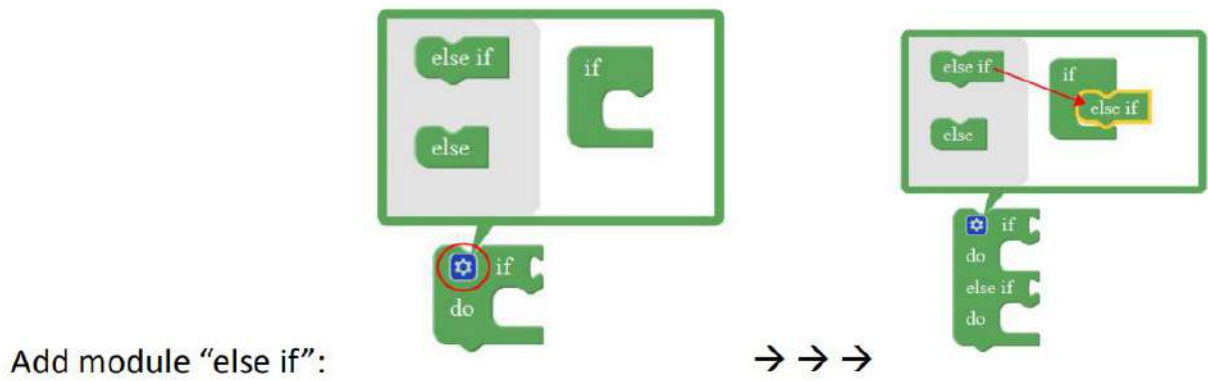
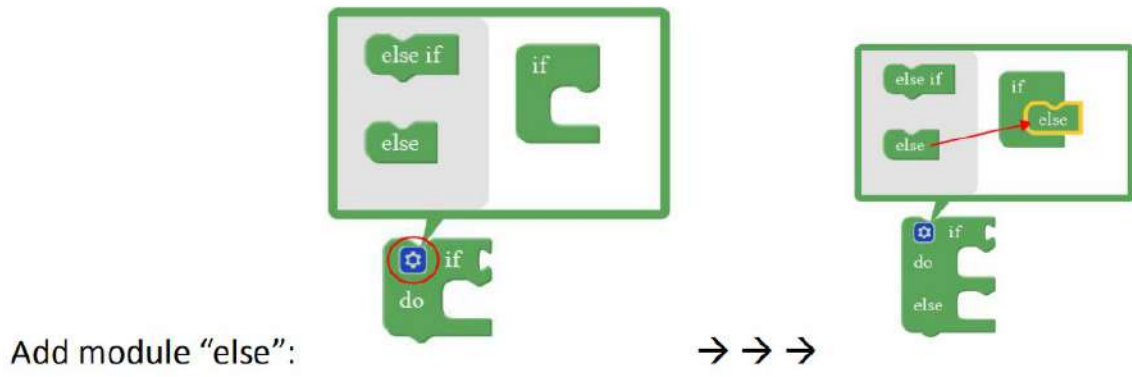


1.Modules of experiment:



Notice for use:

Add module “else” or module “else if” from module “if do” like this.



2 .Experiment



(1) Complete the connection of modules and click the button “Compile”. You can see “Compile success” in the dialog window until compile completed. If you see “Compile failed” means that there is something wrong with the connection of modules. Check and fix it.



(2) Choose “Arduino/Genuino Uno” and correct serial port as shown.



(3) Now that configuration is complete, you can click the button “Upload”.

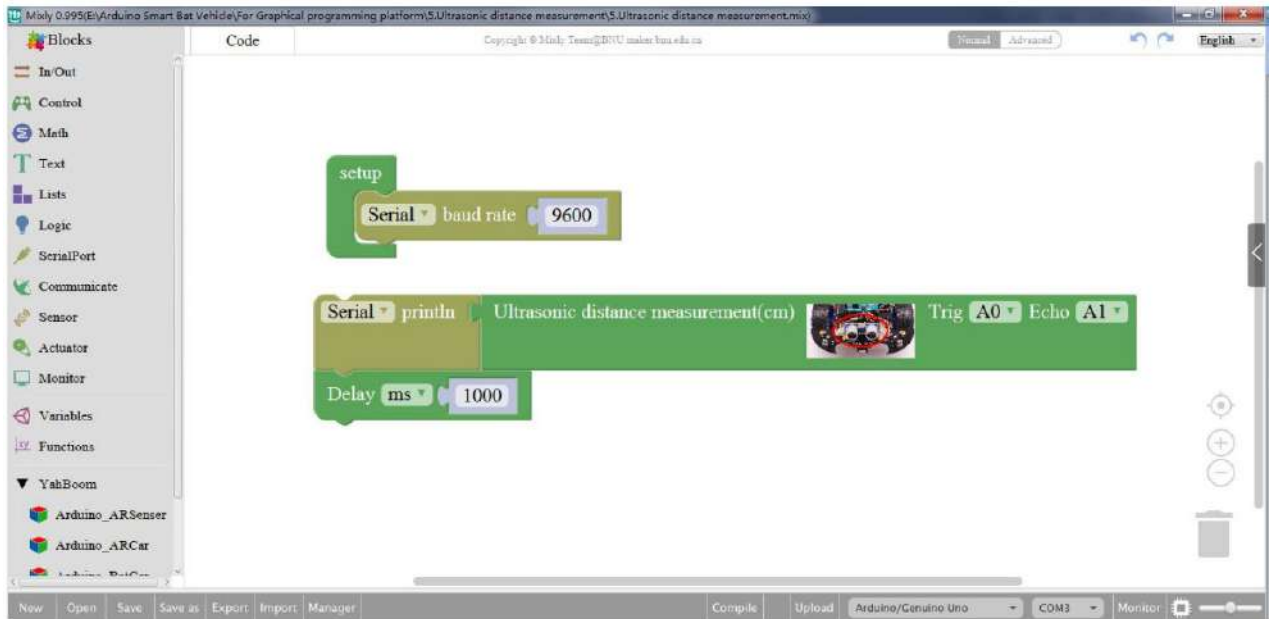


(4) Finally, It’s done if you see “Upload success” in the dialog window. Then you can control the Arduino UNO LED with the button K1 on vehicle board. Push the button, the LED light. Let it go and the LED light goes off. Expand Experiment: Can you try to control the buzzer sounds or not with the button? Let’s try to do it.

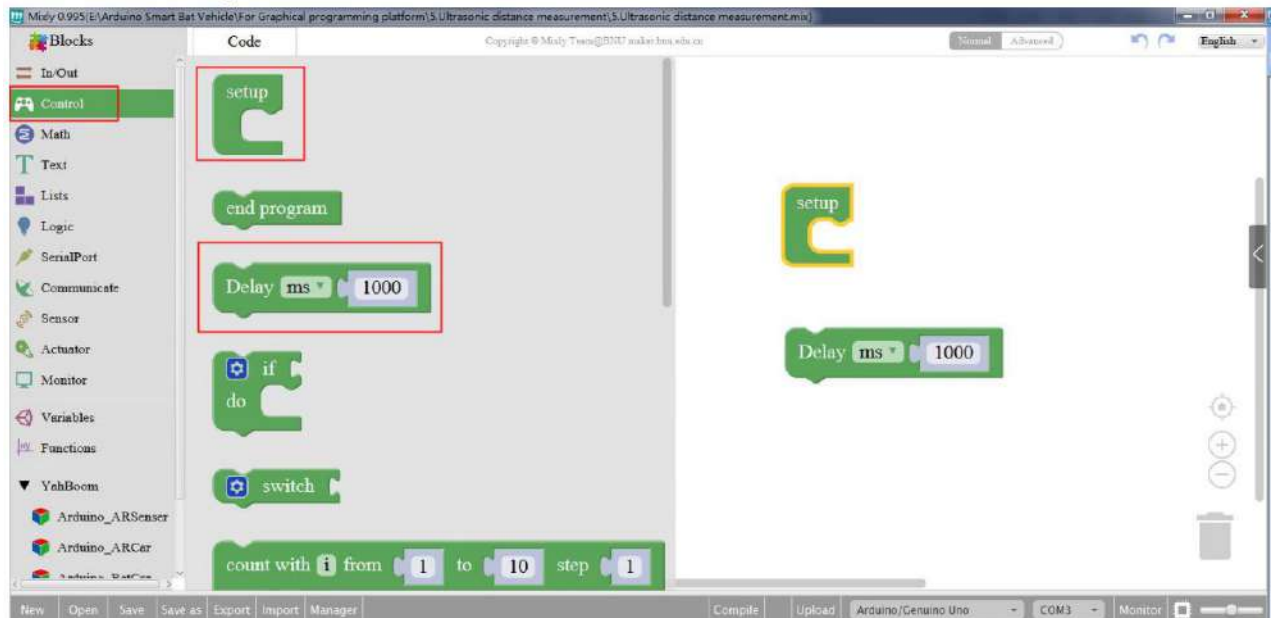
5- Ultrasonic distance measurement

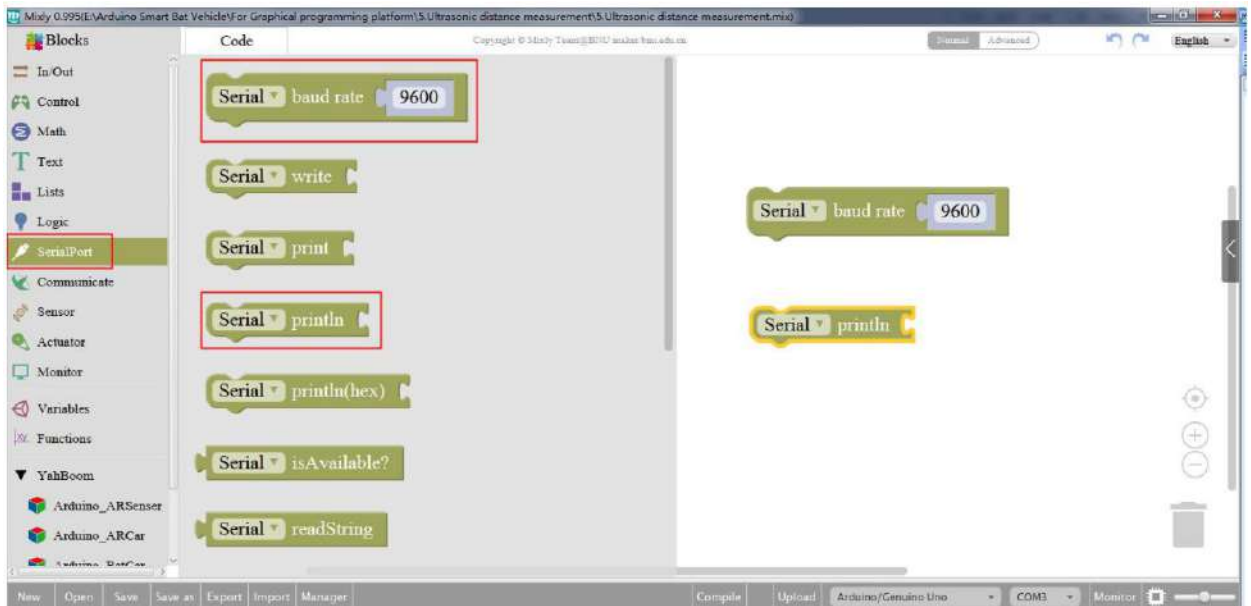
5. Ultrasonic distance measurement

Follow this picture to connect the modules:



1. Modules of experiment:





2 .Experiment



(1) Complete the connection of modules and click the button “Compile”. You can see “Compile success” in the dialog window until compile completed. If you see “Compile failed” means that there is something wrong with the connection of modules. Check and fix it.



(2) Choose “Arduino/Genuino Uno” and correct serial port as shown.

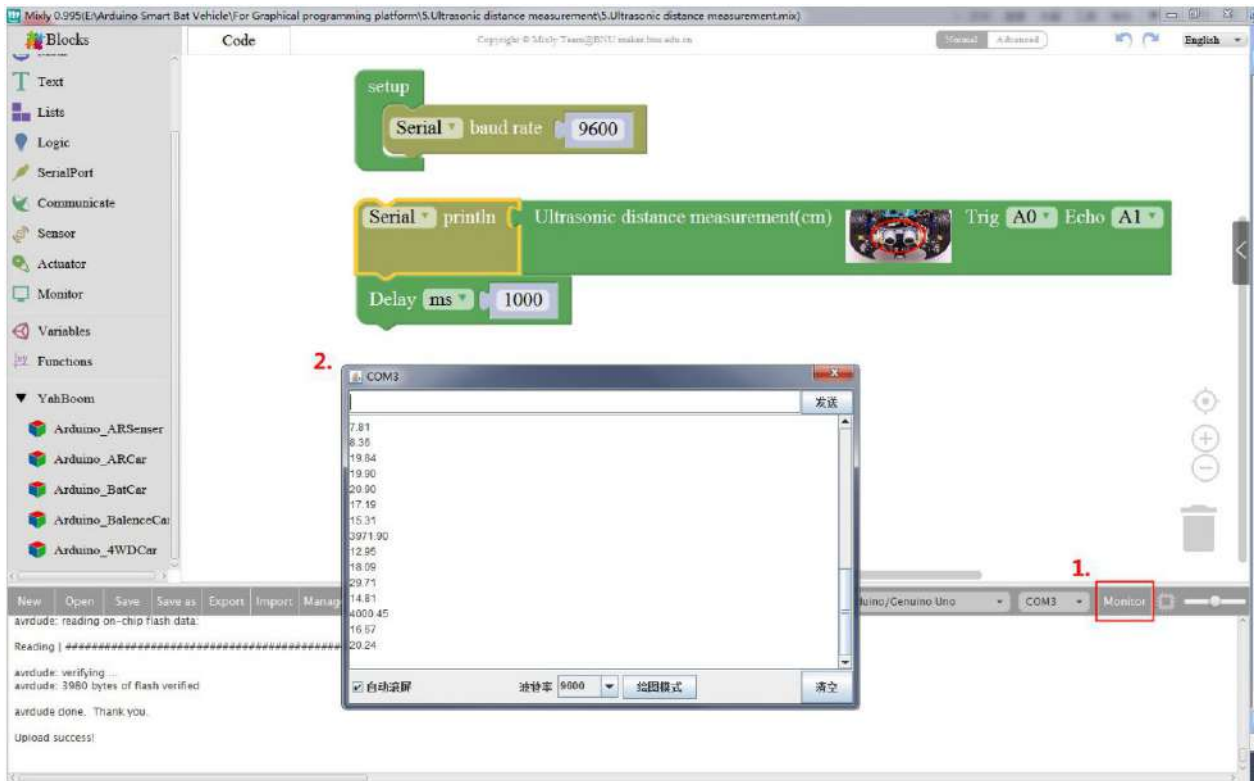


(3) Now that configuration is complete, you can click the button “Upload”.



(4) Finally, It's done if you see “Upload success” in the dialog window.

(5) Click “Monitor” and see the distance measurement.



Expand Experiment: Can you try to control the buzzer sounds to show the distance measurement like parking sensor warning? The buzzer sounds quicker and quicker as the distance get closer and closer. Let's try to do it.

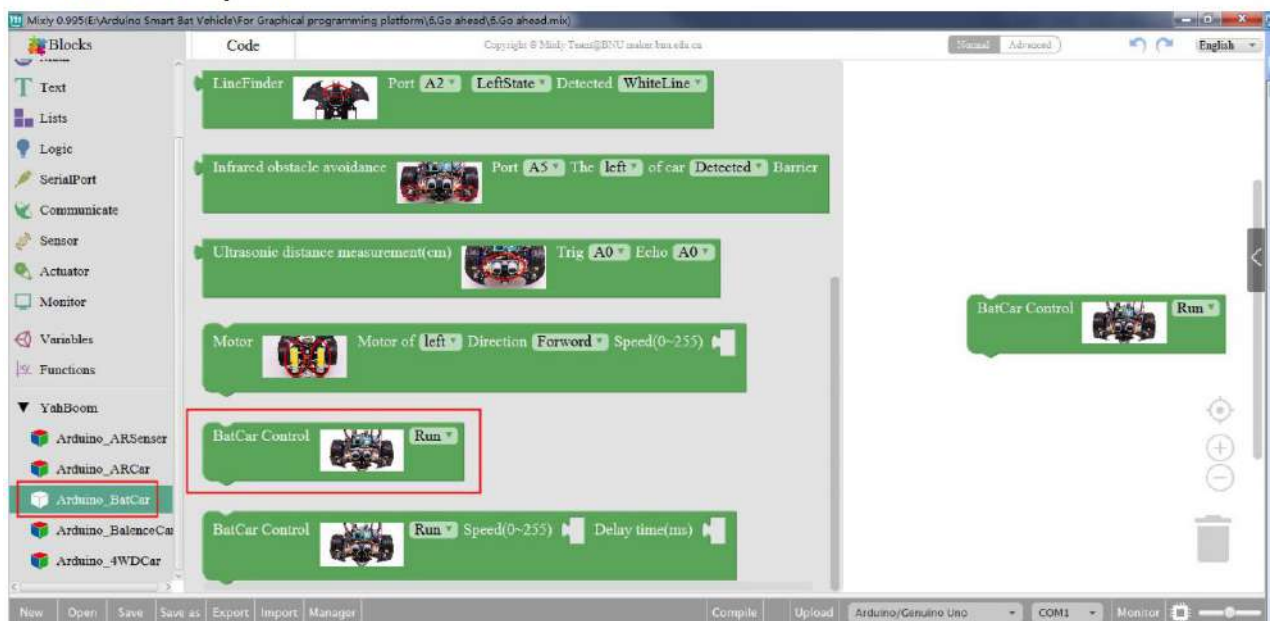
6- Go ahead

6.Go ahead

Follow this picture to connect the modules:



1.Modules of experiment:



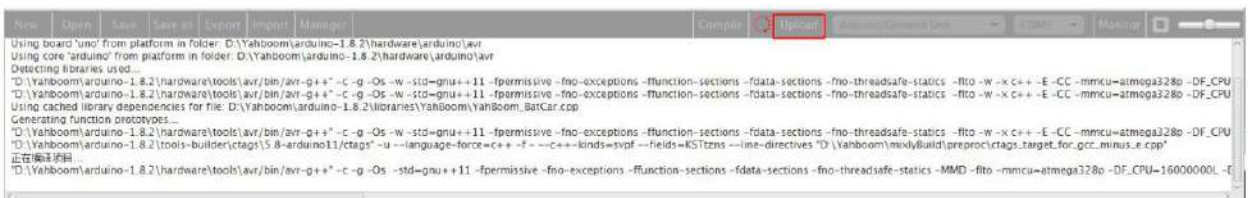
2 .Experiment



(1) Complete the connection of modules and click the button “Compile”. You can see “Compile success” in the dialog window until compile completed. If you see “Compile failed” means that there is something wrong with the connection of modules. Check and fix it.



(2) Choose “Arduino/Genuino Uno” and correct serial port as shown.



(3) Now that configuration is complete, you can click the button “Upload”.



(4) Finally, It’s done if you see “Upload success” in the dialog window. Now unplug the USB line and power on the BatCar. Congratulations! It is running.

Expand Experiment: Can you try to control the BatCar run different directions? Let’s try to do it.



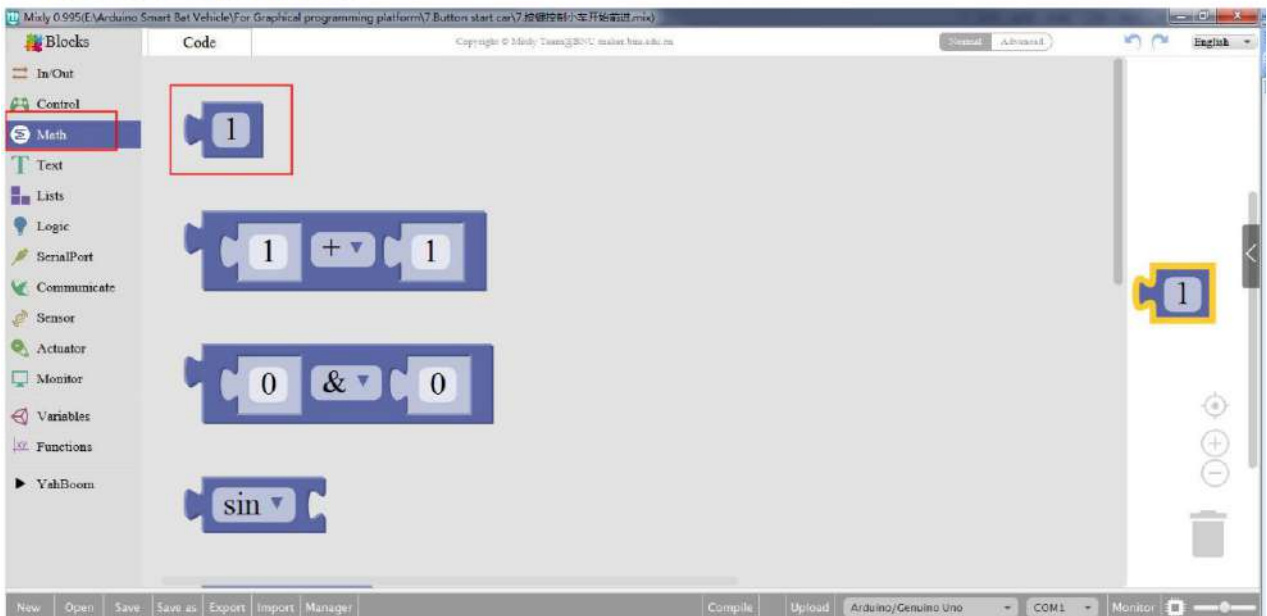
7- Button start car

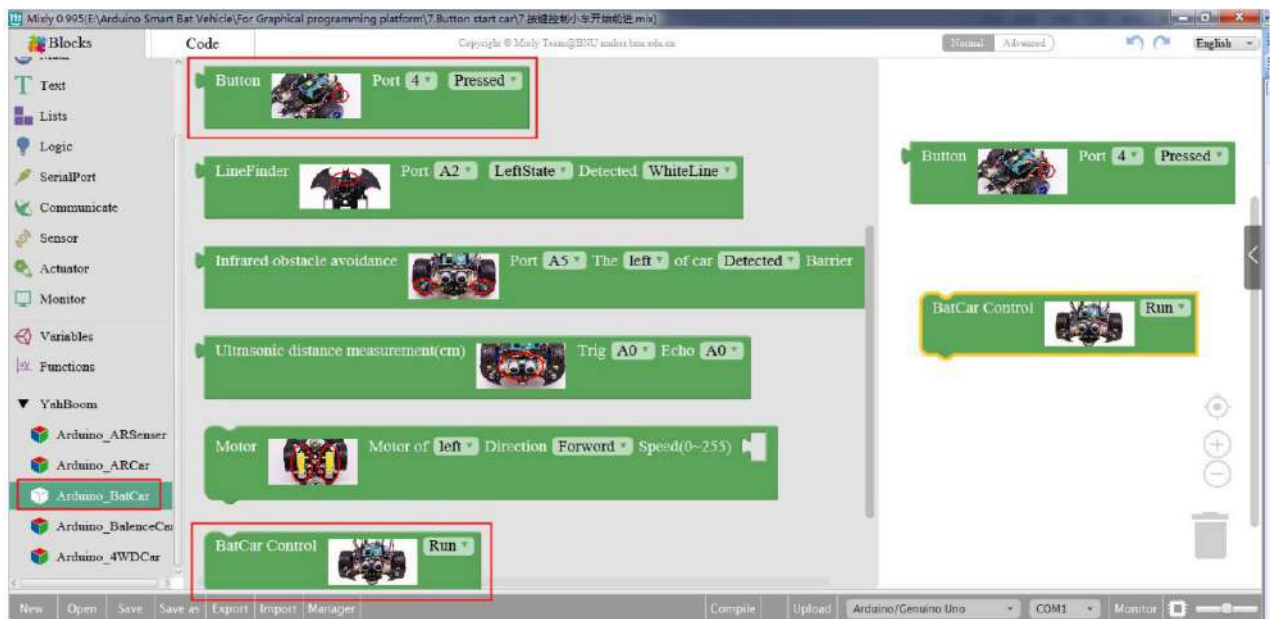
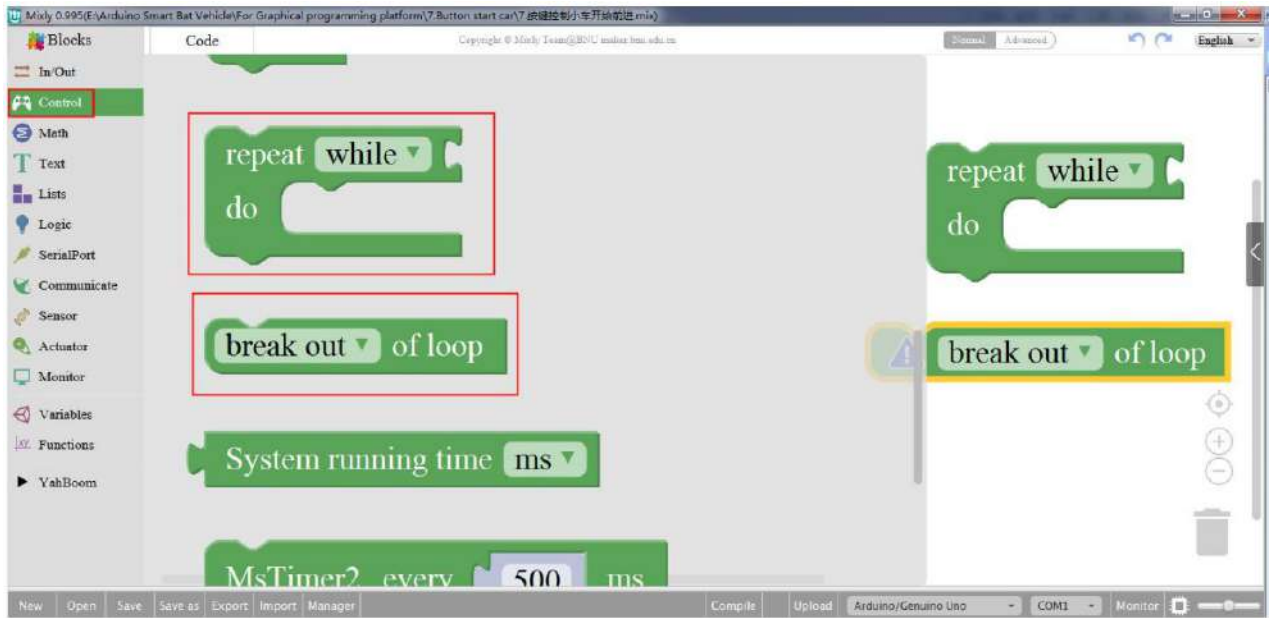
7. Button start car

Follow this picture to connect the modules:



1.Modules of experiment:





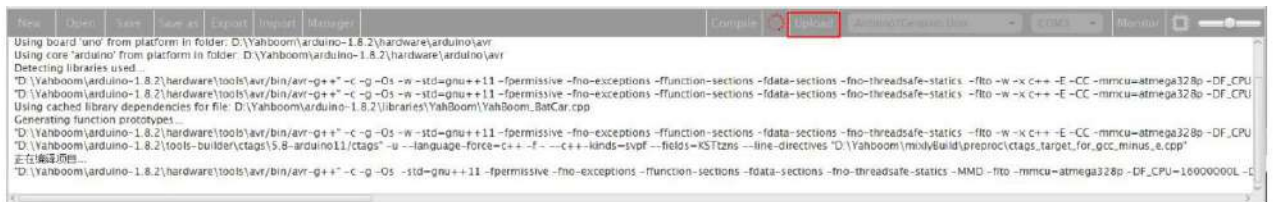
2. Experiment



(1) Complete the connection of modules and click the button “Compile”. You can see “Compile success” in the dialog window until compile completed. If you see “Compile failed” means that there is something wrong with the connection of modules. Check and fix it.



(2) Choose “Arduino/Genuino Uno” and correct serial port as shown.

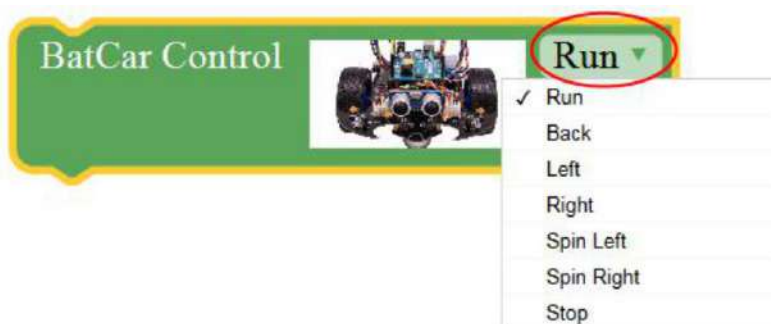


(3) Now that configuration is complete, you can click the button “Upload”.



(4) Finally, It's done if you see “Upload success” in the dialog window. Now unplug the USB line and power on the BatCar. Push the button K1 and the BatCar goes ahead. Well done, it's success.

Expand Experiment: Can you try to control the BatCar run different directions? Let's try to do it.



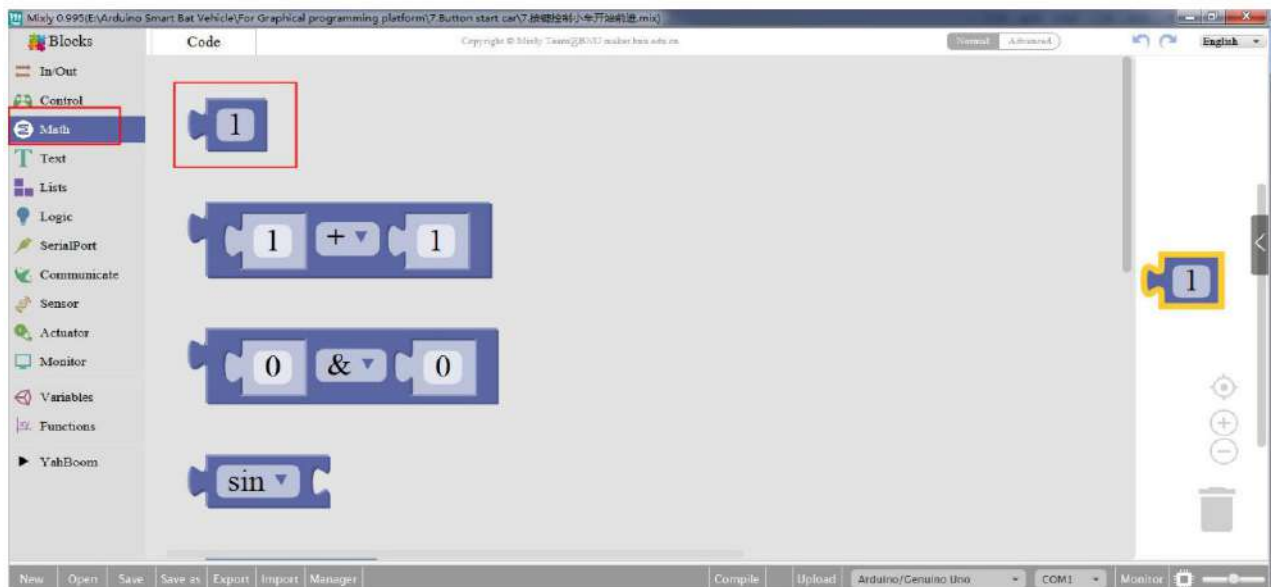
8- Control Speed

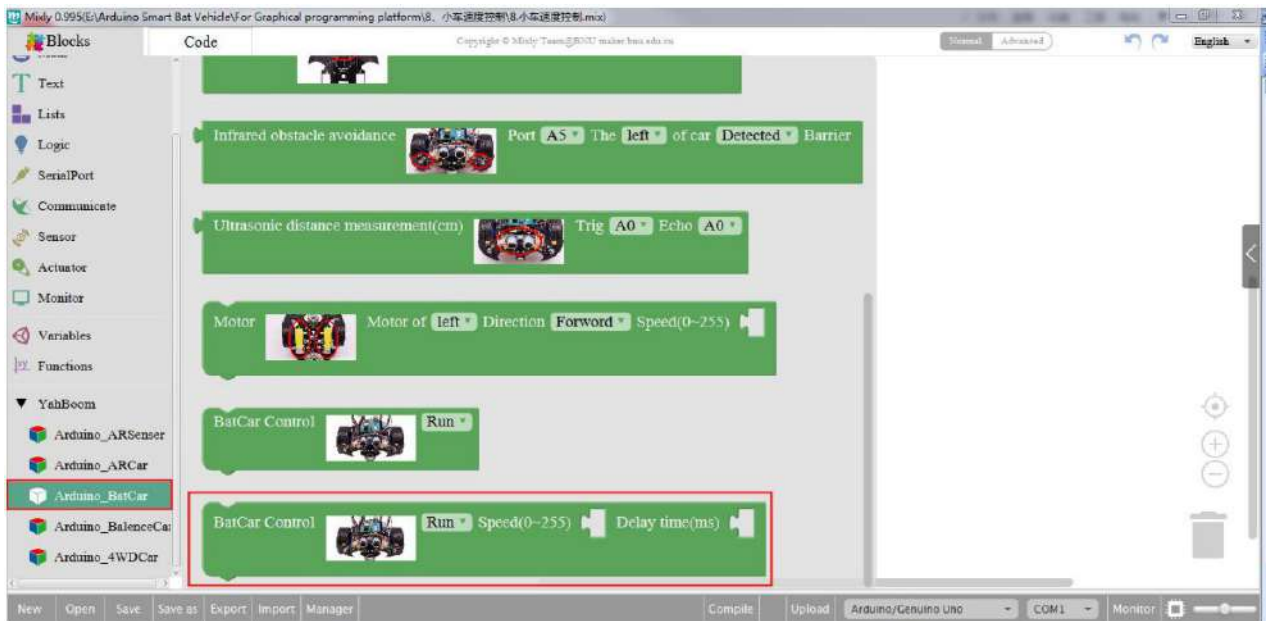
8. Control Speed

Follow this picture to connect the modules:



1.Modules of experiment:





2 .Experiment



(1) Complete the connection of modules and click the button “Compile”. You can see “Compile success” in the dialog window until compile completed. If you see “Compile failed” means that there is something wrong with the connection of modules. Check and fix it.



(2) Choose “Arduino/Genuino Uno” and correct serial port as shown.



(3) Now that configuration is complete, you can click the button “Upload”.



(4) Finally, It’s done if you see “Upload success” in the dialog window. Now unplug the USB line and power on the BatCar. The BatCar will go ahead at different speeds for 1 second.
 Expand Experiment: Can you design the BatCar run different directions at different speeds? Let’s try to do it.

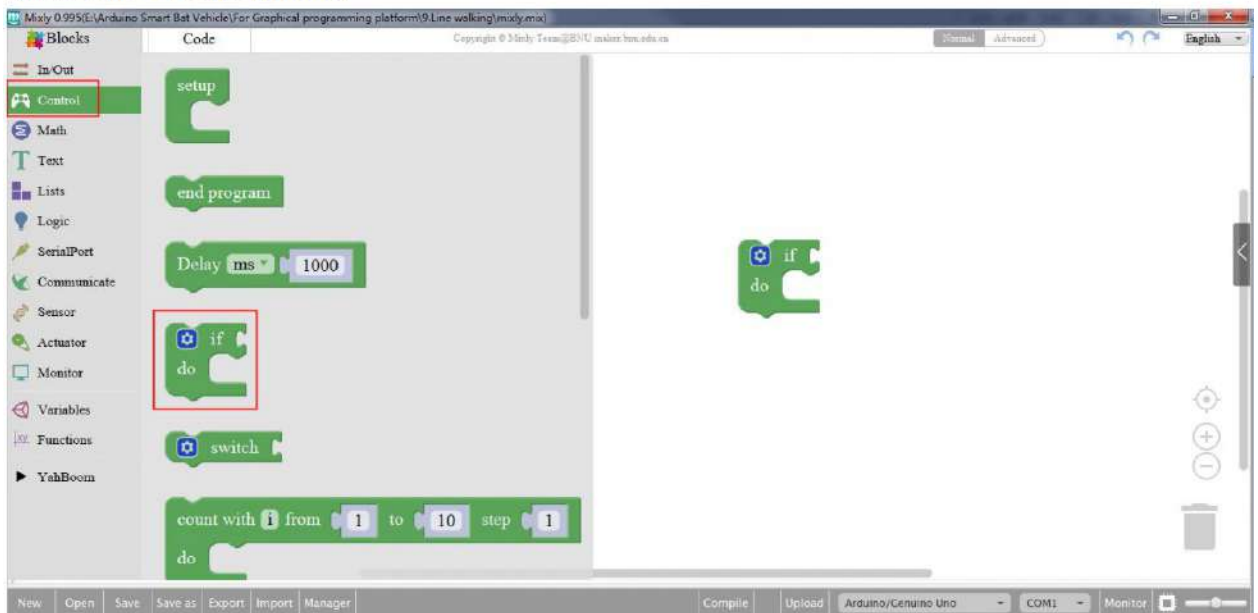
9- Line walking

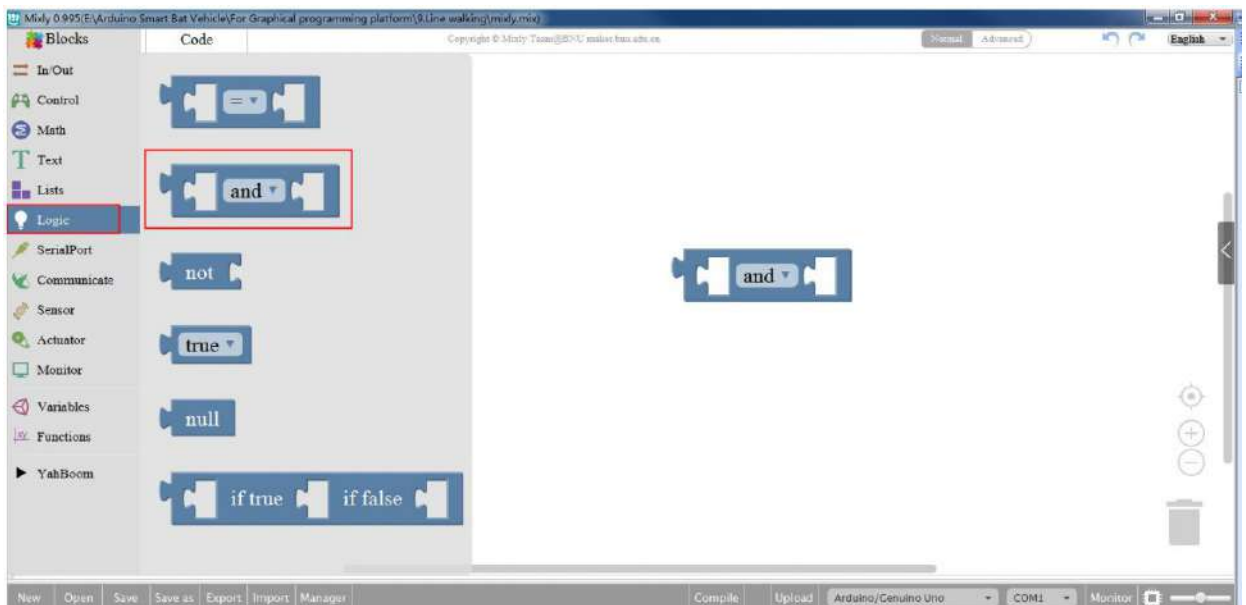
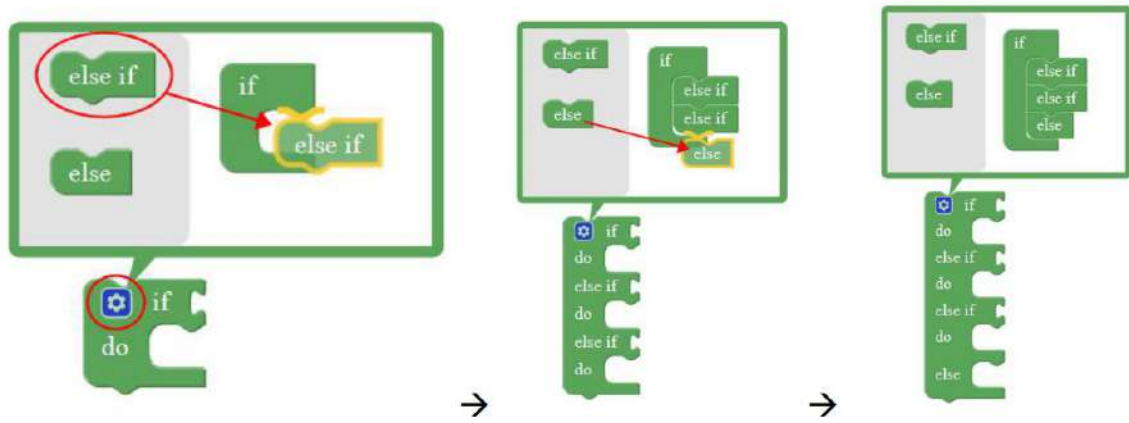
9.Line walking

Follow this picture to connect the modules:



1.Modules of experiment:





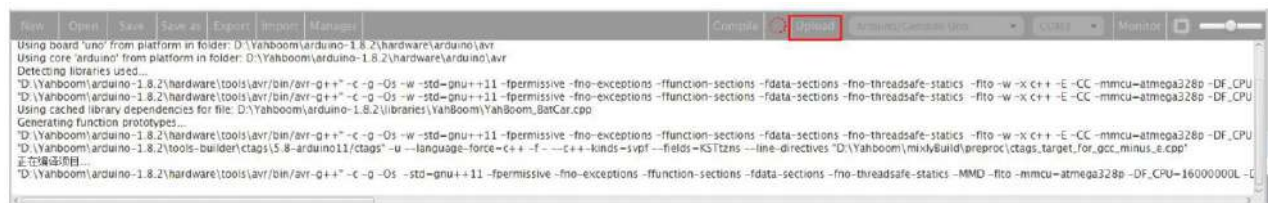
2 .Experiment



(1) Complete the connection of modules and click the button “Compile”. You can see “Compile success” in the dialog window until compile completed. If you see “Compile failed” means that there is something wrong with the connection of modules. Check and fix it.



(2) Choose “Arduino/Genuino Uno” and correct serial port as shown.



(3) Now that configuration is complete, you can click the button “Upload”.

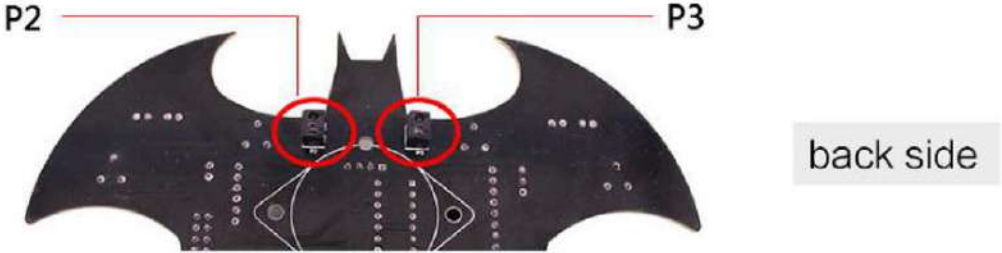
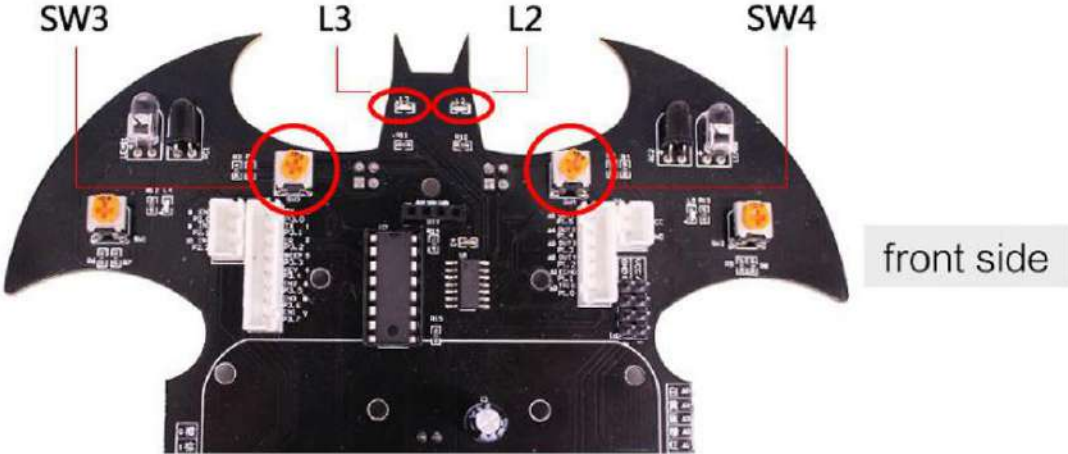


(4) Finally, It’s done if you see “Upload success” in the dialog window. Now unplug the USB line and debugging. After that, place the BatCar on track. Power on, it will go along the line.

Debugging:

- ① Adjust potentiometer [SW3] to make photoelectric sensor [P3] against white undersurface, then LED light [L3] illuminates while against black undersurface, LED light [L3] goes off.
- ② Adjust potentiometer [SW4] to make photoelectric sensor [P2] against white undersurface, then LED light [L2] illuminates while against black undersurface, LED light [L2] goes off.

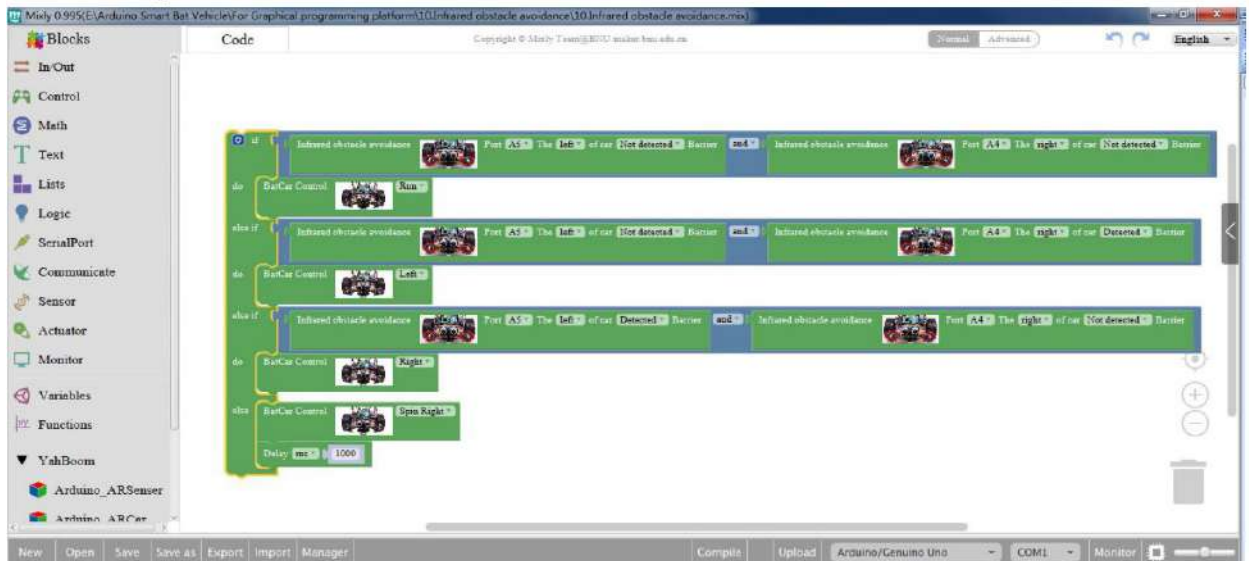
Caution : Don't excessively rotate potentiometer while adjusting. It should be within 30°.



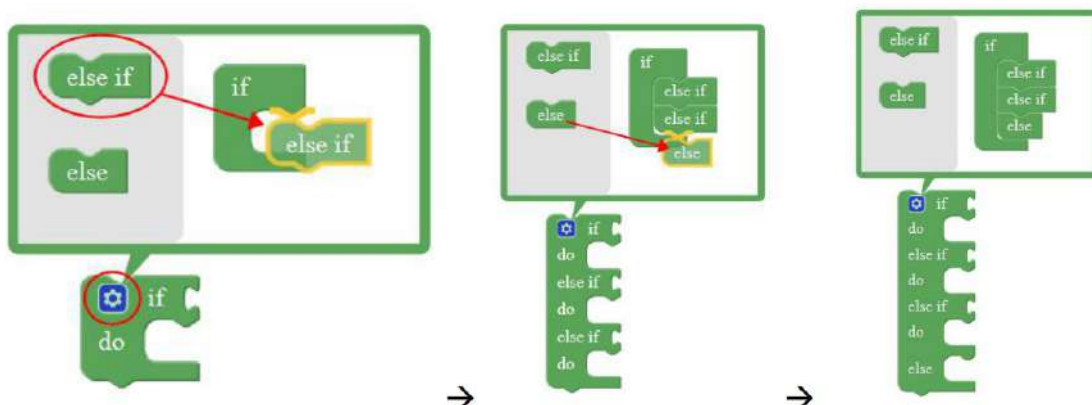
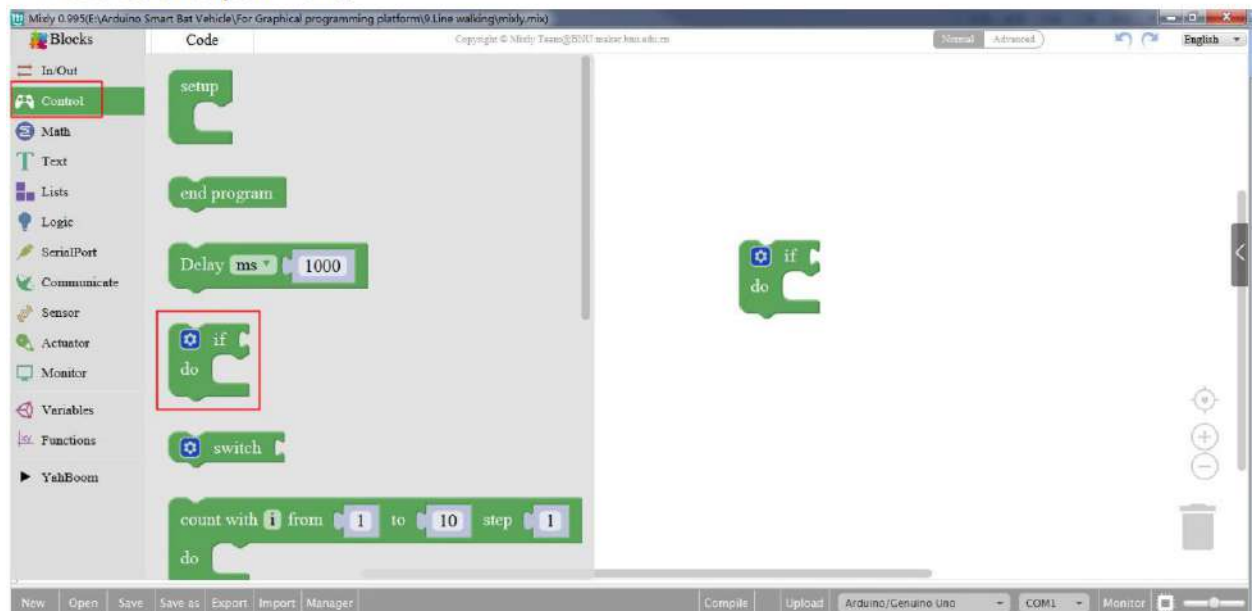
10- Infrared obstacle avoidance

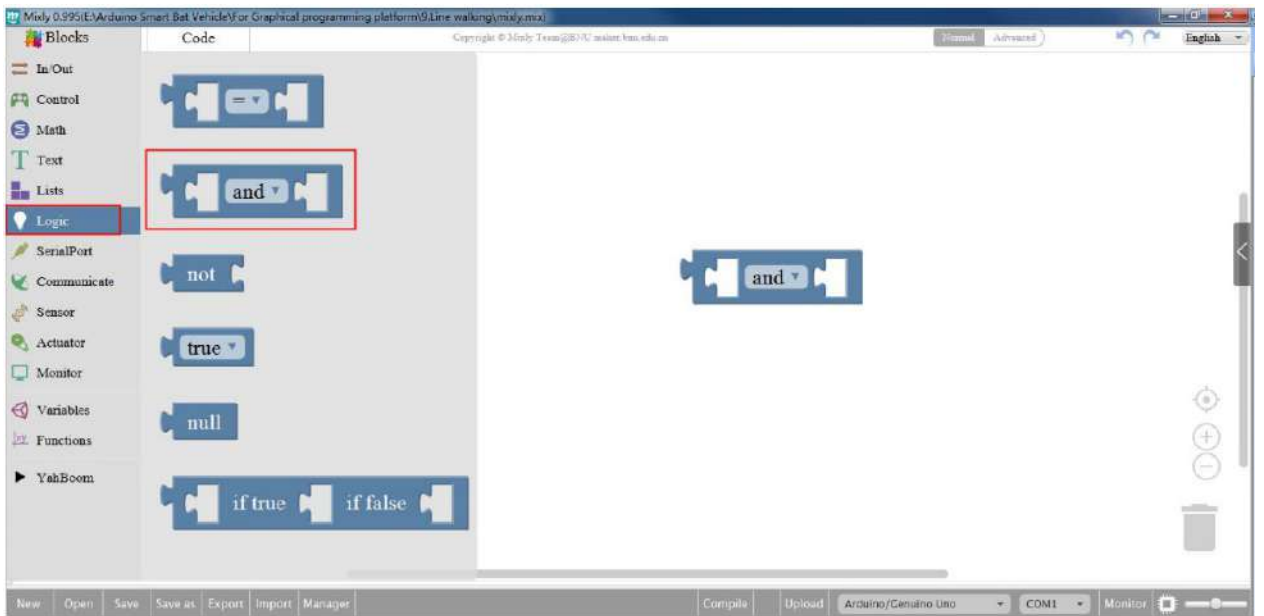
10.Infrared obstacle avoidance

Follow this picture to connect the modules:



1.Modules of experiment:

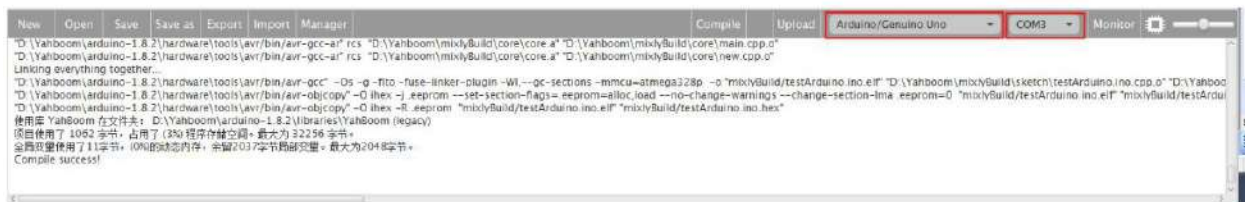




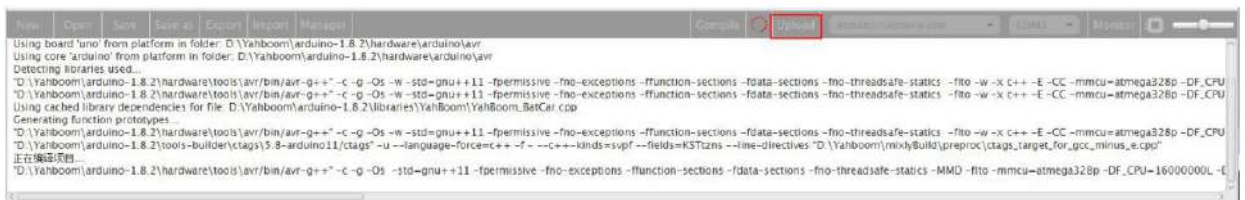
2. Experiment



(1) Complete the connection of modules and click the button “Compile”. You can see “Compile success” in the dialog window until compile completed. If you see “Compile failed” means that there is something wrong with the connection of modules. Check and fix it.



(2) Choose “Arduino/Genuino Uno” and correct serial port as shown.



(3) Now that configuration is complete, you can click the button “Upload”.



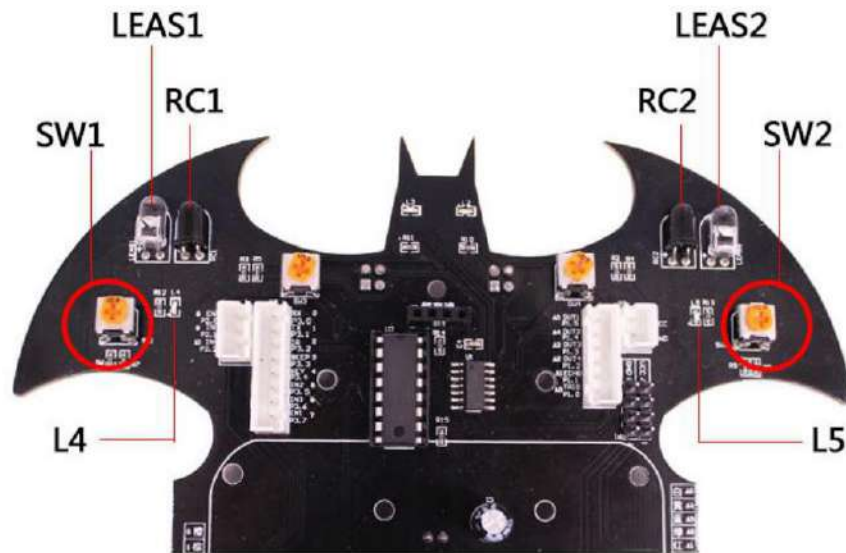
(4) Finally, It’s done if you see “Upload success” in the dialog window. Now unplug the USB line and debugging. After that, place the BatCar on the ground and power on, it will run and avoid the obstacle.

Debugging:

① Adjust potentiometer [SW1] to make the infrared light-emitting diode [LEAS1] and infrared light-receiving diode [RC1] away from obstacle less than 10 cm, then LED light [L4] illuminates, otherwise, it goes off.

② Adjust potentiometer [SW2] to make the infrared light-emitting diode [LEAS2] and infrared light-receiving diode [RC2] away from obstacle less than 10 cm, then LED light [L5] illuminates, otherwise, it goes off.

Caution : Don’t excessively rotate potentiometer while adjusting. It should be within 30°.

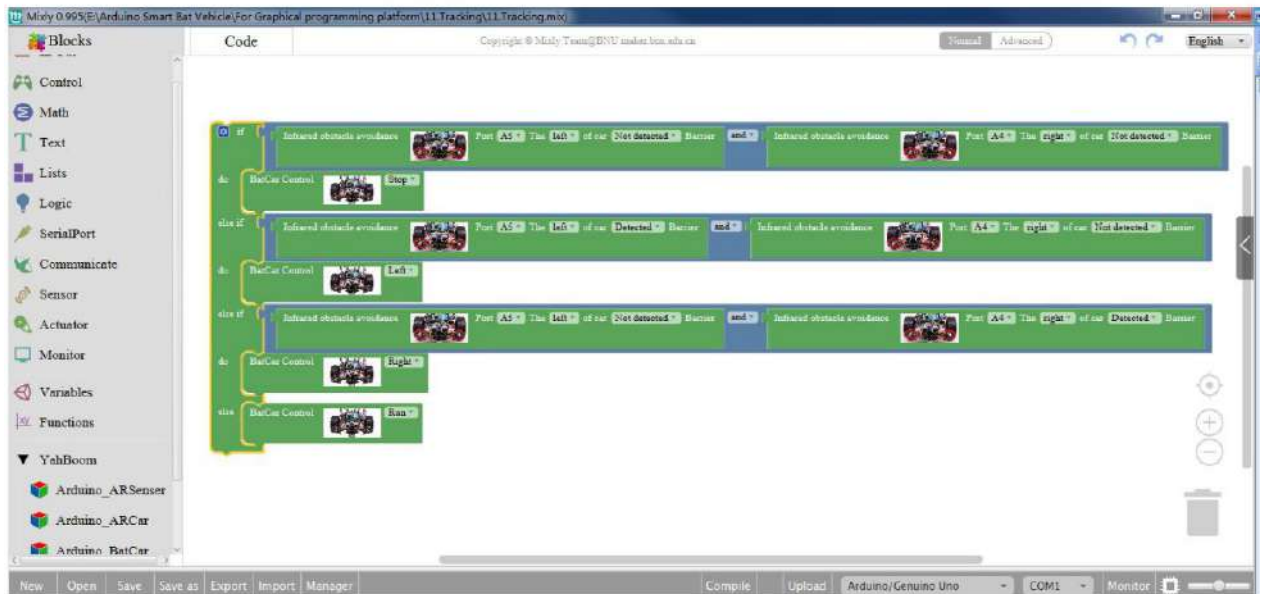


Expand Experiment: Can you change the logic of the program to make it tracking the obstacle instead of avoiding? Let's try to do it and we will tell you how to do it next class.

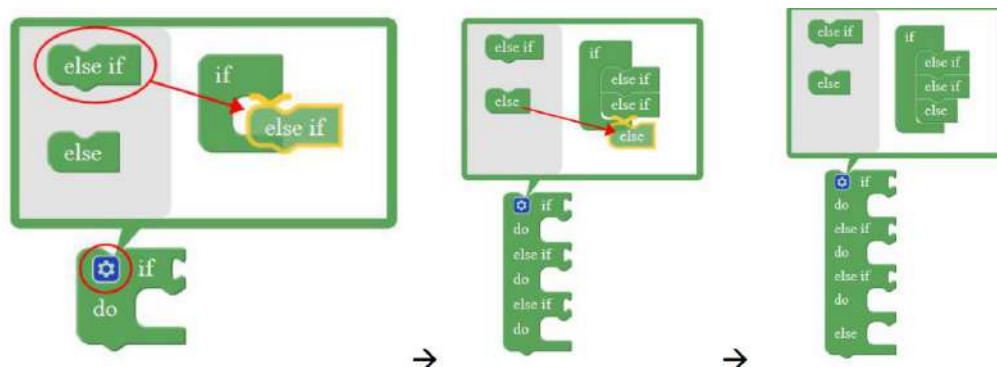
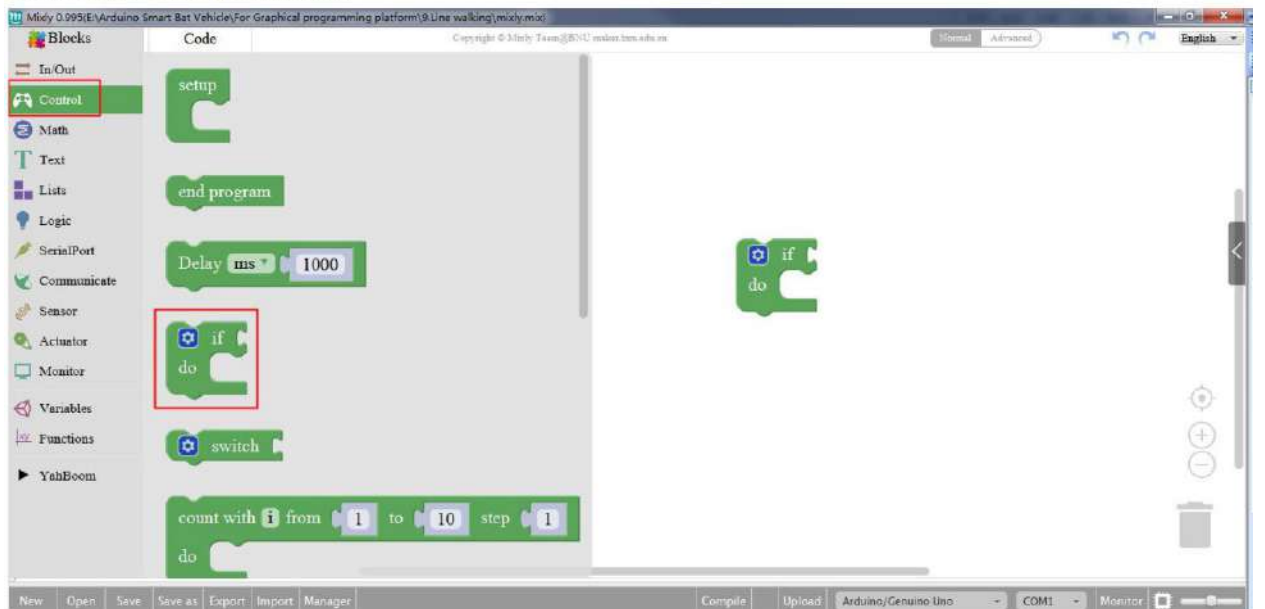
11- Tracking

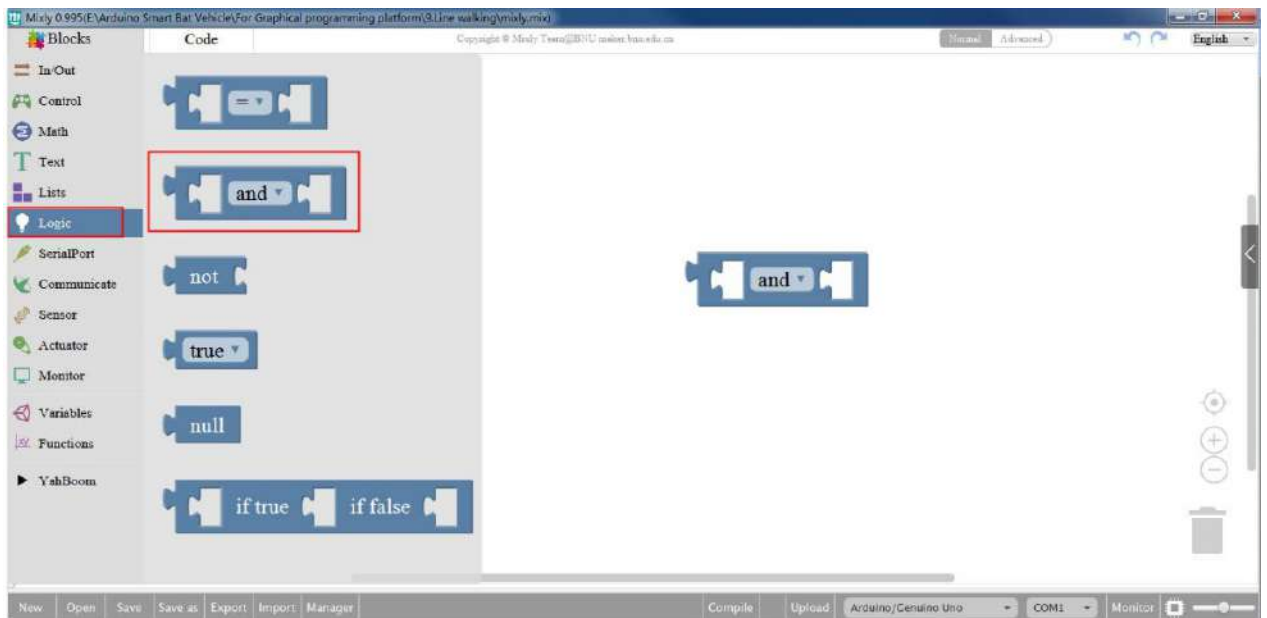
11.Tracking

Follow this picture to connect the modules:



1.Modules of experiment:





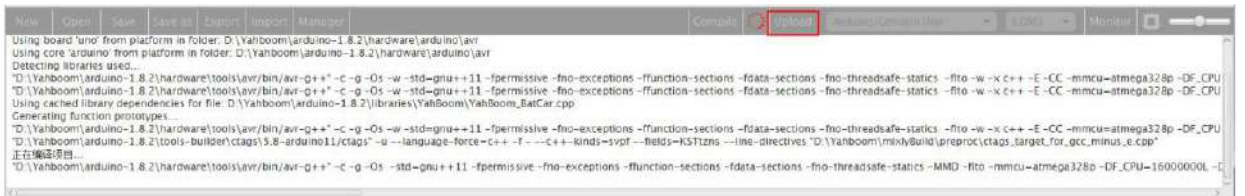
2 .Experiment



(1) Complete the connection of modules and click the button “Compile”. You can see “Compile success” in the dialog window until compile completed. If you see “Compile failed” means that there is something wrong with the connection of modules. Check and fix it.



(2) Choose “Arduino/Genuino Uno” and correct serial port as shown.



(3) Now that configuration is complete, you can click the button “Upload”.



(4) Finally, It’s done if you see “Upload success” in the dialog window. Now unplug the USB line and debugging. After that, place the BatCar on the ground and power on, it will follow up obstacle to drive.

Debugging:

- ① Adjust potentiometer [SW1] to make the infrared light-emitting diode [LEAS1] and infrared light-receiving diode [RC1] away from obstacle less than 10 cm, then LED light [L4] illuminates, otherwise, it goes off.
- ② Adjust potentiometer [SW2] to make the infrared light-emitting diode [LEAS2] and infrared light-receiving diode [RC2] away from obstacle less than 10 cm, then LED light [L5] illuminates, otherwise, it goes off.

Caution : Don't excessively rotate potentiometer while adjusting. It should be within 30°.

