THE HONG KONG UNIVERSITY OF SCIENCE AND TECHNOLOGY
Department of Electronic and Computer Engineering
ELEC 1100

Laboratory 6: Sensors & MCU (4%)

A) Objectives:

- By applying all techniques learned in the previous labs, design and complete the control circuit of the robot vehicle so that it can track a straight line and a split.

B) Equipment and necessary documents:

- Edroid Navidroid assembly kit with tools (with light sensors)
- Navidroid's Guide – Navidroid robot manual

C) Hardware and Software

- Nano-Board (Arduino Compatible) and open-source Arduino Software (IDE)

D) Experimental Procedures:

Experiment 1: Line detection using multi-sensors (~30 mins)

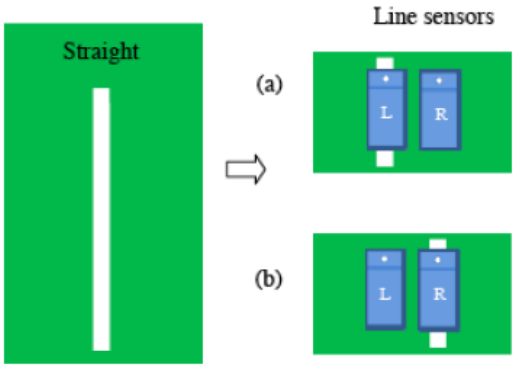
Step 1: Find in the project box, a line sensor with cable shown in Figure 1. Notice that VCC (red wire) should connect to 5V and GND (black wire) to ground (0V).

Warning: Do not mix up the wire connections.
Wrong connections will damage the line sensor.

Figure 1

The line sensor in the Navidroid kit consists of a IR emitter and a photo-diode. It works by illuminating a surface with the IR emitter, the photo-diode picks up the reflected IR and, based on its intensity, determines the reflectivity of the surface. This allows the sensor to detect a dark line on a pale surface, or a pale line on a dark surface. This is a digital sensor, meaning that its output voltage is either "high" (5V) or "low" (0V).

Step 2: Hook up **two** sensors on the Navidroid vehicle and have their photo-diodes facing downward. A straight white line on dark plane shall be detected by two line sensors for two different cases: (a) and (b). L and R represent the left and right sensors.



Case	L	R
(a)	Low	High
(b)	High	Low

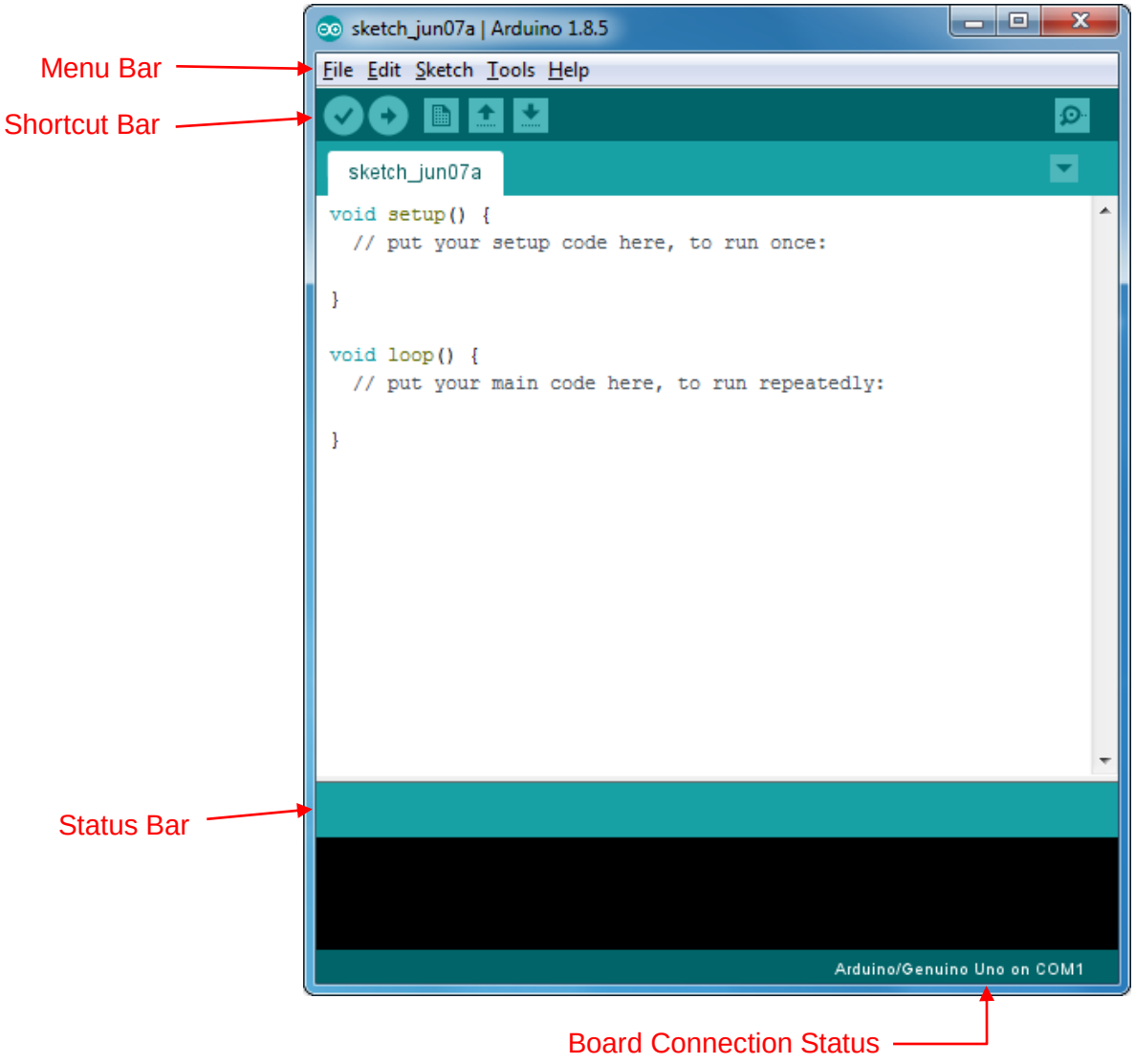
Step 3: The position of the two sensors is up to you. Have the sensors move across the white line and dark surface to see if each sensor can differentiate between them, i.e. a dark surface gives 5V, while a white line gives 0V. If not, do the followings:
a) adjust the sensor height from mat
b) adjust the sensitivity of sensor by tuning the variable resistor using screwdriver

Step 4: Make sure the sensors output voltage (OUT: yellow wire) obey the table above. Demo to your TA for his/her signature.

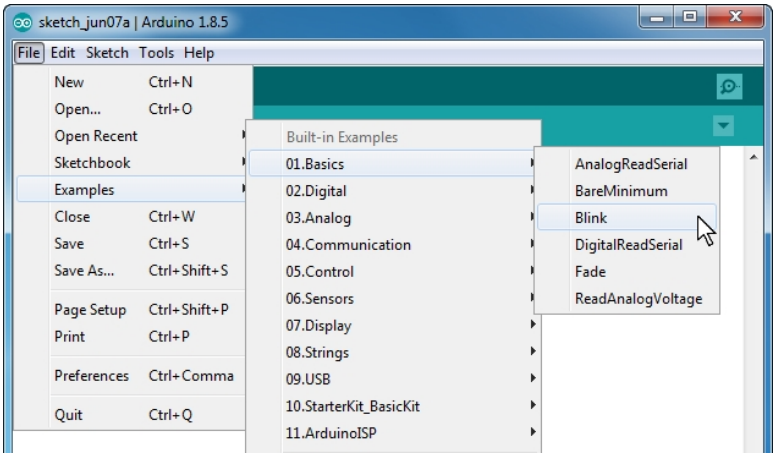
Experiment 2: Make your Nano-Board blinks (~30min)

In this experiment, you will start to use the Arduino Software (IDE) to write programs and upload them to your board.

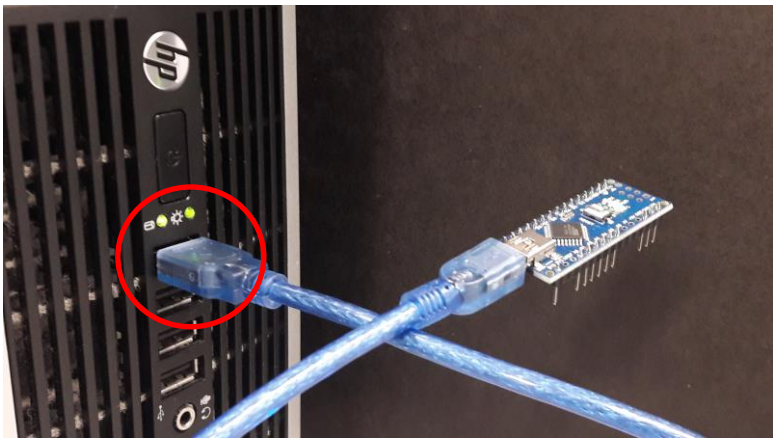
Step 1: Run the Arduino IDE on the desktop of your lab computer. If this is the first time you run the Arduino IDE, you should see a tab (called sketch) filled with the two basic Arduino functions: the *setup ()* and *loop()*.



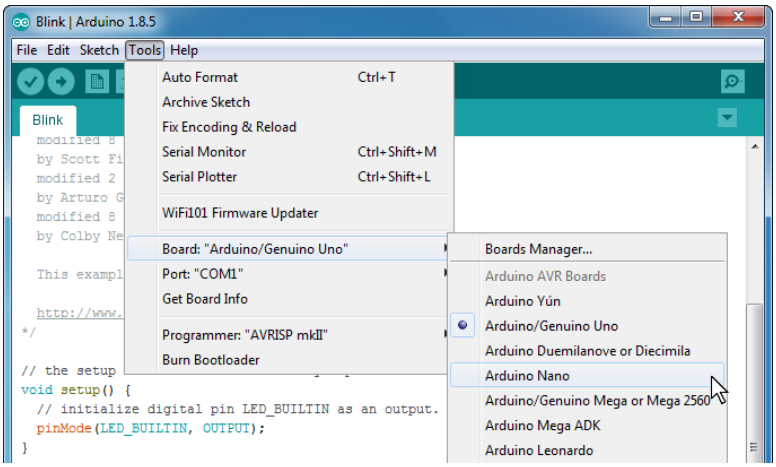
Step 2: Open the LED blink example sketch: **File > Examples > 01.Basics > Blink** .



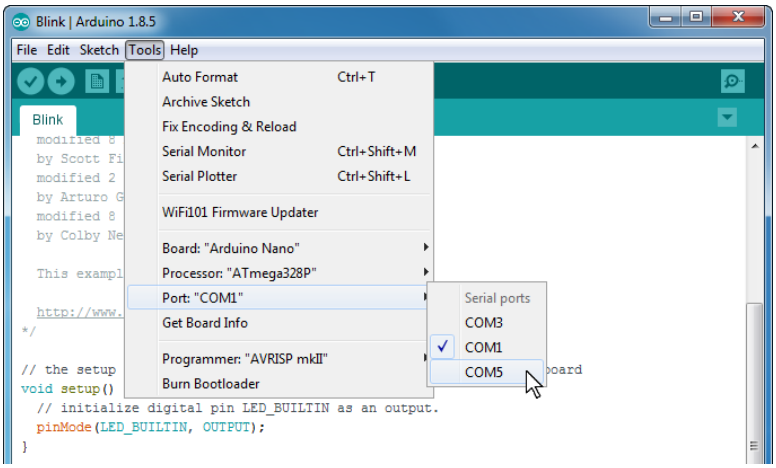
Step 3: Connect your Nano-Board to computer's USB port (use the top one as shown) via provided USB cable. This port has been pre-configured for using Nano-Board.



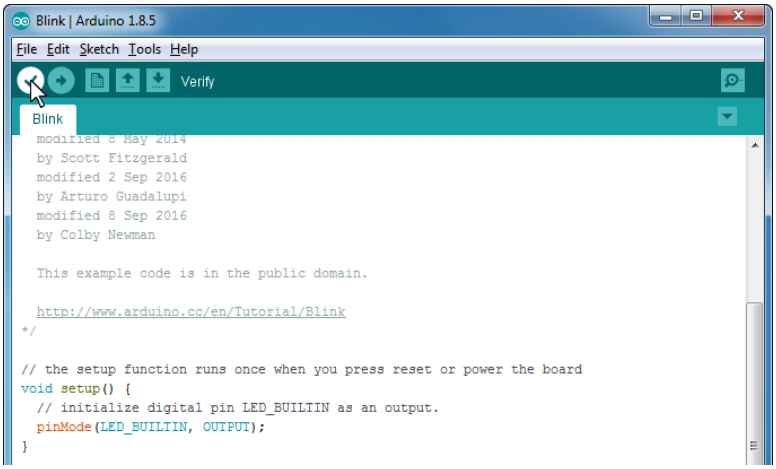
Step 4: Selected the board you have just connected: **Tools > Board > Arduino Nano**.



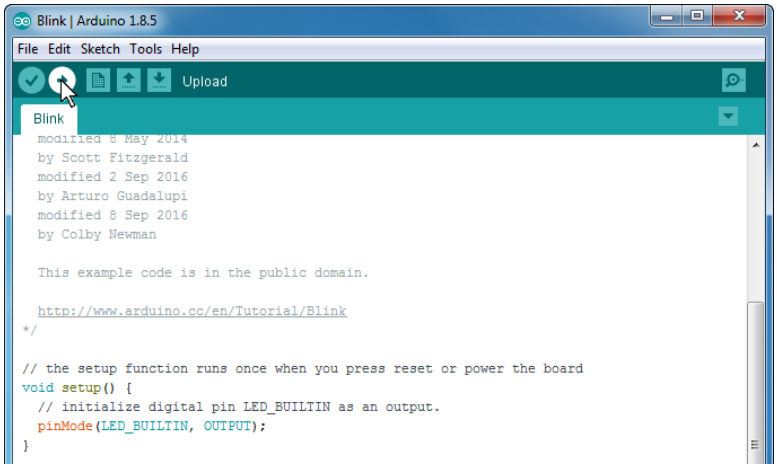
Step 5: Choose the uploading port: **Tools > Port > COM5**. Note that you may have **COM6** or **COM7** in your PC, depending on Windows configuration. *However, you cannot use COM1 and COM3 as Windows had already occupied them.* Please also note that it may take some times to show up the desired port to your Nano-Board after plug in.



Step 6: Compile your codes by clicking the **Verify** button. Your codes will be checked if there is any syntax error. If no error exists, the codes will be compiled.



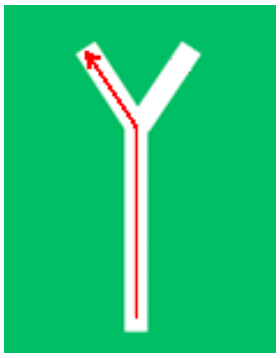
Step 7: Upload your code by clicking the **Upload** button. This action contains verification and upload. Wait a few seconds until the message "Done uploading." appears in the bottom status bar. You should then see on your board the red LED with an “L” next to it start blinking.



Step 8: Change the delay time to 100, either HIGH time or LOW time. Upload the Blink sketch again. You should see the time difference in blinking. Demo to your TA and get his/her signature.

Note: 1) You may write and verify your codes without plug in the Nano-Board.
2) As described in step 7, you may directly use "upload" after minor code changes. The program will do verification, compilation and upload in one click.

Experiment 3: Tracking a Straight Line and a Split (~1 hour)



In this experiment, you will need to do the logic control by using Arduino programming language for your vehicle, so that it can move along the track. It should be able to track a straight line and go through a split (turn left as shown).

Before the experiment, you are required to understand how the inputs **LQ** and **RQ** (to the comparator 74HC85, see Lab 5 manual, at page 2) affect the speed of the motor, and how the inputs **Ldir** and **Rdir** affect the rotational direction of the motors.

Step 1: Confirm the Ldir and Rdir signals from Lab 5.

Demo to your TA that:

- when Ldir & Rdir = High (connect to 5V), the two wheels let your car go **Forward**,
- when Ldir & Rdir = Low (connect to 0V), the two wheels let your car go **Backward**.

Step 2: Complete the truth table.

With two sensors (L and R) detecting the track, totally there are four possible combinations of the sensor output voltages as shown in the table (0 - Low Voltage, 1- High Voltage). For each case, think about what the car action should be and fill in the "Car Action" (e.g. turn left, go forward), and figure out what you should do to the motor and also fill in the "Rotation" columns for Left and Right motors (F for forward, B for backward, S for stop). You may, for example, increase motor's speed, reverse its rotation (going backward) or even stop a motor.

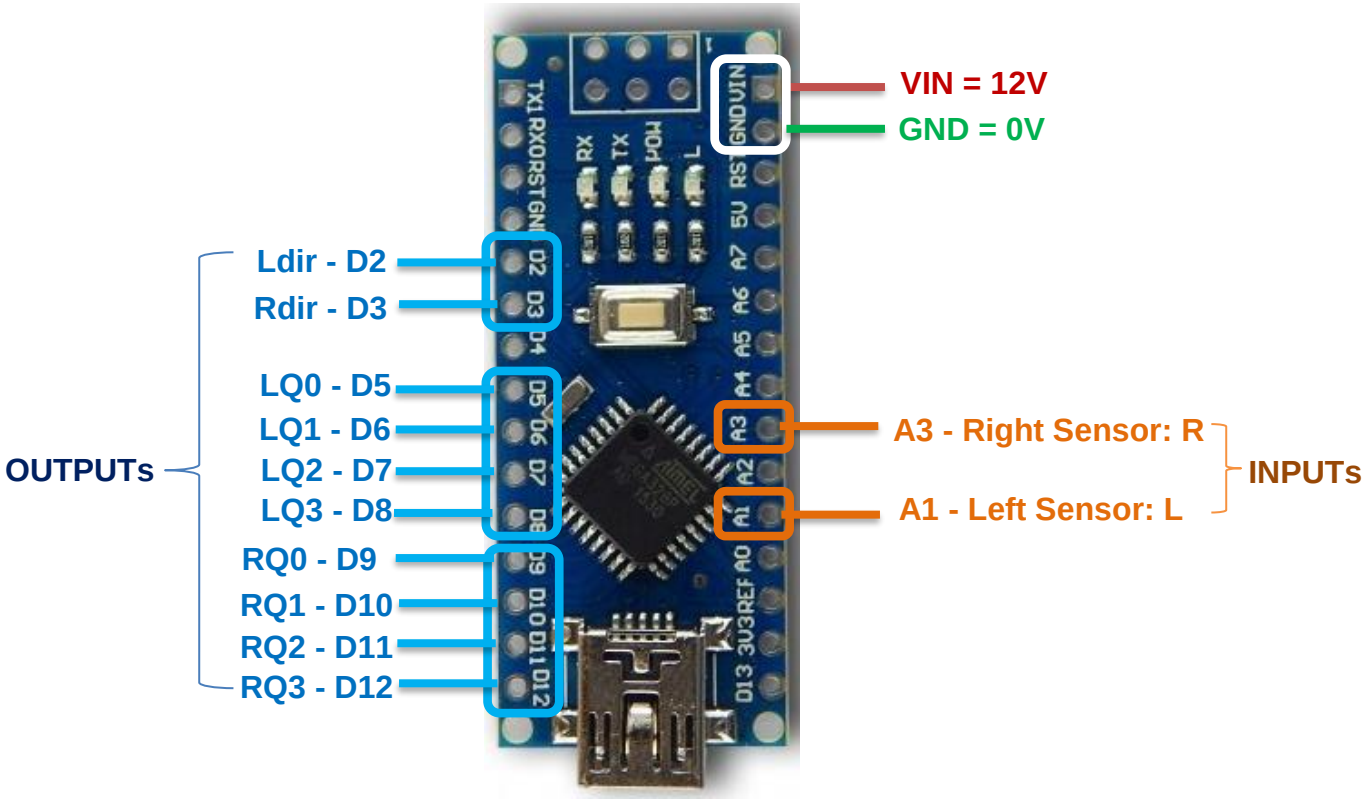
The truth table below is your logic control. Derive the required outputs **Ldir**, **LQ** (LQ3Q2Q1Q0), **Rdir**, and **RQ** (RQ3Q2Q1Q0) according to the sensor output signals **L** and **R**.

Do your own logic design and complete the truth table, making sure your motor car could achieve "go forward" and "turn left" at split.

Sensors		Car	Rotation		Left Motor (L)					Right Motor (R)				
L	R	Action	Left	Right	dir	Q3	Q2	Q1	Q0	dir	Q3	Q2	Q1	Q0
0	0	Turn Left	B	F										
0	1													
1	0													
1	1	Forward	F	F										

Fill in the entries in your summary sheet
(You may refer to Tutorial 8 slides)

Step 3: Connect your Nano-Board to the circuits from Lab#5 on your breadboard as shown below.



Step 4: Import Lab06 Arduino sketch program (you can download from Canvas → Files → Labs → Lab06_Logic) to your Desktop IDE. Now, based on the truth table you have designed, fill in the question mark "???" areas with either a "HIGH" or a "LOW" code.

Step 5: Upload your completed code to your Nano-Board. Demo to your TA that your vehicle can go through the track and turn left at the split.

```
  // initialize output pins.
  digitalWrite(pinLDir, HIGH);        // HIGH: forward rotate
  digitalWrite(pinRDir, HIGH);        // LOW:  reverse rotate
  digitalWrite(pinLQ0, ???);
  digitalWrite(pinLQ1, ???);
  digitalWrite(pinLQ2, ???);
  digitalWrite(pinLQ3, ???);
  digitalWrite(pinRQ0, ???);
  digitalWrite(pinRQ1, ???);
  digitalWrite(pinRQ2, ???);
  digitalWrite(pinRQ3, ???);
}
// the loop function runs over and over again forever
void loop() {

  leftSensor = digitalRead(pinLeftSensor);
  rightSensor = digitalRead(pinRightSensor);

  if ( leftSensor && rightSensor ) {
    digitalWrite(pinLDir, ???);
    digitalWrite(pinRDir, ???);
  }

  if ( !leftSensor && rightSensor ) {
    digitalWrite(pinLDir, ???);
    digitalWrite(pinRDir, ???);
  }

    if ( leftSensor && !rightSensor ) {
    digitalWrite(pinLDir, ???);
    digitalWrite(pinRDir, ???);
  }

  if ( !leftSensor && !rightSensor ) {
    digitalWrite(pinLDir, ???);
    digitalWrite(pinRDir, ???);
  }
```

You have learned all the basics, and you are now ready for your project!

ELEC 1100 Laboratory 6: Summary Sheet

Group Number: _____

Name: _____ Lab Partner: _____

Student ID: | | | | | | | | | Student ID: | | | | | | | | |

Experimental Part

Experiment 1: Line detection using multi-sensors

TA's signature: _____

Experiment 2: Make your Nano-Board blink

TA's signature: _____

Experiment 3: Logic Control for the Navidroid by Arduino Program

Demo to your TA that:

- when Ldir & Rdir = High (connect to 5V), the two wheels let your car go **Forward**,
- when Ldir & Rdir = Low (connect to 0V), the two wheels let your car go **Backward**.

TA's signature: _____

Do your own logic design and complete the truth table, making sure your motor car could achieve "go forward" and "turn left" at split.

Sensors		Car Action	Rotation		Left Motor (L)					Right Motor (R)				
L	R		Left	Right	dir	Q3	Q2	Q1	Q0	dir	Q3	Q2	Q1	Q0
0	0	Turn Left												
0	1													
1	0													
1	1	Forward	F	F										

Demo to your TA that your vehicle can go through the track and turn left at the split.

TA's signature: _____