**ENGG1100 Introduction to Engineering Design**

**Faculty of Engineering**

**The Chinese University of Hong Kong**

**Laboratory 6: Digital Logic 1**

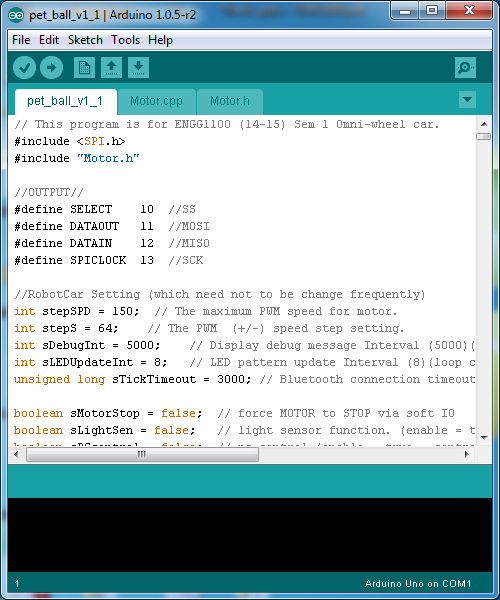
Week 7, 2014 Fall (6 Oct, 2014)

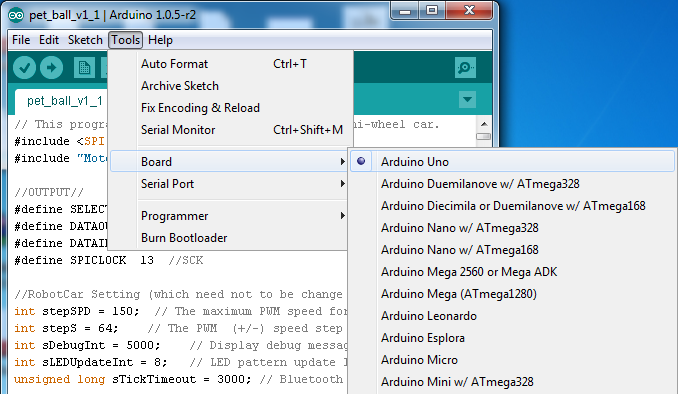
|  |
| --- |
| **Introduction**  Digital logic is the foundation for computers and will be used for controlling your intelligent robot in your course project. In this lab we will learn the basic logic functions and see how they can be implemented using a popular micro-controller board called **Arduino**.  In this lab exercise, signals are represented by two logical levels: ‘0’ or ‘1’. In particular TTL level is used, namely level ‘1’ is 5 Volts, and level ‘0’ is 0 Volt. Here, we will study the logic operations (AND, OR, NOT etc.) of these signals and learn how these functions can help you design your robot. |
| **Objectives**  By completing this lab session, you should:   1. Learn the basic concepts of digital logic 2. Learn how to use the Arduino board to implement digital logic functions |
| Recording data properly from a laboratory experiment is part of the learning process in this course. As such, you are required to complete a document called a **lab sheet**. Prepare your lab sheet **before** coming to the lab.  After completing the lab, submit your lab report sheet (with your names and date) to the tutor. |
| ***Read the lab manual thoroughly before attending the lab!*** |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ***Experiment 1.1 (Written Exercise): Fill in the truth table of the AND logic function. (No hardware experiment is required for this exercise)***  ***pic015***   |  |  |  | | --- | --- | --- | | Inputs | | Output | | A | B | Q | | 0 | 0 |  | | 0 | 1 |  | | 1 | 0 |  | | 1 | 1 |  |   AND logic  ***Experiment 1.2 (Written Exercise): Fill in the truth table of the OR logic function. (No hardware experiment is required for this exercise)***   |  |  |  | | --- | --- | --- | | Inputs | | Output | | A | B | Q | | 0 | 0 |  | | 0 | 1 |  | | 1 | 0 |  | | 1 | 1 |  |   pic016  OR logic  ***Experiment 1.3: Fill in the truth table of the AND logic function after you completed the procedures. (Hardware experiment is required for this exercise)***  ***LEDi can be “ON “ or “OFF; Outi can be ‘0’ or ‘1’***   |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | **Inputs** | | | | **Output Q** | | | S3 | S1 | LED3 | LED1 | LED7 | Out1 | | 0 | 0 | ON | ON |  |  | | 0 | 1 | ON | OFF |  |  | | 1 | 0 | OFF | ON |  |  | | 1 | 1 | OFF | OFF |  |  |   //program segment in the main loop of Lab6.ino  //for reference only, Din1() will read the logic status of S1, etc.  void loop()  { // Experiment 1.3 OUT1=S1 AND S3  if(Din1() && Din3()) Out1(1);  else Out1(0);  :  } |
| **Procedures:**   1. Connect the DC 9V power to the Smart-car board (no need to connect the Serial COM cable). Turn on the power supply and turn on the power switch on the board.   pic031**Figure 1**   1. A program inside the system is written to simulate the AND operation with input S1=A, S3=B and LED7=Q of the diagram shown in experiment 1.1. 2. Start to examine the different input combinations of S1 and S3 by pressing the corresponding push buttons as shown in the Figure 1, and record the output of Q (LED7 or out1)   **NOTICE:**  *When you release Si (i=1 or 2 or 3 or 4), it will be at logic level* ***‘1’*** *the corresponding LEDi is* ***off****.*  *When you press Si (i=1 or 2 or 3 or 4), it will be at logic level* ***‘0’*** *the corresponding LEDi is* ***on****.*  *When Out1(or Out2, Out3, Out4) is at logic level* ***‘0’*** *the corresponding LED7(or LED8, LED9, LED10) is* ***off****.*  *When Out1(or Out2, Out3, Out4) is at logic level* ***‘1’*** *the corresponding LED7(or LED8, LED9, LED10) is* ***on****.*   1. Complete the truth table above.   ***Experiment 1.4: Fill in the truth table of the OR logic function.***   |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | **Inputs** | | | | **Output Q** | | | S3 | S1 | LED3 | LED1 | LED9 | Out3 | | 0 | 0 | ON | ON |  |  | | 0 | 1 | ON | OFF |  |  | | 1 | 0 | OFF | ON |  |  | | 1 | 1 | OFF | OFF |  |  |     //Program segment in the main loop of Lab6.ino  //for reference only, Din1() will read the logic status of S1, etc.  void loop()  { :  // Experiment 1.4 OUT3=S1 OR S3  if(Din1() || Din3()) Out3(1);  else Out3(0);  :  }   1. Connect the DC 9V power to the Smart-car board (no need to connect the Serial COM cable). Turn on the power supply and turn on the power switch on the board. 2. A program inside the system is written to simulate the OR operation with input S1=A, S3=B and LED9=Q of the diagram shown in experiment 1.2. 3. Examine the results of different combinations of S1 and S3 by pressing the corresponding push buttons as shown in Figure 1, and record the output of Q (LED9 or out3). 4. Complete the truth table above.   **Question:**  Do the results of experiment 1.3 and1.4 agree with that of experiment 1.1 and 1.2? Explain briefly. |
| ***Experiment 2.1: Fill in the truth table of the following combinational logic function.***  //Program segment in the main loop of Lab6.ino  //for reference only, Din2() will read the logic status of S2, etc.  void loop()  { :  // Experiment 2.1 OUT2=(S2 AND S3) AND S4  if((Din2() && Din3()) && Din4()) Out2(1);  else Out2(0);  :  pic032}  **Out2 = (S2 AND S3) AND S4**   1. Connect the DC 9V power to the Smart-car board (no need to connect the Serial COM cable). Turn on the power supply and turn on the power switch on the board. 2. Examine the results for different combinations of S2, S3 and S4 by pressing the corresponding push buttons as shown in Figure 1, and record the output of Out2 (LED8). 3. Complete the following truth table.  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | | **Inputs** | | | | | | **Output** | | | S4 | S3 | S2 | LED4 | LED3 | LED2 | LED8 | Out2 | | 0 | 0 | 0 | ON | ON | ON |  |  | | 0 | 0 | 1 | ON | ON | OFF |  |  | | 0 | 1 | 0 | ON | OFF | ON |  |  | | 0 | 1 | 1 | ON | OFF | OFF |  |  | | 1 | 0 | 0 | OFF | ON | ON |  |  | | 1 | 0 | 1 | OFF | ON | OFF |  |  | | 1 | 1 | 0 | OFF | OFF | ON |  |  | | 1 | 1 | 1 | OFF | OFF | OFF |  |  |   ***Experiment 2.2: Fill in the truth table of the following complex logic function.***  //Program segment in the main loop of Lab6.ino  //for reference only, Din2() will read the logic status of S2, etc.  void loop()  {:  // Experiment 2.2 OUT4=(S2 AND S3) OR S4  if((Din2() && Din3()) || Din4()) Out4(1);  else Out4(0);  }  pic033  **Out4 = (S2 AND S3) OR S4**   1. Connect the DC 9V power to the Smart-car board (no need to connect the Serial COM cable). Turn on the power supply and turn on the power switch on the board. 2. Examine the results of different combinations of S2, S3 and S4 by pressing the corresponding push buttons as shown in Figure 1, and record the output of Out4 (LED10). 3. Complete the following truth table.  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | | **Inputs** | | | | | | **Output** | | | S4 | S3 | S2 | LED4 | LED3 | LED2 | LED10 | Out4 | | 0 | 0 | 0 | ON | ON | ON |  |  | | 0 | 0 | 1 | ON | ON | OFF |  |  | | 0 | 1 | 0 | ON | OFF | ON |  |  | | 0 | 1 | 1 | ON | OFF | OFF |  |  | | 1 | 0 | 0 | OFF | ON | ON |  |  | | 1 | 0 | 1 | OFF | ON | OFF |  |  | | 1 | 1 | 0 | OFF | OFF | ON |  |  | | 1 | 1 | 1 | OFF | OFF | OFF |  |  |   ***Experiment 3.1: Learn how to use the Smart Car system board to control the motor***  Connect a direct current D.C. motor to the left motor output connector JL as shown in the figure below.  pic042  CLOCKWISE  Observe the relation between the logic inputs, logic outputs and the motor operation. Then fill in the table and answer the questions.  **NOTICE:**  *When you release Si (i=1 or 2 or 3 or 4), it will be at logic level* ***‘1’*** *the corresponding LEDi is* ***off****.*  *When you press Si (i=1 or 2 or 3 or 4), it will be at logic level* ***‘0’*** *the corresponding LEDi is* ***on****.*  *When Out1(or Out2, Out3, Out4) is at logic level* ***‘0’*** *the corresponding LED7(or LED8, LED9, LED10) is* ***off****.*  *When Out1(or Out2, Out3, Out4) is at logic level* ***‘1’*** *the corresponding LED7(or LED8, LED9, LED10) is* ***on****.*  **Procedures:**   1. Press SW1 and press SW2 at the same time; that means logic input 1 and logic input 2 are both set to “0” (LED1=ON and LED2=ON). Record the logic output 1 and logic output 2 in the first line of the table in the lab report sheet by observing the LED7 and LED8, if the LED is on, that means the logic output is “1” otherwise the logic output is “0”. Record the motor operation whether the motor is moving CLOCKWISE, ANTI-CLOCKWISE or STOP. 2. Press SW2 only; that means logic input 1 is “1” and logic input 2 is “0” (LED1=OFF and LED2 =ON). Record the logic output 1 and logic output 2 in the 2nd line of the table by observing the LED7 and LED8. If the LED is on, that means the logic output is “1”, otherwise the logic output is “0”. Record the motor operation whether the motor is moving CLOCKWISE, ANTI-CLOCKWISE or STOP. (We assume that you are looking at the same side as the two metal contacts of the motor for power.) 3. Press SW1 only, that means logic input 1 is “0” and logic input 2 is “1” (LED1=ON and LED2 =OFF). Record the logic output 1 and logic output 2 in the 3rd line of the table by observing the LED7 and LED8, if the LED is on, that means the logic output is “1” otherwise the logic output is “0”. Record the motor operation whether the motor is moving CLOCKWISE, ANTI-CLOCKWISE or STOP. 4. Without pressing SW1 and SW2; that means logic input 1 is “1” and logic input 2 is also “1” (LED1=OFF and LED2 =OFF). Record the logic output 1 and logic output 2 in the 4th line of the table by observing the LED7 and LED8. If the LED is on, that means the logic output is “1” otherwise the logic output is “0”. Record the motor operation to see whether the motor is moving CLOCKWISE, ANTI-CLOCKWISE or STOP. 5. Complete the truth table shown below. 6. Can you conclude, from the above experiment, how the motor operation depends on the logic output 1 and logic output 2?  |  |  |  |  |  | | --- | --- | --- | --- | --- | | Logic Inputs | | Logic Outputs | | Motor Operation | | SW2 | SW1 | LED7 | LED8 | | 0 | 0 |  |  |  | | 0 | 1 |  |  |  | | 1 | 0 |  |  |  | | 1 | 1 |  |  |  |   ***Experiment 3.2: Learn how to use the magnetic sensor input to control the motor***  Connect a magnetic sensor to the S1 connector as shown in the figure below.  pic043  Observe the relation between the magnetic sensor status, logic inputs, logic outputs and the motor operation. Then fill in the table and answer the questions.  **Procedures:**   1. Keep the magnetic sensor away from the magnetic strip as shown in the figure. Record the SW1 and SW2 states by observing LED1 and LED2 (keep in mind that input LED ON means the logic state is “0”, otherwise it means “1”). Record the logic output 1 and logic output 2 in the first line of the table by observing the LED7 and LED8, if the LED is on, that means the logic output is “1” otherwise the logic output is “0”. Record also the motor operation whether the motor is moving CLOCKWISE, ANTI-CLOCKWISE or STOP. 2. Put the magnetic sensor over the magnetic strip. Record the SW1 and SW2 states by observing LED1 and LED2 (keep in mind that input LED ON means the logic state is “0”, otherwise it means “1”). Record the logic output 1 and logic output 2 in the first line of the table by observing the LED7 and LED8. If the LED is on, that means the logic output is “1” otherwise the logic output is “0”. Record also the motor operation to see whether the motor is moving CLOCKWISE, ANTI-CLOCKWISE or STOP. 3. Fill in the truth table.  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | Magnetic sensor status | Logic inputs | | Logic outputs | | Motor operation | | SW2 | SW1 | LED7 | LED8 | | Away from magnetic strip | 1 |  |  |  |  | | Over the magnetic strip | 1 |  |  |  |  |   **Question:**  Can you conclude, from the above experiment, what is the relation between the magnetic sensor connected to S1 and the switch SW1?  ***Experiment 3.3: Learn how to use the light sensor as an input to control the motor***  Connect a Light sensor (Light Dependent Resistor LDR) to the S1 connector as shown in the figure below. It is known that the resistance of an LDR will be reduced when light is shined on it.  D:\Work\DrEdenMa\2014_15_ENGG1100\Lab6\exp3.3.jpg  Light Dependent Resistor LDR  Observe the relation between the light sensor status, logic inputs, logic outputs and the motor operation. Then fill in the table and answer the question.  **Procedures:**   1. Put the light sensor under room light as shown in the figure. Record the SW1 and SW2 states by observing LED1 and LED2 (keep in mind that input LED ON means the logic state is “0”, otherwise it means “1”). Record the logic output 1 and logic output 2 in the first line of the table by observing the LED7 and LED8, if the LED is on, that means the logic output is “1” otherwise the logic output is “0”. Record also the motor operation whether the motor is moving CLOCKWISE, ANTI-CLOCKWISE or STOP. 2. Cover the light sensor from any light source as shown in the figure. Record the SW1 and SW2 states by observing LED1 and LED2 (keep in mind that input LED ON means the logic state is “0”, otherwise it means “1”). Record the logic output 1 and logic output 2 in the first line of the table by observing the LED7 and LED8. If the LED is on, that means the logic output is “1” otherwise the logic output is “0”. Record also the motor operation to see whether the motor is moving CLOCKWISE, ANTI-CLOCKWISE or STOP. 3. Fill in the following table.  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | Light sensor status | Logic inputs | | Logic outputs | | Motor operation | | SW2 | SW1 | LED7 | LED8 | | Under room light | 1 |  |  |  |  | | Completely dark | 1 |  |  |  |  |   **Question:**  Can you conclude, from the above experiment, what is the relation between the light sensor connected to S1 and the switch SW1?  ***Conclusions and discussions for experiment 1 to 3***  Write about 100 words on what you have learned from this laboratory exercise.    In experiment 4, you are going to complete your PCB with tests in-between the procedure. Specific checkpoints will be introduced in the following way:  (**Check Point:** Is this working?)  **Please make sure** your board is working correctly **before** proceeding to further steps. Start troubleshooting if you cannot see what is expected at the checkpoints. We recommend you seek help from tutors/technicians during troubleshooting.  ***Experiment 4.1: Finishing and testing power circuit***  **Procedures:**   1. Solder the three GND points and join them together at J15. 2. Solder the switching regulator module onto U3. 3. Apply 9V to BAT1, **Jumper on BAT2 & BAT3** as in Lab 5 and turn the switch SW1 on. (**Check Point:** LED D1 should be lit.) 4. Adjust the VR (variable resistor) on U3 and monitor the output at J6 (7V test point) with an oscilloscope.   (**Check Point:** Make sure the output voltage is 7V and without ripple. A large ripple means that this switching regulator has problem. Ask technician to get a new one.)   1. **Disconnect power.** 2. Solder U11 (7805), then re-connect power.   (**Check Point:** Check the 7805’s output (pin-3, using pin-2 as ground) is 5V.)   1. **Disconnect power.** 2. Make sure that the capacitor and resistors around U4, U5 (both CPUs), U6 (74HC595) **and two 16.0MHz crystal (Y1, Y2)** are soldered. They should have been soldered in Lab 4. 3. Solder SW2 (RESET key), LED-bar (D4) and R35 (resistor network 1K\*10) and J8 (socket for Bluetooth device). 4. Solder 8 LEDs, D11-D18 which are placed round the board. 5. Re-connect power. Use a DMM or oscilloscope to check both CPU socket’s pin-7.   (**Check Point:** Check both CPU socket’s pin-7 is 5V.)  ***Experiment 4.2: Download Program with Wires***  **Procedures:**   1. Insert IC chips: U4 (**ATMega328P** with the word “**DOWNLOADER**”), U5 (**ATMega328P**), U6 (74HC595). 2. Connect Arduino downloader to J8 (the socket for Bluetooth):  |  | | --- | | DEVICE pin +5V ----------- BOARD pin +5V  DEVICE pin GND ----------- BOARD pin GND  DEVICE pin TXD ----------- BOARD pin RXD  DEVICE pin RXD ----------- BOARD pin TXD |  1. Download the program *Lab\_6.ino* to the board (refer to appendix A if necessary). Remember press the RESET key when **“Uploading…”** message shows up.   (**Check point**: LED D11-D18 will flash.)   1. Disconnect the downloader from J8.   ***Experiment 4.3: Download Program with Bluetooth***  **Procedures:**   1. Take out the Bluetooth modules. Observe the sticker label. The one with “M” is the MASTER and the one with the “S” is the SLAVE. 2. Insert the Bluetooth device (SLAVE) to J8 with sticker facing the board. 3. Connect the Bluetooth device (MASTER) to the downloader:  |  | | --- | | DEVICE pin +5V ----------- BLUETOOTH pin +5V  DEVICE pin GND ----------- BLUETOOTH pin GND  DEVICE pin RXD ----------- BLUETOOTH pin TXD  DEVICE pin TXD ----------- BLUETOOTH pin RXD | | BLUETOOTH pin +5V ----------- BOARD pin +5V  BLUETOOTH pin GND ----------- BOARD pin GND  BLUETOOTH pin TXD ----------- BOARD pin RXD  BLUETOOTH pin RXD ----------- BOARD pin TXD |  1. Download the program again; no need to press RESET key this time. LED D4.1 and D4.2 should flash once during programming.   (**Check point:** LED D4.1 and D4.2 do flash once during programming.) |

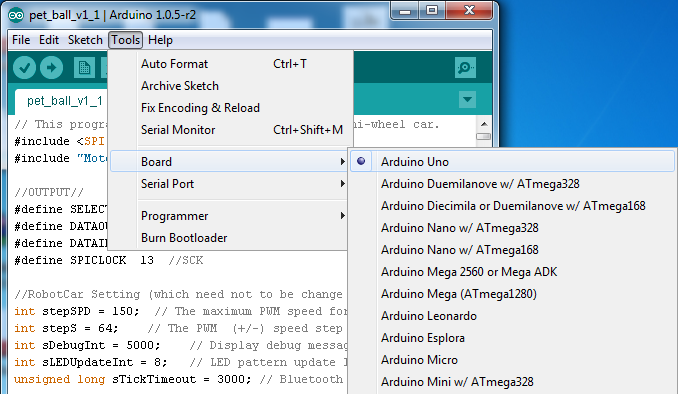
**Appendix A: Programming the Project board**

**Procedures:**

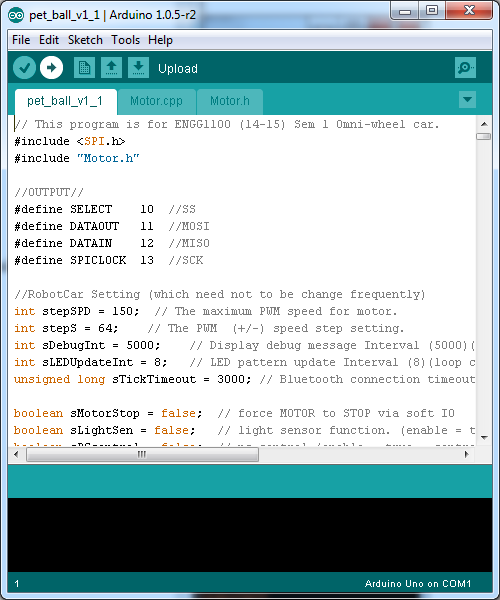
1. Downloading the Arduino testing program from the ENGG1100 home page.
2. Open the program by double-click the .*ino* file. It will be opened in Arduino programming environment.
3. Make sure the *Tools -> Board* is set to Arduino Uno.



1. Make sure the *Tools -> Serial Port* is set to correct COM port. (The COM port number for Arduino downloader can be checked from the **Device Manager**.)

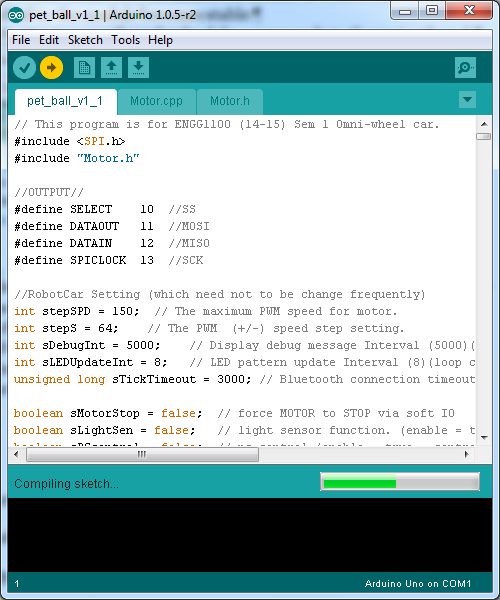


1. Turn on the project board. Make sure the Bluetooth module is connected to the project board. Wait until the red LED in the module becomes stable.
2. Press the **Upload** button to compile and upload the source code to the project board.

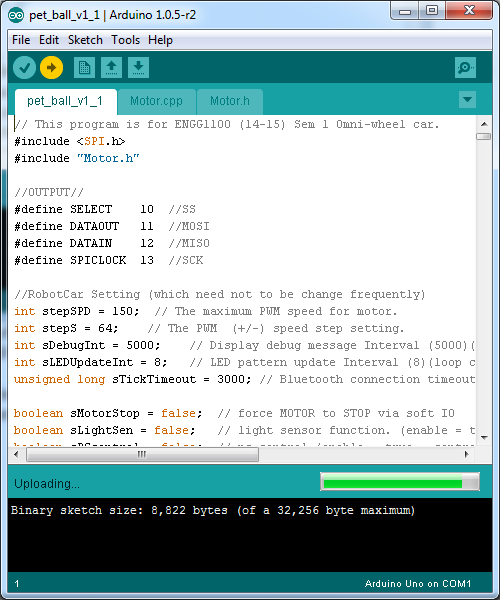


Upload Button

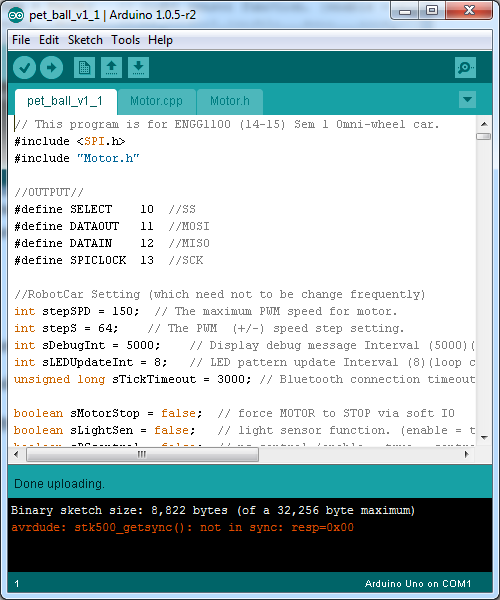
1. It starts to compile the source code and the message **“Compiling sketch…”** should be shown on the screen.



1. When the system finishes compiling, it will start to upload the program to the board automatically and the message “**Uploading…**” will be shown.



1. When uploading is done, the following screen will be displayed.



1. You may run your program by pressing the RESET button.