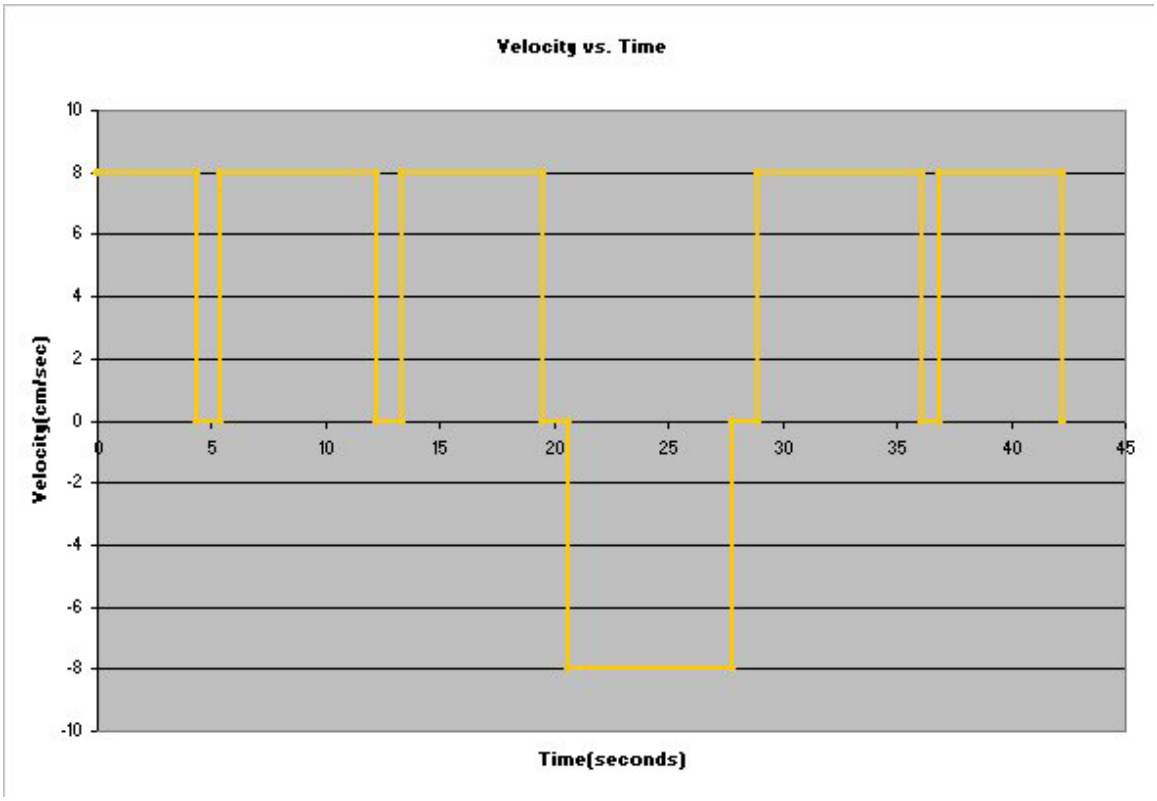
# Scientific Data Collection and Analysis Lab, Module 2a

Overview:

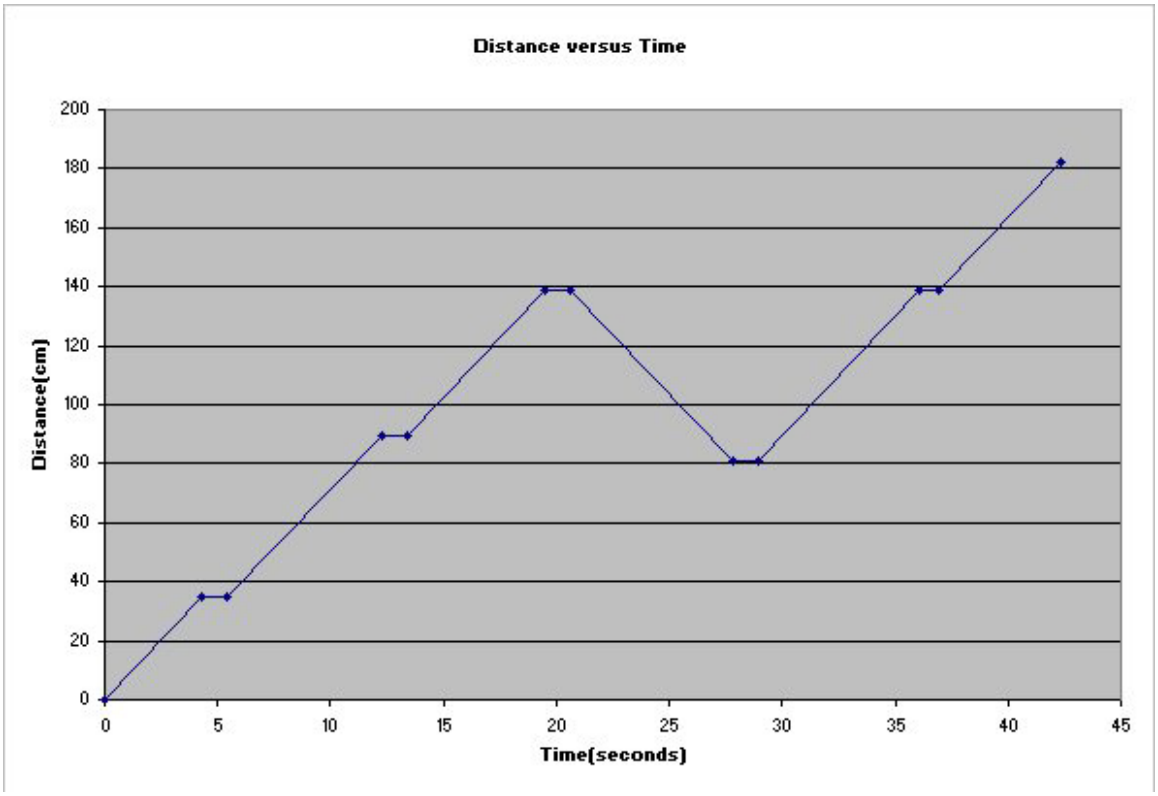
In this lab students will learn about scientific data collection and analysis. Students will program their NewCDbot robot to collect a specified number of data points, usually more than 10 but less than 50 data points is sufficient. The NewCDbot will be collecting data with regards to its changes in body velocity and the times at which these changes occur. Once a robot has finished collecting the data it will stop and students can then proceed to download the collected data to a desktop computer or laptop depending upon which is available. At this time students will begin creating spreadsheets with the velocity and time data. From the spreadsheets students can then create a graph of velocity versus time. The final graph to be created by the students will be that of distance versus time. In this lab students will gain an understanding of scientific data collection and analysis. More specifically, students will learn how robots can be used in science to help scientists collect data in environments where the scientists may not be able to go. An understanding of the steps required for creating spreadsheets and graphs along with the ability to read these graphs would be another skill acquired through this lab. This lab will require approximately 50 minutes to complete. If the NewCDbot robots have not previously been assembled then the lab would require an additional 50 minutes to complete.

Lab Setup and Preparation:

It will be assumed that the students have been properly introduced to the NewCDbot robot and that they have already completed assembly of their robot. If not then please refer to Module 1. At least one desktop computer or laptop must have the necessary software installed: the NewCDbot Programmer Software (NCPS) and OOPic Multi-Language Compiler. If more than one computer is available then it would be beneficial to install the required software on each available computer as this will help to reduce bottlenecks in the computer time when the students are programming their NewCDbot, downloading data, or creating their spreadsheets and graphs. The computers to be used need to have programs available for creating spreadsheets and graphs. Each of the NewCDbots should have an additional EEPROM chip in its E1 slot on the OOPic micro-controller and there should be at least one data-downloader cable available. A small area in the classroom should be cleared so that the students will have a place to let their robots roam around for the data collection phase. Some teachers choose to allow the robots to roam around in the hallway and collect their data in this manner. Check to be sure that each of the NewCDbot robots has the proper number of batteries installed and that they are in good working condition. It is always recommended that the robots use rechargeable batteries, as this will reduce the cost associated with operating and maintaining the robots. If rechargeable batteries are used then they can always be charged before a lab to ensure that the robots will have good batteries, which in turn will help to reduce any unwanted delays in the lab. It would also be beneficial to collect 10-50 data points and use this data to create a sample spreadsheet and graph for illustrative purposes. Below are sample graphs of Velocity versus Time and Distance versus Time.




*A sample graph of Velocity versus Time.*




*A sample graph of Distance versus Time.*

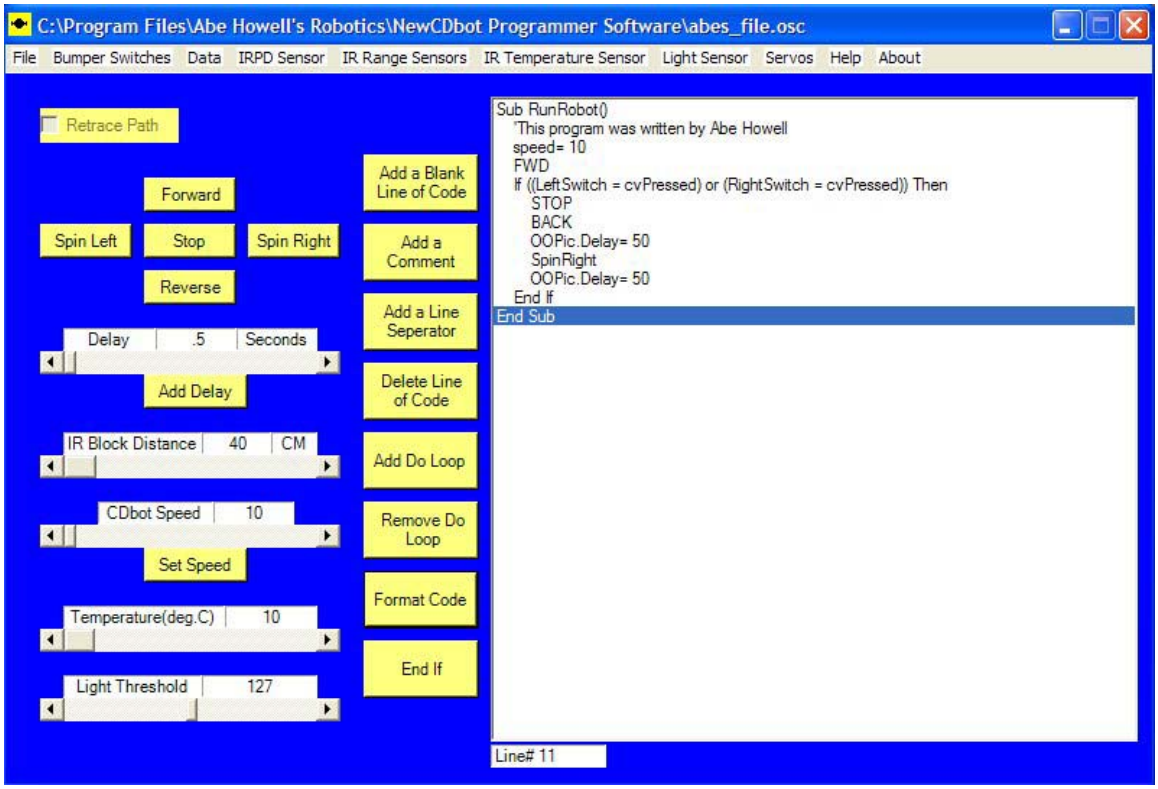
The Day of the Lab:

Students should be partitioned into groups of four or five for this lab. However, this number strongly depends upon the number of available NewCDbot Kits. It has been found that groups of four or five appear to work best and allow each the students to obtain a close interaction with the NewCDbot. At this time several questions can be presented to the students for them to answer in a group-wise manner. Several sample questions are listed below.

*(1) What is scientific data collection?*
*(2) What kind of data do scientists collect?*
*(3) How do scientists collect their data?*
*(4) What kind of data would your robot collect?*

At this time the student groups should be given five or ten minutes to answer these questions. At the end of the lab another five to ten minutes should be provided so that the students can discuss their answers to the above questions.

Once the students have completed answering the above questions a brief description of the lab experiment can be given to the students. It would be beneficial to demonstrate programming a NewCDbot for the data collection so that the students will have an understanding of what to do. Start by opening the NewCDbot Programmer Software (NCPS) and create a new program by clicking the File menu and then New.



*Screenshot of the NewCDbot Programmer Software*

Click the delete button to delete the commented line of code that reads, “Delete this line of code and add your own code”. Explain that clicking the “Forward” button will cause the NewCDbot to start driving forward. Now explain that we must tell the NewCDbot what to do when it sees an obstacle. We can do this using the “IRPD Sensor” Menu. We will use the “If Either Front IRPD is Blocked” Menu command, which is under the "Front IRPD" Menu Item. The code should appear as shown below.

```
Sub RunRobot()
    Call FWD
    If GetFrontLeftIRPD = 0 Then
End Sub
```

*Screenshot of the NewCDbot Programmer Software*

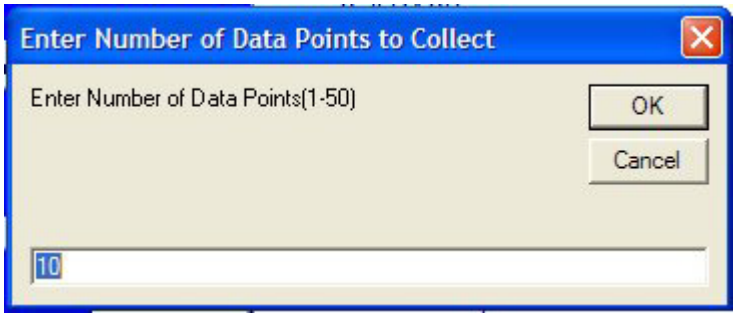
Explain that we must tell the NewCDbot what to do when it sees an obstacle. Since the NewCDbot has detected an obstacle with its left eye, we may want it to backup for a specified amount of time and then spin right for a specified amount of time. Using the “Reverse”, “Add Delay”, and “Spin Left” buttons will accomplish this. Click the Reverse button to make the NewCDbot backup away from the obstacle. To specify the amount of time to backup we need to use the Add Delay button, but first we must set the delay time using the Delay scrollbar. The default 0.5 seconds is fine for this exercise. Now we can click the Add Delay button. Next we want the robot to spin to the right for a set amount of time, we will first click the “Spin Right “ button and then the Add Delay button. To complete the “If Front LeftIRPD Blocked” Menu command we must click the “End If” button. Complete the exact same process for the NewCDbot’s right IRPD sensor and the code should be as shown below. Use the “Format Code” button to format the code with the proper tab spacing.

```
Sub RunRobot()
    Call FWD
    If GetFrontLeftIRPD = 0 Then
        Call BACK
        OOPic.Delay= 50
        Call SpinRight
        OOPic.Delay= 50
    End If
    If GetFrontRightIRPD = 0 Then
        Call BACK
        OOPic.Delay= 50
        Call SpinLeft
        OOPic.Delay= 50
    End If
End Sub
```

*Screenshot of the NewCDbot Programmer Software*

Now explain to the students that this code will only execute once if we downloaded it to the NewCDbot. To make the NewCDbot check for obstacles continuously we must add a Do Loop using the “Add Do Loop” button. By adding the Do Loop to our program our NewCDbot will now check its sensors repeatedly for any obstacles that may be in its way. Finally, we need to tell our NewCDbot to collect a set number of data points; we

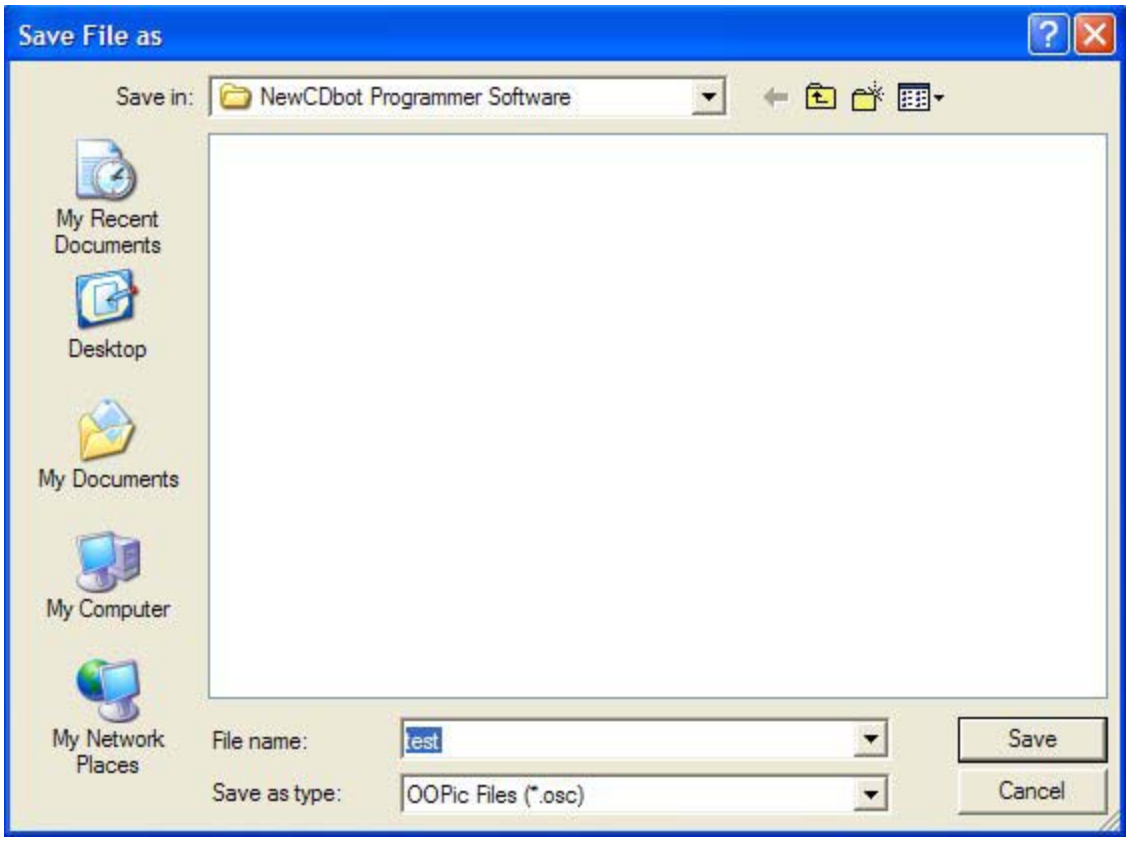
will use the Data Menu command, “Collect Data”. After clicking the Collect Data menu we will be asked to input the number of data points to collect. We will choose to collect 10 data points and then click the okay button. We have now configured our NewCDbot to collect a total of 10 data points. Once the NewCDbot has collected the 10 data points it will then stop so that we can download the data to a computer. Your code should appear as shown below.



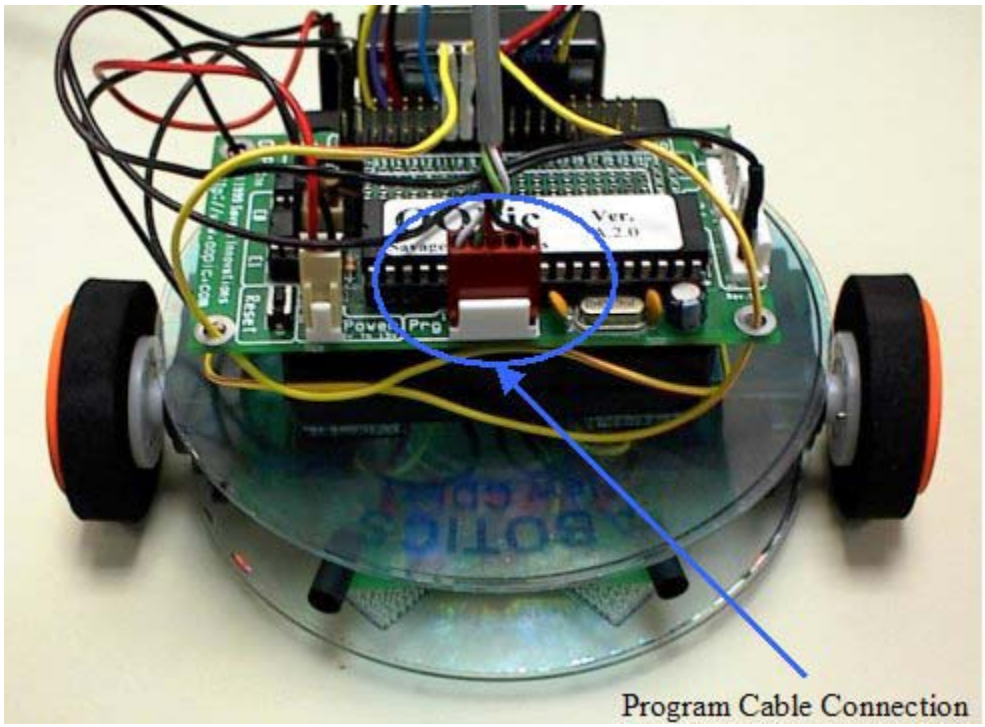
*Screenshot of the NewCDbot Programmer Software*

```
Sub RunRobot()
    NumDataPts= 10
Do
    Call FWD
    If GetFrontLeftIRPD = 0 Then
        Call BACK
        OOPic.Delay= 50
        Call SpinRight
        OOPic.Delay= 50
    End If
    If GetFrontRightIRPD = 0 Then
        Call BACK
        OOPic.Delay= 50
        Call SpinLeft
        OOPic.Delay= 50
    End If
Loop
End Sub
```
*Final screenshot of the NewCDbot Programmer Software*

Now the process of saving the program to file can be demonstrated. Click the File menu and then click the SaveAs menu item and a dialog box will appear asking you to enter a name for your program. Enter a name for the program, for this demonstration we will enter “Test” as the name for our program.

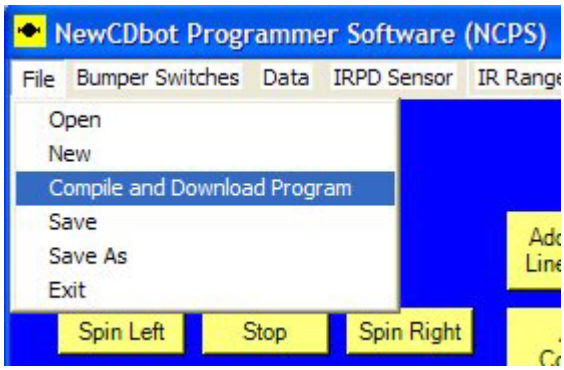


*Screenshot of the NewCDbot Programmer Software SaveAs*



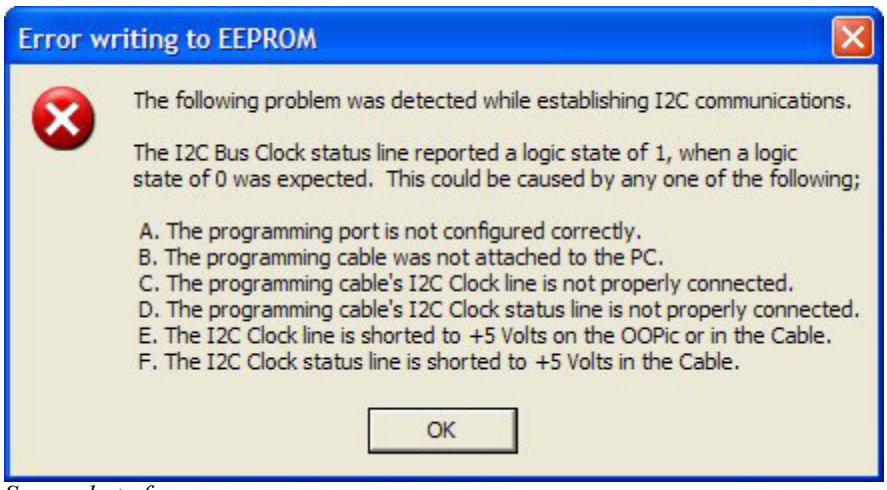
*Picture of programming cable connected to NewCDbot*

We are now ready to compile and download our "Test" program to the NewCDbot. To compile and download our program we must first connect the programming cable to our NewCDbot. Observe the picture shown above for exactly where to connect your programming cable on the NewCDbot. Once the programming cable has been connected as shown above, click on the File menu and then select "Compile and Download Program" as shown below. This will open the OOPic Multi-Language Compiler and load your program.



*Screenshot of the NewCDbot Programmer Software*

Once the OOPic Multi-Language Compiler has loaded, hit the "F5" key and the program will begin compiling and then finally it will download the compiled program to your NewCDbot. Be sure to switch on the NewCDbot before downloading! The on-switch is located on top of the NewCDbot by the 9-volt battery. If there's already a program loaded into your NewCDbot it may start to move when you switch on the power. It may be necessary to hold your NewCDbot up in the air during the compiling and downloading process. If you forget to connect your NewCDbot to the programming cable or don't switch on your NewCDbot then you may receive an error message like the one shown below.

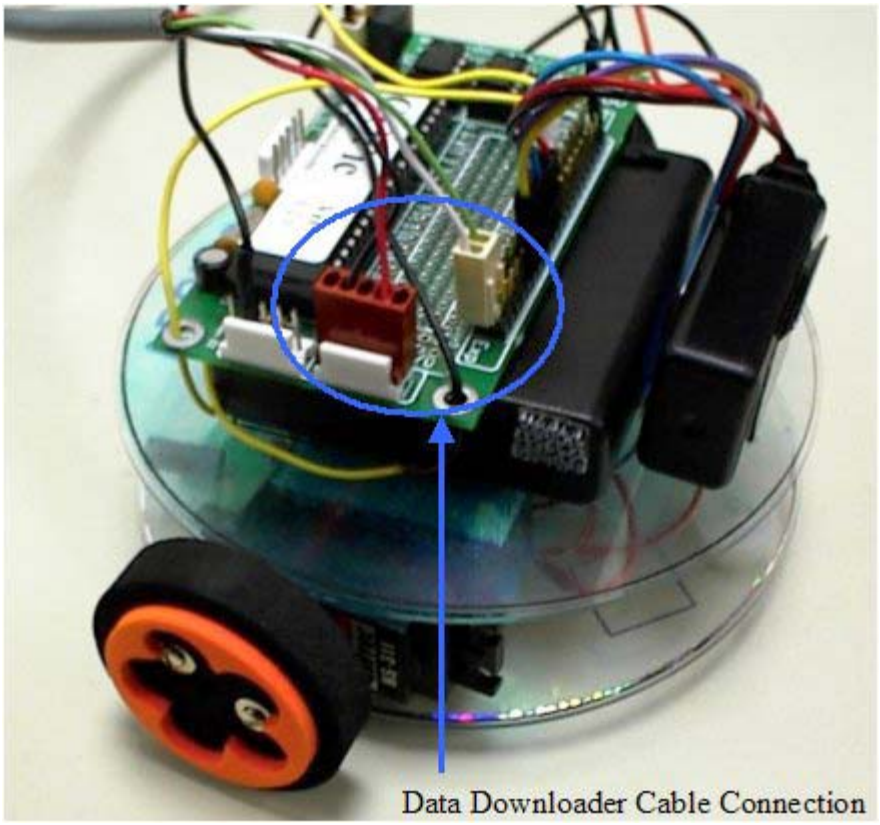


*Screenshot of error message*

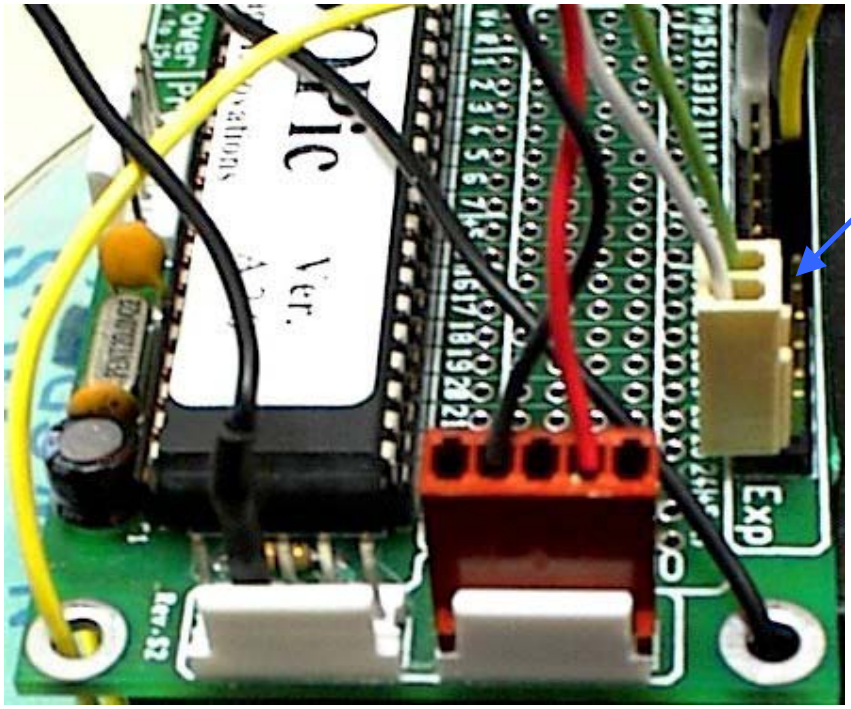
*Successful program download*

If the program download was successful then you should receive a message box like the one shown above. It should state the total number of bytes that were downloaded to the robot. Turn off the robot as soon as the download is complete. The programming cable can now be removed and the robot can be placed on the floor or wherever it desired for the demonstration.

After the robot has collected its data it will stop and the process of downloading the data to a computer can be illustrated and explained. First bring the NewCDbot back to the desktop or laptop computer that was used for downloading the "Test" program and connect the Serial Data Downloading cable up to the NewCDbot as shown in the picture below.

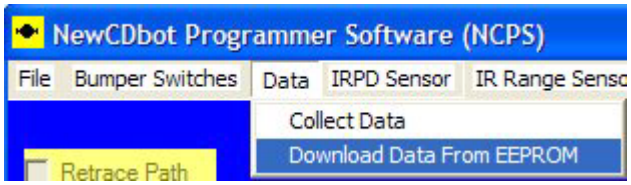


*Data Downloader Cable Connections*

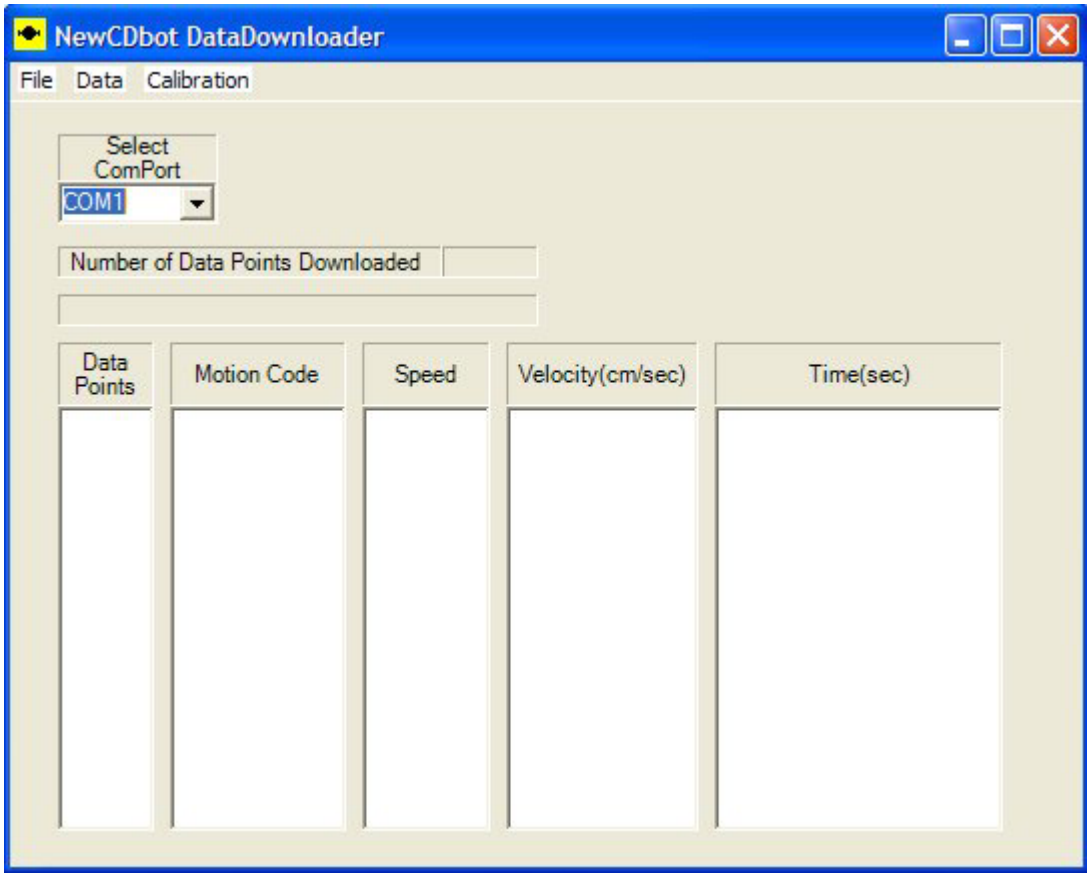



*Data Downloader Cable Connections*

Be sure to connect up the connector with the white and green wire first, otherwise the robot may reset itself and you will have to send it back out to collect the data again. Once the cable has been properly connected, open the NewCDbot Data Downloader Software by clicking on the "Data" menu and then selecting "Download Data From EEPROM" as shown below.

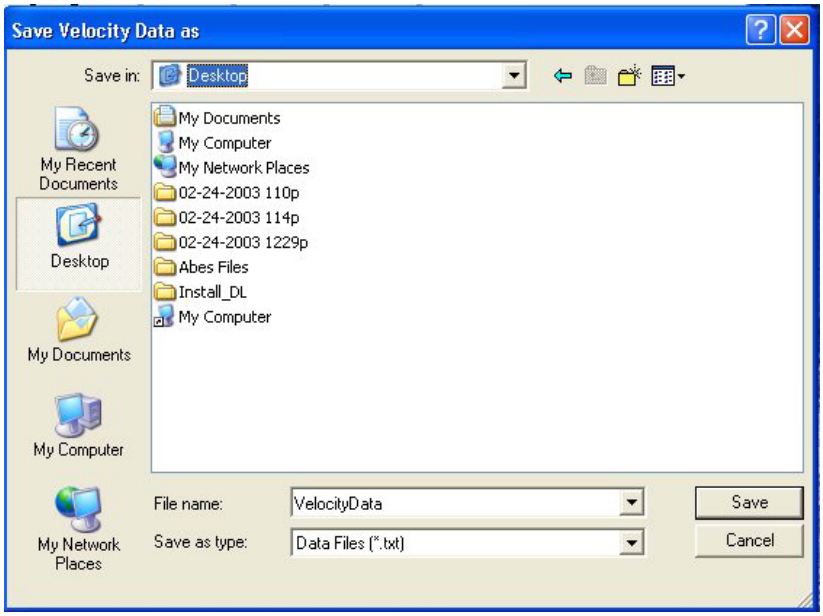



*Download Data From EEPROM*



*NewCDbot Data Downloader Program*

Be sure to select the proper communication Port, COM1 is usually the correct one. It is better if this is determined ahead of time before the actual lab day. Now you are ready to click the Data menu and select Download Data From EEPROM. The Download Progress meter should begin incrementing to indicate the current status of the download session. Once the download is complete, all the data will be displayed in its proper list. The downloaded data can be saved to file using the Data menu and by selecting Write Data to File. A file save dialog box will appear prompting you for the name of the Velocity Data file and then for the name of the Time Data file, enter whatever names you like for each file. Now that the collected data has been saved to file it can be used to create spreadsheets and graphs.



*Screenshot of Velocity Data save dialog box*

To create the spreadsheets and graphs, students will simply open the appropriate Velocity and Time data files using notepad or any other text editor and then copy and paste their collected data into a spreadsheet. Once the Velocity and Time data are copied into a spreadsheet, then the students can easily create a graph of Velocity versus Time. Which program is used to create the spreadsheets and graphs is left entirely up the teacher as the teacher and the students may already be familiar with a specific software package. To create the graph of Distance versus Time students will have to add up the areas under each of the steps in the Velocity versus Time graph and that will yield the appropriate distances needed for the Distance versus Time graph. This may be beyond the scope of the planned lesson and it is left up to the teacher to determine whether or not to cover this topic.

Now that the students have been given an example of what they are to accomplish in this lab they can finally begin working on their lab. During this time students will more than likely have questions about what to do or how to operate their NewCDbot. This lab can be shortened if the NewCDbots are pre-programmed with the data collection program that was discussed previously in this paper. Once the students have created their spreadsheets and graphs it would be a good time to discuss “what is happing” in the graphs. The graphs help to tell a story about where the NewCDbot was going when it was driving around.

After the students have completed this lab they should be asked to go back to the initial four questions that were asked in the beginning of the lab and add any new ideas that they may have, now that they have direct experience with regards to scientific data collection.

Please email any questions to abe@abotics.com