The RxDuino

Instruction Manual

July 18, 2012

Table of Contents

Table of Figures
Introduction
Package Contents
Installation
Getting Started4
Modes and Monitors4
Controlling the RxDuino
Installing RxDuino Console Software8
RxDuino Console Interface
RxDuino Console Commands9
Updating RxDuino Firmware9
Reporting Issues11
Using The Bug Tracker11
Reporting a Bug11
Cable Construction14

Table of Figures

Figure 1: OBD2 Port Location	4
Figure 2: RxDuino Console Thumbnail	8
Figure 3: RxDuino Console Interface	8
Figure 4: COM Selection	9
Figure 5: Firmware Update	10
Figure 6: Firmware Update Console	10
Figure 7: Bug Tracker Login	11
Figure 8: Bug Tracker Header	11
Figure 9: Bug Tracker Project Selection	12
Figure 10: Bug Tracker Report Details	12
Figure 11: DB9 Pinouts	14

Introduction

Thank you for purchasing your RxDuino! This set of instructions will help you get started using your new device, help you install the software on your PC, and perform firmware updates.

Package Contents



- The RxDuino
- OBD2-to-DB9 Cable
- USB-to-FTDI Cable

Installation

Installation of the RxDuino is a simple one-step process, simply connect the DB9 end of your OBD2-to-DB9 cable to the RxDuino, and connect the other end to your OBD2 port. An ample length of cable has been provided to allow you to store and mount the RxDuino within 6' of the OBD2 port.



Figure 1: OBD2 Port Location

To uninstall the device, simply remove the cable from the OBD2 port and the vehicle's MID will return to its normal function.

Getting Started

Congrats! Now that you have installed the RxDuino, you are ready to start using it and all of the wonderful features it provides. You will notice that as soon as the RxDuino is installed, you will be greeted with a welcome message displaying the product name and current firmware version.

After a slight delay, the RxDuino will start reading information from your OBD2 port and displaying the data on the screen. At the time of this document, the first mode that will be displayed is the ECU Monitor, and the first monitor is engine RPM.

Modes and Monitors

The RxDuino has plenty of modes and monitors to help you keep track of all of your vehicles vitals. The following tables summarizes all of the available functions of the RxDuino.

Mode	Monitors	Description	Alternate	Description
ECU Monitor	 Engine RPM Coolant Temperature 	Detect information about the vehicle from the ECU. This mode is	Response Time	Modify the response time of the ECU monitor. This allows

	 Vehicle Speed Mass Airflow Sensor Throttle Position Wideband AFR Openloop AFR Engine Load Catalyst Temp Intake Air Temp Short Term Fuel Trim Long Term Fuel Trim Vehicle Run Time Fuel Level Cylinder Advance 	similar to what other diagnostic tools do.		you to delay the display information a bit in case the data is moving too fast.
Miles Per Gallon	Instant MPG	Detects the instantaneous Miles per gallon	AVG Mpg	Detects the average miles per gallon for the current run time. This resets when the vehicle is shut off
Wheel Speed	Wheel Speed in Format: FL/FR/RL/RR	Display the current wheel speed for each individual wheel on the vehicle	N/A	N/A
Tire Pressure	Tire1Tire2Tire3Tire4	Display the pressure for each individual tire. Unfortunately there is no way to detect which tire is which when talking to the TPS module, therefore tires are named generically.	Tire Temperat ure	Display the temperature for each individual tire.
Text Mode	Default Text: "Rotary Power"	Display custom text on the screen.	Edit Text	Edit the display text via steering controls. This text is stored in memory and retained whenever the vehicle is turned off.
Set Clock	N/A	Let user know that they are currently on clock edit mode. Clock will not be modified yet.	Set Clock Enable	Allow clock modifications. This is useful if user has changed their stereo and has lost controls of the vehicles time display.
MIL Codes	MIL codes in format: MIL:None (if none) MIL:P####	Displays any currently stored MIL codes or, displays "None" if no	Clear MIL Codes	Clear all MIL codes as well as any stored codes.

		codes present		
System Off	Removes display	Turns the device into a low power mode and does not display anything on the screen or read any information from the ECU	N/A	

Controlling the RxDuino

Controlling the RxDuino is one of its best features as it is integrated into the vehicle's steering wheel through the cruise control buttons. This is done without any modifications to the vehicle or installation of any additional sensor. The following table describes each button and function.

Cruise Control Button	Function
RES/+	Press: Go to previous Monitor Hold: N/A
SET/-	Press: Go to next Monitor Hold: N/A
CANCEL	Press: Go to next mode Hold: Switch current mode to alternate
ON OFF	Press: Toggle RxDuino Hold: N/A

The important thing to remember about the ON/OFF button is that it technically does nothing to the RxDuino. At all times, the RxDuino is checking to see if the user has enabled Cruise Control. In the event that cruise control is ON or SET, the buttons presses will be ignored by the RxDuino. This means that the buttons will only control the cruise control functions. If the vehicle's cruise control is OFF, the buttons will control the RxDuino. This is done so that while the user is manipulating the cruise control, they are not toggling through RxDuino settings.

٢

Installing RxDuino Console Software

The RxDuino Console application is currently written in C# .NET 4.0, which means you will need to have the .NET framework installed on your PC. Some applications require this to be installed, so you may already have it, but if not you can download it for free from Microsoft. The application doesn't require any special installation. To launch the application, simply click on 'RxDuino Console.exe'.



Figure 2: RxDuino Console Thumbnail

Once the application is executed, you will be greeted with the RxDuino Console splash screen that will check the server for updates.

RxDuino Console Interface

After the RxDuino Console has checked for updates, you will be presented with the main interface. This simplistic interface was designed to give the user control of the RxDuino's diagnostic functions without confusion.

	RxDuino Console	Refresh
	Eile Help	COM Port
Available	Select COM Port: COM4	List
СОМ	Initializing	
Ports	Current Monitor Pid: 12 Byte Count: 2 Current Monitor Text: RPM: Current Mode Text: ECU Monitor Display Text: Rotary Power Custom Delay: 100 HSCAN Initialized MSCAN Initialized MSCAN Initialized Printing Welcome Message [RxDUINO][v1.0]	Connect To Selected COM Port
Update		
Firmware	Su	omit
	Co	mmand
Command	Firmware Version: v1.0 Update Firmware Clear Console Exit	
Input	Ex	it
	Figure 3: RxDuino Console Interface	plication

RxDuino Console Commands

The RxDuino Console has a set of commands to allow you to debug and interact with the RxDuino. The following commands are available:

Command	Description
RST	Reset the RxDuino
TXT <enter text=""></enter>	Sets the "Text Mode" text to " <enter text="">"</enter>
NEXT	Switch to the next mode
PREV	Switch to previous mode
UP	Switch to previous monitor
DOWN	Switch to next monitor
ALT	Switch to alternate mode
OFF	Turn off RxDuino
PING	Check RxDuino Connection To PC
DUMP	Print debug information

Updating RxDuino Firmware

The RxDuino console has an interface to update the firmware. To update, you must first select the COM port for the device:

🐢 RxDuino Console			
<u>F</u> ile <u>H</u> elp			
Select COM Port:	COM7 •	Connect	Refresh

Figure 4: COM Selection

Once you have selected your COM port you will notice that the "Update Firmware" button is now enabled. Click the button and you will be presented with the following dialog:

See Update Firmware	X
C:\RxDuino.rxd	Browse Update

Figure 5: Firmware Update

Firmware files will be in a *.rxd format. To select the file, click on the Browse button and use the file select dialog that pops up to find your new firmware. Select the firmware and click "Update" to start uploading the information to the RxDuino. While the update is in progress, you will notice a dialog box open up displaying progress.

C:	\Users\C	orei7\A	ppData\Local\Temp\firmu	pdate.exe								x	
500	4500	0x00	0×00									-	
500	4500	<u>o~oo</u>	efuse	Θ	Θ	Θ	0	no	1	Θ	0	4	
500	4000	0.00	lock	Θ	Θ	Θ	G	no	1	Θ	Θ	4	
500	4500	0x00	0x00										
9	0	0~00	calibration	Θ	Θ	0	Q	no	1	Θ	0		1
Ů	0	0200	signature	Θ	Θ	Θ	G	no	3	Θ	0		
Θ	Θ	0x00	0x00										
			Programmer Type Description Hardware Version Firmware Version Vtarget Varef Oscillator SCK period	: Ardu : Ardu n: 3 n: 4.4 : 0.3 : 0.3 : 28.5 : 3.3	uino uino U U 800 kH us	z							
firm	update	e.exe	: AUR device init:	ialize¢	d and	ready	, to	accept :	instruc	tions			
Reading ###################################													
firm	update	e.exe	: Device signature	e = 0x	1e950f							-	-

Figure 6: Firmware Update Console

Note: The RxDuino has safety switches to prevent it from getting "bricked" during a failed install. Although this mechanism is available, it is not recommended to disconnect the RxDuino during updates.

Reporting Issues

Using The Bug Tracker

Using the bug tracker will be very valuable to notifying the developer and the community of issues that you the user are experiencing. Everyone that purchased an RxDuino was given a username and a password (temp password) so that they are able to access the system. To access the bug tracker, please visit the link at the top of this post, http://www.therxduino.com/bug-report/my_view_page.php.

Logging Into The Bug Tracker

The first time you open the bug tracker you will be given the option to log in or to log in Anonymously. If you do not have an account, or would like to view bugs without logging in, you can click "Log In Anonymously" at the right of the login page

Login	[Login Anonymously]
Username	admin
Password	
Remember my login in this browser	
Secure Session	$\ensuremath{\overline{\mathbb{V}}}$ Only allow your session to be used from this IP address.
	Login

Figure 7: Bug Tracker Login

Reporting a Bug

To report a bug, you must first be logged into the bug tracker. If you are having trouble logging in or need a password reset, please contact me and I will be more than happy to help you out. Once logged in, you can click on "Report Issue" on the top menu bar



Figure 8: Bug Tracker Header

Users should have the option of choosing what product you would like to submit a bug for, either the RxDuino, or the RxDuino console. For this tutorial, we will select the RxDuino to report a bug on. Simply select RxDuino from the drop down box, and click "Select Project"

Select Project	
Choose Project	RxDuino -
Make Default	
Select Project	

Figure 9: Bug Tracker Project Selection

You will now be taken to the bug details page where you can fill out information about the bug you are looking to report. The most important fields are the required fields (appended with a *)

Enter Report Details		
*Category	(select)	
Reproducibility	have not tried 🗸	
Severity	minor 👻	
Priority	normal 👻	
Product Version	-	
Assign To	-	
Target Version	-	
*Summary		
*Description		

Figure 10: Bug Tracker Report Details

At the moment there is only one Category, and one person that you can assign to, so there should be no confusion there. Severity and Priority can be used at your own judgement, but please note that they may be edited if the ticket finds itself at less of a priority than another pending ticket. Please feel free to select the target version as the latest available version, but you can leave that blank if you prefer as it is not a required field.

The most important field is the "Description" field. Please try to be concise so that I can easily reproduce the error.

Cable Construction

There may be instances where the user would like to troubleshoot their connection by checking their wiring. The construction of the cable is made up of two parts, an open ended OBD2 cable, and a female DB9 connector. To verify that your cable is correctly assembled, you may open the DB9 end and check to make sure that the pinouts match the following diagram. Please take note that the diagram below is a view of the back (solder) side of the DB9 adapter.



Figure 11: DB9 Pinouts