

Data Packet Designer and Simulator startup Guide

CodeName 4x4

Contents

[I. Requirements 3](#_Toc506894021)

[II. Creating a custom template 4](#_Toc506894022)

[III. Downloading and installing the 4x4 Firmware 8](#_Toc506894023)

[1. Deploy the simulator firmware to the 4x4 8](#_Toc506894024)

[2. Debug the firmware on the 4x4 9](#_Toc506894025)

[IV. Connecting the 4x4 to a computer through UART 11](#_Toc506894026)

[V. Downloading and installing the DataPacket Simulator 12](#_Toc506894027)

[VI. Testing the flow 13](#_Toc506894028)

[VII. Troubleshooting 16](#_Toc506894029)

[1. The maximum baud rate for the device is X. 16](#_Toc506894030)

# Requirements

In order to perform all the required steps, this guide assumes that:

* You have a Codename 4x4 device with a micro USB cable.
* You have a UART to USB dongle with at least 2 pin plugs for TX / RX. The dongle driver must be installed.
* Your 4x4 was properly set up using the document “**GettingStartedTP3.3.pdf**”.
* The Wi-Fi connection of the device has been properly set up using dutil.
* You have deployed the Data Packet Designer starter solution using the PowerShell Script.

# Creating a custom template

Open the Data Packet Designer web app which you deployed using the PowerShell script and login.

Before you can test the simulator, you first need to create a new data template. This data template will be based on pre-built starter templates.

|  |
| --- |
| Select Refridgeration  Start by picking a starter template. For example “Refrigeration”. |

|  |
| --- |
| Select Create your own  You can now pick a pre-built template that best suits your needs:   * Create your own comes with a minimal number of properties for the header * Refrigerator Simple comes with one body property * Refrigerator Smart comes with many body property to showcase a more complex device |

|  |
| --- |
| Click “Save” when finished  Add or remove nodes, rename properties or reorder them.  A property should never be empty.  Name your template  Display a JSON view of the template  You can now design the message that will be send to IoT Hub using the simulator and your device.  Once you save the template, it will appear in your home, under “Your Saved Data Template” so you can edit it later. |

|  |
| --- |
| Click “Simulate”  Once the template is saved, you can still choose to make some modification, or click on “Simulate” to go to the next screen and install the simulator. |

# Downloading and installing the 4x4 Firmware

This guide assumes that the step 1 and 2 were already completed.

|  |
| --- |
| On the Simulation page, click on “Download and Deploy 4x4 Code” |

|  |
| --- |
| Connect your device to your computer using a micro SD cable. The network adapter “sl0” should now be connected. |

## Deploy the simulator firmware to the 4x4

|  |
| --- |
| Extract the downloaded file and locate the file “MS.IoT.FourByFour.App.imagepackage”. |
| Start a new instance of the Command Prompt(Admin mode) for 4x4 Visual Developer, and enter the following command:  “**dutil sideload deploy -p "<PathToImagePackage>\MS.IoT.FourByFour.App.imagepackage**"  You should see a message “Success” if the deployment succeeded. Then enter:  “**dutil sideload start**”  The firmware is now running on your 4x4. |

## Debug the firmware on the 4x4

If you wish to compile and deploy the code to the 4x4 for debugging, please proceed as it:

|  |
| --- |
| Extract the downloaded file and start the Visual Studio Project “**MS.IoT.FourByFour.App.sln**” |

|  |
| --- |
| Once Visual Studio has finished loading the project, press F5 to start the debugging. If the device is properly connected to the computer through USB and the Wi-Fi connection is set, you should see the text above in the output panel of VS.  Note: At this point it is normal for the firmware to not know the SimulatorConnectionString. This variable will be automatically sent by the Simulator to the 4x4 through UART. |

# Connecting the 4x4 to a computer through UART

|  |
| --- |
| Plug in the USB to UART dongle to your computer. Make sure that the driver is properly installed before attempting a connection. |

|  |
| --- |
| Connect the RX pin of the 4x4 to the TX pin of your dongle, and the TX pin of the 4x4 to the TX pin of your dongle. You might need to look at the documentation of your dongle. |

# Downloading and installing the DataPacket Simulator

|  |
| --- |
| Download and extract the Simulator Installation. |

|  |
| --- |
| Install the simulator “**MS.IoT.Simulator.Installer.msi**”. |

# Testing the flow

|  |
| --- |
| Launch the newly installed application “4x4 Simulator”. |

|  |
| --- |
| Log in using the same credentials that you used with the DataPacket Designer portal. |

|  |
| --- |
| From this point and forward, you will see two indicators:   * USB: Connected indicates that the USB connection is established. The network SL0 is connected. * UART: Connected indicate that a proper communication is established between your 4x4 device and the simulator.   If the UART connection doesn’t indicate “Connected” please make sure:   * That the proper dongle is selected using the drop down list. * That the RX pin on the 4x4 is connected to the TX ping on the dongle, and that the TX pin on the 4x4 is connected to the RX pin on the dongle.   Select the user template you want to test in order to get to the next screen. |

|  |
| --- |
| This screen will preview an example of a message that will be sent to IoT Hub. The “status” runs a few additional checks to ensure that the device is connected to Wi-Fi, and will transfer the IoT Hub connection string to the device.  You should see the status “Connected” when all the checks have passed.  When you are ready to start the test, click on “Simulate!” |

|  |
| --- |
| On the left, you will see the messages being sent from the Simulator to the 4x4 via UART.  The 4x4 sends the messages to IoT hub deployed in your subscription by the PowerShell script.  These messages are then redirected by the stream Analytics job and stored into CosmosDB.  On the right, you will see messages arriving from CosmosDB.  This shows that messaging flow is complete. |

# Troubleshooting

## The maximum baud rate for the device is X.

This error is likely due to the driver used to install the USB to UART dongle. In some cases, the driver can be bugged and won’t report the right maximum baud value for the dongle.