

# **Intrepid Control Systems, Inc.**

## **Example Scripts for Wireless Data Logger**

### **Document Number: AN-ICSI-1008**

Rev 2.0 08/2014

### Contents

| 1. | Int | roduc   | tion:  | 3 |
|----|-----|---------|--|---|
| 2. | Sc  | ripts f | or Wireless Data Logger  | 3 |
| 2  | 2.1 | Scri    | pt 1: Log Entire Bus messages without any start condition                  | 3 |
|    | 2.1 | l.1     | Objective:   | 3 |
|    | 2.1 | 1.2     | Procedure:   | 3 |
| 2  | 2.2 | Scri    | pt 2: Log selected messages on condition. Sleep condition active.          |   |
|    |     | Upl     | oad data to server   | 8 |
|    | 2.2 | 2.1     | Objective:   | 8 |
|    | 2.2 | 2.2     | Procedure:   | 8 |
| 2  | 2.3 | Scri    | pt 3: Log DTC and DID Response. Update collection on server periodically 1 | 7 |
|    | 2.3 | 3.1     | Objective:   | 7 |
|    | 2.3 | 3.2     | Procedure:   | 7 |
| 2  | 2.4 | Scri    | pt 4: Log selected messages before and after a trigger 2                   | 5 |
|    | 2.4 | 4.1     | Objective:   | 5 |
|    | 2.4 | 4.2     | Procedure:   | 5 |
| 3. | Co  | ontact  | Us:  | 2 |

#### 1. Introduction:

Wireless data loggers (neoVI PLASMA/neoVI ION) can log many types of data. The neoVI PLASMA/neoVI ION does this by using Core Mini proprietary executable that runs within the logger VNET Module (network adapter card). The script runs independently of the Android OS, and it's highly flexible to allow for making requests, initiating network management and calibration sessions, and more.

This document gives some examples to explains how to write script and upload to wireless neoVI data logger to log data for different scenarios.

#### 2. Scripts for Wireless Data Logger

#### 2.1 Script 1: Log Entire Bus messages without any start condition

#### 2.1.1 Objective:

- To start data logging immediately.
- Collect Entire bus messages.
- Create collection of 50000 messages.

#### 2.1.2 Procedure:

Launch Vspy and create new Platform or select existing from current platform list. (Please refer the <u>Script 1.vs3zip</u> file)

- **2.1.2.1** Upload DBC file:
  - Select 'Setup' tab  $\rightarrow$  <u>Network database</u>.
  - Select HS CAN from Networks column on the left.
  - Add .dbc file to this network, from a tab 'Add' on the right.
  - Save the Platform Changes.
  - Same way you can add DBC and VS3 database for other network

#### 2.1.2.2 Configure <u>VehicleScape DAQ</u>:

Select Measurements  $\rightarrow$  VehicleScape DAQ  $\rightarrow$  Standalone Logging  $\rightarrow$  Messages

|   | Channels PC Loggin  | g Standalone Logging   | Gateway Online   | DAQ 1  |
|---|---|--|--|--|
| •\v• Messages   | Histogram   | Bus Query  |  |  |
| etup the <u>format</u> of the file  | e name. This is also y  | our collection name.   |  |  |
| Entire Bus Collection   | SO  |  | Append Time  | and Date to file name  |
| ssage Collection Options  | >(3)  |  |  |  |
| Which messages should w<br>specific messages. You   | e collect? Select " <u>Ent</u><br>a can select these me   | <u>tire Bus</u> " to <b>capture all</b><br>essages on the "Channel   | messages. Choose<br>s" tab.  | "Selected Channels" to only capture  |
| Entire Bus  | <b>(4)</b>  | O Selected Channels  |  | Advanced Options   |
| lection Start Options   | ~   |  |  |  |
| ilection start Options  |   |  |  |  |
| When should we start colle<br>a conditional start Sel   | ecting? Select " <u>Imme</u><br>lect "Using Trigger" is   | diately" to start when   | the logger is powe   | red. Select " <u>When Expression is True</u> " for   |
| Conditional Start, Sci  | eet osing migger is   | used to capture bero   |  |  |
| <ul> <li>Immediately</li> </ul>   |   | <b>f</b> * {SECOND}>=(   | )  |  |
| When Expression is  | True  | OR neoVI MIC   | button press   |  |
| Ulsing Trigger Expre  | ssion   | OR neoVI MOT   | TE / Push Button Pend  | ant on MISC 5 button press   |
| O boing migger expre  |   |  | -,   |  |
| O baing migger expre  |   |  |  |  |
| Should we always collect  | t the DAQ channe  | els even if we're not logo   | ing? (Needed if start  | expression/trigger is a DAQ item.)   |
| Should we always collect  | ct the DAQ channe   | <b>els</b> even if we're not logo  | jing? (Needed if start   | expression/trigger is a DAQ item.)   |
| Should we <b>always collec</b>  | ct the DAQ channe   | els even if we're not logg   | jing? (Needed if start   | expression/trigger is a DAQ item.)   |
| ihould we <b>always collec</b>  | ct the DAQ channe   | even if we're not log <u>o</u>   | jing? (Needed if start   | expression/trigger is a DAQ item.)   |
| ihould we <b>always collec</b><br>Always DAQ  | ct the DAQ channe   | even if we're not logg   | jing? (Needed if start   | expression/trigger is a DAQ item.)   |
| hould we <b>always collec</b><br>Always DAQ<br>ection Configuration<br>How should we finish eac   | t the DAQ channe  | els even if we're not logg<br>"Finish After # of Messa<br>"Finish an Everynarian" t  | ging? (Needed if start   | expression/trigger is a DAQ item.)   |
| ihould we <b>always collect</b><br>Always DAQ<br>Always DAQ<br>Always DAQ<br>Always DAQ<br>Always DAQ<br>Always DAQ<br>Always DAQ<br>Always DAQ<br>Always DAQ<br>Always Collect<br>Always Collect<br>Always Collect<br>Always DAQ<br>Always Collect<br>Always Collect<br>Always DAQ<br>Always Collect<br>Always DAQ<br>Always Collect<br>Always DAQ<br>Always Collect<br>Always DAQ<br>Always Collect<br>Always DAQ<br>Always Collect<br>Always Collect<br>Always DAQ   | t the DAQ channe  | els even if we're not logg<br>"Einish After # of Mess:<br>"Finish on Expression" tr  | ging? (Needed if start<br>ag <u>es</u> " to finish each co<br>o finish collecting <b>whe</b>                   | expression/trigger is a DAQ item.)<br>illection once we've collected a<br>in an expression is true.  |
| Should we <b>always colled</b><br>Always DAQ<br>lection Configuration<br>How should we finish eac<br><b>specified number of n</b><br>Finish After # of N  | the DAQ channe  | els even if we're not logg<br>" <u>Finish After # of Messa</u><br>" <u>Finish on Expression</u> " to<br>O Finish   | ang? (Needed if start<br>ages" to finish each co<br>o finish collecting <b>whe</b><br>on Expression            | expression/trigger is a DAQ item.)<br>illection once we've collected a<br>in an expression is true.  |
| Should we always collect<br>Always DAQ<br>lection Configuration<br>How should we finish eac<br>specified number of m<br>Finish After # of M<br># of Messages  | t the DAQ channe  | els even if we're not logg<br>"Einish After # of Mess:<br>"Finish on Expression" tr<br>O Finish<br>~3 05176MB  | ping? (Needed if start<br>ages" to finish each co<br>o finish collecting <b>whe</b><br>on Expression           | expression/trigger is a DAQ item.)<br>sllection once we've collected a<br>an an expression is true.  |
| Should we always collect<br>Always DAQ<br>lection Configuration<br>How should we finish eac<br>specified number of m<br>Finish After # of N<br># of Messages  | the DAQ channe<br>the collection? Choose<br>messages. Or, pick<br>fessages<br>50000 v   | els even if we're not logg<br>"Finish After # of Messi<br>"Finish on Expression" tr<br>~3.05176MB  | ging? (Needed if start<br>ages" to finish each co<br>o finish collecting <b>whe</b><br>on Expression           | expression/trigger is a DAQ item.)<br>illection once we've collected a<br>en an expression is true.  |
| Should we <b>always colled</b><br>Always DAQ<br>Ilection Configuration<br>How should we finish ead<br><b>specified number of n</b><br>Finish After # of M<br># of Messages  | the DAQ channe<br>the collection? Choose<br><b>nessages</b> . Or, pick<br>(dessages)<br>50000   | els even if we're not logg<br>"Finish After # of Messa<br>"Finish on Expression" to<br>O Finish<br>~3.05176MB  | aing? (Needed if start<br>a <u>aces</u> " to finish each co<br>o finish collecting <b>whe</b><br>on Expression | expression/trigger is a DAQ item.)<br>vilection once we've collected a<br>en an expression is true.  |
| Should we <b>always colled</b> Should we <b>always colled</b> Always DAQ Mection Configuration How should we finish ead specified number of m   | the DAQ channe<br>the collection? Choose<br>nessages. Or, pick<br>4essages<br>50000   | els even if we're not logg<br>"Finish After # of Messa<br>Finish on Expression" to<br>Finish<br>~3.05176MB   | aing? (Needed if start<br>ages" to finish each co<br>o finish collecting whe<br>on Expression<br>Collec        | expression/trigger is a DAQ item.)<br>illection once we've collected a<br>in an expression is true.  |
| Should we always colled<br>Always DAQ<br>Always DAQ<br>Hection Configuration<br>How should we finish eac<br>specified number of r   | the DAQ channe<br>the collection? Choose<br>nessages. Or, pick<br>fessages<br>50000   | els even if we're not logg<br>"Einish After # of Messa<br>"Einish on Expression" to<br>O Finish<br>~3.05176MB<br>we do? Select "Stop" to                               | ages" to finish each co<br>o finish collecting whe<br>on Expression  | expression/trigger is a DAQ item.)<br>illection once we've collected a<br>in an expression is true.  |
| Should we always collect<br>Always DAQ<br>lection Configuration<br>How should we finish eac<br>specified number of r<br>Finish After # of N<br># of Messages<br>start Options<br>Show we collect S0000 me<br>capture. Select "Restart"  | the DAQ channe<br>ct the DAQ channe<br>the collection? Choose<br>nessages. Or, pick<br>4essages<br>50000 V<br>ssages, what should<br>to start capturing   | els even if we're not logg<br>"Einish After # of Mess:<br>"Einish on Expression" tr<br>$\sim$ 3.05176MB $f_{\pi}$<br>we do? Select "Stop" to<br>g again (will wait for | ages" to finish each co<br>o finish collecting whe<br>on Expression  | expression/trigger is a DAQ item.)<br>illection once we've collected a<br>in an expression is true.<br>tion Upload (wireless only)<br>tend of the collection, upload the data to<br>we nearly 72 feater to abundle               |
| Should we always collect<br>Always DAQ<br>Always DAQ<br>Alection Configuration<br>How should we finish eac<br>specified number of n<br>Finish After # of M<br># of Messages<br>Start Options<br>Start Options<br>St | the DAQ channe<br>the collection? Choose<br>messages. Or, pick<br>fessages<br>50000 v<br>sagges, what should<br>to start capturing<br>ce Restart to resta | *Einish After # of Mess:<br>*Finish on Expression" tr<br>~3.05176MB<br>we do? Select "Stop" to<br>g again (will wait for<br>art regardless of the s                    | ages" to finish each co<br>o finish collecting whe<br>on Expression<br>end the<br>start<br>tart                | expression/trigger is a DAQ item.)<br>expression once we've collected a<br>en an expression is true.<br>tion Upload (wireless only)<br>tend of the collection, upload the data to<br>ess neoVI? Select both for fastest network. |

Figure 1.a: Settings for Script 1

- Collection name: Write a Collection name in collection name box (Figure 1.a(1)) (Default name will be Collection 1) and check the box on its side to append Date & Time to the file name (Figure 1.a (2)). So it will create collection by selected name with date and time.
- Message collection option (Figure 1.a (3)): Select 'Entire bus' option.
   So it will log entire message present on bus.
- Collection start option (Figure 1.a (4)):
  - Select 'when Expression is true'
  - Click on *f* tab. It will open 'Select start Expression' window.

- Go in 'Misc' and select 'Second' property from Time category of list from the right side block (Figure 2 (2)).
- Double click on selected property it will come in Expression block on top of window (Figure 2 (1)).
- Set expression 'Second >=0'. This Expression is always true type expression .hence it is equivalent to immediate condition. We can also use other always true type expression.
- Click on OK option

| <b>=</b>                  | S                              | elect Start Expression |                |                     | - 🗆 🗙       |
|---------------------------|--------------------------------|------------------------|----------------|---------------------|-------------|
| Enter Expression for sign | al                             |                        |                | Help                | OK Cancel   |
|                           |                                |                        | Custom         | Format              |             |
| Description Second        | $\square$                      |                        | Format         | ~                   | Min         |
| Expression {SECOND}>=0    |                                |                        | Units          |                     | Max         |
| Clear                     | Evaluate as text               |                        |                | Discrete Values     |             |
| f* Expression Builder     |                                |                        |                |                     |             |
| •••Rx Messages            | Sort By: 🗸                     | Add To Expression      | Add Operator   | Calculator Pan      | el          |
| 🛛 🞯 Database              |                                |                        |                | <u>^ /</u>          | * _         |
| Tx Messages               | Find                           | Clear                  |                | 7 9                 |             |
| Signal Groups             | () Abs Time (Sec)              | ~                      | >              |                     | +           |
| ₽DAQ                      | (2) Ain Report Interval (ms)   |                        | <              | 4 5                 | <b>b</b>    |
| <b>@Jobs</b>              | (2) Analog Sample Rate (1000s) |                        | <>             | 1 2                 | 3 )         |
| 🍄 App Signals             | 🖓 Day of Week                  |                        | >=             | 0                   | . (         |
| a Retworks                | Hour (2)                       |                        | >>             |                     |             |
| 구.Nodes                   | (2) Minute                     |                        | <<             | - 🉄 Application Sig | inals       |
| Misc                      | Month                          |                        | bit0(          | Name                |             |
| Function Blocks           | Second                         |                        | DIT1(<br>bit2( |                     |             |
| Physical IO               | 191 time (mc)                  |                        | bit3(          | Add                 |             |
| Logger                    | N Year                         | ~                      | bit4           |                     |             |
|                           |                                |                        | bit5(          | ✓ Hide auto-gene    | rated items |
|                           | Properties                     |                        | bit7(          | Test                |             |
|                           | (None)                         |                        | byte0(         |                     |             |

Figure 2: Settings for Collection Start expression

**Note:** Upload data at the end of collection option is not applicable for 'immediate' option. Hence here we selected 'when Expression is true' option and written condition equivalent to immediate.

- Collection configuration (Figure 1.a (5)): Select 'Finish after # of messages' option → select # messages 50000. Here we can set number of messages as per requirement.
- Restart option (Figure 1.a (6)): Select 'Restart' option. So it will restart logging after completion of collection. It will again check for start collection expression for restart logging.
- Collection upload (Figure 1.a (7)): Check box of 3G and WiFi option as per availability or select both.
- Reporting option (Figure 1.b (8)): Check box of 'LED' option. It will give process status on LEDs of device.
- Power management (Figure 1.b (9)): Select 'Never' option. So device will not go in sleep mode.
- Generate (Figure 1.b (10)): Select 'Generate decoding database',
   'Transfer to SD card' and check box of 'generate for wireless device'
   (Figure 1.b (11)).

| Reporting Options<br>How would you like feedback about the status of the logger? Select " <u>LED</u> " to <b>flash the LEDs</b> based on t<br>Select " <u>neoVI MOTE</u> " to <b>use the neoVI MOTE</b> to display status and manually trigger.  | he status of the capture.   |
|--|---|
| ✓ LEDs neoVI MOTE Beep on Wakeup (PLASMA only)<br>When the logger is running, the <b>red</b> LED will blink at a fast rate. When data is captured to the SD card, slowly. When capturing post-trigger data, the <b>green</b> LED will flash quickly.   | , the green LED will flash  |
| Power Management  Power Management  Disable all logger TX'ing on this:  Expression  When should the neoVI go into low-current draw (sleep) mode? Select "Never" to always stay awake.  Present" to go to sleep once a message is not on the bus after a certain period of time. Likewise,  an expression is true for a period of time and "On No Bus Activity" will sleep once the bus is quiet.  Cause the peoVI to stay awake or make or make on the bus | Select " <u>When Message Not</u><br><u>On Expression</u> " will sleep <b>once</b><br>Note: <b>Any bus activity will</b> |
| Never     When Message Not Present     On Expression     Generate     Vehicle Spy can generate a database that will decode all the data logged by this setup. It does this b     currently have loaded. If you want this, check "Generate decoding database" and then choose if you'd     to SD card" or just "Save to local file system". Then, click "Generate" and go!  | On No Bus Activity  |
| <ul> <li>✓ Generate decoding database</li> <li>○ Transfer to SD card</li> <li>○ Save to local file system</li> <li>✓ Generate f</li> <li>✓ Advanced C</li> </ul>   | or Wireless Devices   |
| Generate   |   |

Figure 1.b: Settings for Script 1

#### 2.1.2.3 Create wivi file:

- To create .wivi file of all configuration. Click on tab (Figure 1.b (12)) on VehicleScape DAQ.
- It will open the 'Coremini Executable Generator' window.
- Make sure that coremini compiled without error (Figure 3 (1)).
- Click on 'Export wivi file' option (Figure 3 (2)). It will export wivi file then Save file in destination folder
- Upload the file to device from wivi server and check output.

| CoreMini Executable Generator  | ×              |  |  |  |  |  |  |
|--|----------------|--|--|--|--|--|--|
| Build SD Card Partition Advanced Settings  | Help           |  |  |  |  |  |  |
| Build       SD Card Partition       Advanced Settings       Help <ul> <li>Capturing EMS_Message_(copied from DB)</li> <li>Capturing NEFM_Message (copied from DB)</li> <li>Capturing MBFM_Message (copied from DB)</li> <li>Using standard filtered circular capture block for Collection 1.</li> <li>Generating message database for reading data files</li> <li>Message database file: C:\IntrepidCS\Vehicle Spy 3(60045)\Data Directory\Loggere Scipts\DAQ 1_MessageDatabase.vsdl</li> <li>File added to SD Card image: C:\IntrepidCS\Vehicle Spy 3(60045)\Data Directory\Loggere Scipts\DAQ 1_MessageDatabase.vsdl</li> <li>File added to SD Card image: C:\IntrepidCS\Vehicle Spy 3(60045)\Data Directory\Loggere Scipts\DAQ 1_MessageDatabase.vsdl</li> <li>File added to SD Card image: C:\IntrepidCS\Vehicle Spy 3(60045)\Data Directory\Loggere Scipts\DAQ 1_MessageDatabase</li> <li>Exe Compressed to 5120 bytes (31.4 % of original size)</li> <li>CoreMini Header (270 bytes)</li> <li>Network Objects Compiled: 217 (868 bytes)</li> <li>Process Lists: 28 (28 bytes)</li> <li>Function Block Script Steps Compiled: 15 (310 bytes)</li> <li>Large Function Block Script Steps Compiled: 10 (180 bytes)</li> <li>Function Block Capture Static Filters Compiled: 3 (24 bytes)</li> <li>Function Block Size: 63 (504 bytes)</li> <li>Function Block Size: 53 (504 bytes)</li> <li>Function Block Size: 63 (504 bytes)</li> <li>Argument Stack Size: 63 (504 bytes)</li> <li>Exection Block Size: 63 (504 bytes)</li> </ul> |                |  |  |  |  |  |  |
| Expression Stack Size: 188 (1504 bytes)  | >              |  |  |  |  |  |  |
| Compile CoreMini<br>CoreMini Compiled with Warnings at UTC 2014/04/30 05:50:28. Compile  | oy to Clipboar |  |  |  |  |  |  |
| Device Configuration and Download  |                |  |  |  |  |  |  |
| neoVI (USB) neoECU (CAN) RS232/UART/Bluetooth Configure Device (Bit Rates, Initial Pin States, Resistor Enables, etc): Here Hardware Setup Download CoreMini :   |                |  |  |  |  |  |  |
| Device neoVI FIRE 52567 V Storage SD Card V  |                |  |  |  |  |  |  |
| Send Extract Clear Run CoreMini After Download Advanced Settin   | ngs            |  |  |  |  |  |  |

Figure 3: Settings for complied Wivi file

**Note:** Refer Script1.vs3zip file for details.you can Create .wivi file and upload in to device. Check output with simulator (<u>Simulator.vs3zip</u> file).

2.2 Script 2: Log selected messages on condition. Sleep condition active. Upload data to server.

#### 2.2.1 Objective:

- To start data logging as on particular message comes on bus and stop logging after 1min.
- Collect selected messages.
- Device should go in sleep mode if no message present on bus.
- Upload collection on server before sleep mode.

#### 2.2.2 Procedure:

Launch Vspy and create new Platform or select existing from current platform list.

(Please refer the <u>Script 2.vs3zip</u> file)

- **2.2.2.1** Upload DBC file:
  - Select 'Setup' tab  $\rightarrow$  Network database.
  - Select HS CAN from Networks column on the left.
  - Add .dbc file to this network, from a tab 'Add' on the right.
  - Save the Platform Changes.
  - Same way you can add DBC and VS3 database for other network.
- **2.2.2.2** Copy messages from database to receive message list:
  - Go in Spy networks → Messages Editor. It will open Message Editor Window.
  - Select Database tab and Copy message from database message list.
  - Select Receive tab and paste. It will paste message in receive messages list.

- 2.2.2.3 Create Application signal as a timer
  - Go in Scripting and automation  $\rightarrow$  Application signal.
  - Click on '+' button (Figure 4 (1)). It will create new application signal.
  - Write description 'Timer' to application signal.
  - Select signal type as analog.
  - Select 'This value is timer' (Figure 4 (2)).
  - Select Timer type as a 'Down Timer' and Time Resolution to 'millisecond'

| 🙀 Application Signals 🖾 🚸 VehicleScape DAQ 🔀 | 😝 Network Databases 🖾   |  |
|--|---|--|
|  | Description   |  |
| Filter                                       | Timer   |  |
| Timer (sig0)                                 | Signal Type Analog V Format V                                       |  |
|  | Units Min 0.0000000 Max 0.0000000 .                                 |  |
|  | The Application Signal is an array Array Size 8                     |  |
|  | Value Type  |  |
|  | $\bigcirc$ This value is used as a general purpose variable         |  |
|  | Initial Value 0   |  |
|  | O This value is calculated on an interval                           |  |
|  | Interval (sec) 0.01 Setup   |  |
|  | This value is a timer     Timer Tune     Recelution                 |  |
|  | Down Timer vy millissond  |  |
|  |   |  |
|  | Enable Text API Save And Restore Panels Handlers                    |  |
|  | Persistent Signal - Will be saved to SD Card when used in CoreMini. |  |
|  | Never optimize signal out even if unused                            |  |

Figure 4: Application signal setting as a Down timer

#### **2.2.2.4** Configure VehicleScape DAQ:

- 1. Select Measurements  $\rightarrow$  VehicleScape DAQ
- 2. Select Channels:

Here we can select particular messages which you have to log.

- Go in VehicleScape DAQ  $\rightarrow$  channels.
- Double click on messages from message list at left side of window.
- It will display selected messages on 'Selected channels for test' block (Figure 5(1)).
- Keep polling setup as default.

| Database/Hardware Se | etup Channels PC Logging Stand | dalone Logging G | Gateway Online DAQ 1  |   |
|----------------------|--------------------------------|------------------|---|---|
| Search               | Daeta liet Calacted >>         | V Clear          | Polling Setup         When polling for items, at what rate should we send requests? DTC Request Rate (ms)         (In milliseconds, smaller is faster)         Low 500       Normal         150       High         0       2000         Mask       Set Export Order |   |
|                      | - usice list                   | 0                | Selected Channels for Test  |   |
| Signal               | Source                         | Protocol         | Signal Priority Source 🔺 Type Network Condition Proto Expor Alarma  | s |
| EMS_Message_         | HS CAN (\$100)                 | Normal Mode . 🔺  | A 🚰 EMS_Message_ HS CAN (\$100) Message HS CAN Always Norm 1  |   |
| ngSpd 👯              | HS CAN EMS_Message_ (\$100)    | Normal Mode .    | TCU_Message HS CAN (\$200) Message HS CAN Always Norm 2   |   |
| 👫 Gear_status        | HS CAN EMS_Message_ (\$100)    | Normal Mode .    | MBFM_Message HS CAN (\$300) Message HS CAN Always Norm 3  |   |
| ✓ TCU_Message        | HS CAN (\$200)                 | Normal Mode .    |   |   |
| Signal_0             | HS CAN TCUMessage (\$200)      | Normal Mode .    |   |   |
| ✓ MBFM_Message       | HS CAN (\$300)                 | Normal Mode .    |   |   |
| Signal_0             | HS CAN MBFM_Message (\$300)    | Normal Mode .    |   |   |
| " IC_Message         | HS CAN (\$400)                 | Normal Mode .    |   |   |
| Signal_0             | HS CAN IC_Message (\$400)      | Normal Mode .    |   |   |
| P RPAS_Message       | HS CAN (\$500)                 | Normal Mode .    |   |   |
| Signal_0             | HS CAN RPAS_Message (\$500)    | Normal Mode .    |   |   |

Figure 5: Selected messages.

#### 3. Standalone Logging:

| enselt in a line e se inte  | Channels PC Lo   | gging Standalone Logg  | ing Gateway Or  | nine DAQ 1   |  |
|---|--|--|---|--|--|
| •'+• Messages   | a. Histogram   | Bus Query  |   |  |  |
| etup the format of the  | fle name. This is al   | so your collection nam   | ie.   |  | -  |
| Entire Bus Collection   | >0   |  | Append  | Time and Date to file r  |  |
| ssage Collection Option   | ≥3   |  |   |  |  |
| Which messages should<br>specific messages. Y   | we collect? Select   | "Entire Bus" to capture<br>e messages on the "Chan   | all messages. Ch  | oose "Selected Chann   | to only capture  |
| Entre Bus   | 4  | O Selected Channe  | els   | Ad   | Ivanced Options  |
| lection Start Options   | >  |  |   |  |  |
| When should we start o  | ollecting? Select "In  | nmediately" to start wh  | en the logger is  | powered. Select "Whe   | n Expression is True" for  |
| conditional start. S  | Select " <u>Using Trigge</u>   | " is used to capture be  | efore and after a   | n event.   |  |
| O Immediately   |  | f=   |   |  |  |
| When Expression   | is True  | OR neoVI M   | IC button press   |  |  |
| O Using Trigger Exp   | ression  | OR neoVI N   | NOTE / Push Button  | Pendant on MISC 5 bu   | tton press   |
|   |  |  |   | start succession between   | ris a DAO item )   |
| should we always coll   | ect the DAQ cha  | nnels even if we're not i  | ogging? (Needed if  | start expression/trigge  | 10 0 DAG NOW   |
| Should we <b>always coll</b><br>Always DAQ  | lect the DAQ cha   | nnels even if we're not i  | ogging? (Needed if  | start expression/digge   | to o ong nemy  |
| Should we <b>always coll</b>  | lect the DAQ cha   | nnels even if we're not i  | ogging? (Needed if  | start expression/eigge   | o o o wię nenky  |
| Should we <b>always coll</b><br>Always DAQ<br>lection Configuration   | lect the DAQ cha   | nnels even if we're not i  | ogging? (Needed if  | start expression/orgge   | is a pregnancy   |
| Should we <b>always coll</b><br>Always DAQ<br>lection Configuration<br>How should we finish e   | ect the DAQ cha  | nnels even if we're not i<br>oose <u>"Finish After # of Me</u>   | ogging? (Needed if  | sch collection once we   | 've collected a  |
| hould we always coll<br>Always DAQ<br>lection Configuration<br>How should we finish e<br>specified number o   | ect the DAQ cha  | nnels even if we're not i<br>oose " <u>Finish After # of Mr</u><br>pick " <u>Finish on Expression</u>  | ogging? (Needed if<br><u>cssages</u> " to finish e<br>if to finish collecting   | sch collection once we<br>when an expressio  | 've collected a  |
| hould we always coll<br>Always DAQ<br>lecton Configuration<br>How should we finish e<br>specified number o<br>Finish After # o  | ect the DAQ cha  | nnels even if we're not i<br>oose <u>"Finish After # of My</u><br>pick <u>"Finish on Expression</u><br>O Fin   | ogging? (Needed if<br>essages" to finish e<br>1" to finish collecting<br>ish on Expression  | start expression/origge<br>sch collection once we<br>when an expressio   | 've collected a n is true.   |
| Should we always coll<br>Always DAQ<br>lection Configuration<br>How should we finish e<br>specified number o<br>Finish After # o<br># of Messages   | Act the DAQ cha<br>act collection? Cho<br>f messages. Or, 1<br>f Messages<br>50000 V   | nnels even if we're not i oose <u>"Finish After # of My</u> pick <u>"Finish on Expression</u> OFin ~3.05176MB  | ogging? (Needed if<br><u>essages</u> " to finish e<br><sub>1</sub> " to finish collecting<br>ish on Expression  | start expression/origge<br>sch collection once we<br>when an expressio   | 've collected a n is true.   |
| Should we always coll<br>Always DAQ<br>lection Configuration<br>How should we finish e<br>specified number o<br>Finish After # o<br># of Messages   | Act the DAQ cha<br>ach collection? Cho<br>f messages. Or, 1<br>fMessages<br>50000 v  | nnels even if we're not i<br>oose <u>"Finish After # of My</u><br>pick <u>"Finish on Expression</u><br>  | ogging? (Needed if<br><u>assages</u> " to finish e<br>ish on Expression   | start expression/origge<br>sch collection once we<br>when an expressio   | 've collected a n is true.   |
| Should we always coll<br>Always DAQ<br>lecton Configuration<br>How should we finish e<br>specified number o<br>Finish After # o<br># of Messages<br>start Options   | ect the DAQ cha<br>sach collection? Cho<br>f messages. Or, 1<br>f Messages<br>50000 v<br>messages, what sh   | nnels even if we're not i<br>ose <u>"Finish After # of Me</u><br>pick <u>"Finish on Expression</u><br>   | ogging? (Needed if<br><u>essages</u> " to finish e<br>" to finish collecting<br>ish on Expression   | start expression/shage<br>sch collection once we<br>when an expressio<br>Collection Upload (wirel                              | 've collected a<br>n is true.  |
| Should we always coll<br>Always DAQ<br>lecton Configuration<br>How should we finish e<br>specified number o<br>Finish After # o<br># of Messages<br>start Options<br>specified source source of<br>apture. Select Telesta<br>specified Select Telesta | Act the DAQ cha<br>ach collection? Che<br>f messages. Or, 1<br>f Messages<br>50000 v<br>messages, what sh<br>et? to start captu<br>force Restart to re | nnels even if we're not i<br>ose <u>"Finish After # of Me</u><br>pick <u>"Finish on Expression</u><br>OFin<br>-3.05176MB //<br>puld we do? Select <u>"Stop</u> "<br>uring again (will wait fi<br>estart regardless of th | ogging? (Needed if<br><u>cssages</u> " to finish e<br>" to finish collecting<br>ish on Expression<br>ish on Expression<br>to end the<br>or start<br>start | ach collection once we<br>when an expressio<br>Collection Upload (wirel<br>At the end of the collect<br>Wireless neoVI? Select | 've collected a<br>n is true.<br>ess only)<br>ton, upload the data to<br>both for fastest network. |

Figure 6.a: Settings for Script 2

- Go in VehicleScape DAQ  $\rightarrow$  Standalone Logging  $\rightarrow$  Messages.
- Collection name: Write a Collection name in collection name box (Figure 6.a (1)) (Default name will be Collection 1) and check the box on its side to append Date & Time to the file name (Figure 6.a (2)).So it will create collection by selected name with date and time.
- Message collection option (Figure 6.a(3)): Select 'Selected channels' option. So it will log only that channels which is selected in channel tab.
- Collection start option (Figure 6.a (4)):
  - Select 'when Expression is true'
  - Click on *f* tab. It will open 'Select start Expression' window.

- Click on 'Rx Messages' (Figure 7 (1)) and select 'Present' property (Figure 7 (2)) from property list then double click on 'TCU\_message'(Figure 7 (3)). It will come in Expression block on top of window (Figure 5. (4)).
- Click on OK option

| <b>=</b>                   | Se   | elect Start Expression |                | - 🗆 🗙                     |
|----------------------------|--|------------------------|----------------|---------------------------|
| Enter Expression for sign  | al   |                        |                | Help OK Cancel            |
| Description and the        | 0  |                        | Custom         | Format                    |
| Description TCU_Message (F | <sup>Present</sup> ) (4)                   |                        | Format         | V Min                     |
| Expression {TCU_Message    | (Present) :in1-0}                          |                        | Units          | Max                       |
| Clear                      | Evaluate as text                           |                        |                | Discrete Values           |
| f* Expression Builder      | $\cap$                                     |                        |                |                           |
| •Rx Messages               | Fort By: Networks                          | Add To Expression      | Add Operator   | Calculator Panel          |
| 🖲 Database                 |  |                        |                | ^ / * -                   |
| Tx Messages                | Find                                       | Clear                  |                |                           |
| Signal Groups              | □ 굮 HS CAN                                 | <u>^</u>               | >              | / 3 5 +                   |
| ₽DAQ                       | Ems_Message_ (100)     P→ EmsSad (Applace) |                        | <              | 4 5 6                     |
| @Jobs                      | Gear status (Analog)                       |                        |                | 1 2 3 )                   |
| Carter Signals             | □ ••• IC_Message (400)                     |                        | >=             | 0(                        |
| Retworks                   | Signal_0 (Analog)                          |                        | >>             |                           |
| <b>P</b> -Nodes            | □ •••• MBFM_Message (300)                  |                        | <<             | *** Application Signals   |
| "# Misc                    | Signal_0 (Analog)                          |                        | bit1(          | Name                      |
| Function Blocks            | 3  |                        | bit2(          |                           |
| Physical IO                | E-=\* TCU_Message (200)                    |                        | bit3(          | Add                       |
| Logger                     | Signal 0 (Analog)                          | ×                      | bit4(<br>bit5( |                           |
|                            | Properties                                 |                        | bit6(          | Hide auto-generated items |
|                            | Present                                    | <u>^</u>               | bit7(          | Test                      |
|                            | Present Toggle                             |                        | byte0(         | Result:                   |
|                            | Update Rate (abs)                          |                        | byte2(         |                           |
|                            | Per Second<br>Count                        |                        | byte3(         |                           |
|                            | Change Count                               |                        | one( 🗸         |                           |
|                            | Start Time                                 | *                      | · · ·          |                           |

Figure 7: Expression Builder to give an expression for start on expression

- Collection configuration (Figure 6.a (5)):
  - Select 'Finish on expression' option.
  - Click on *f*<sup>\*</sup> tab. It will open 'Select start Expression' window.
  - Click on 'App signals' and double click on 'Timer' application signal. it will come in Expression box on top of window.
  - Write {Timer :sig0-index(0)}=1 in expression box.
  - Click on OK option.
  - So here we will set timer value in function block and timer is down timer so it will start decreasing. So logger will stop collecting messages after Timer value reach to 1.

| Reporting Options<br>How would you like feed<br>Select " <u>neoVI MOTE</u> " to                           | back about the status of the logger? Sele<br>use the neoVI MOTE to display status a  | ct " <u>LED</u> " to <b>flash the LEDs</b> based o<br>and manually trigger.  | n the status of the capture.  |
|---|--|--|---|
| LEDs  | neoVI MOTE Beep on Wak   | eup (PLASMA only)  |   |
| When the logger is run<br>slowly. When capturin   | nning, the <b>red</b> LED will blink at a fast rate.<br>ng post-trigger data, the <b>green</b> LED will fla  | When data is captured to the SD ca<br>ish quickly.   | ard, the green LED will flash   |
| Power Management  | 9  |  |   |
| When should the neoVI<br><u>Present</u> " to go to sleep<br>an expression is true<br>cause the neoVI to s | go into low-current draw (sleep) mode? Se<br>once a message is not on the bus af<br>for a period of time and " <u>On No Bus Activ</u><br>tay awake or wake up!.  | lect " <u>Never</u> " to <b>always stay awak</b><br>ter a certain period of time. Likewise<br><u>ty</u> " will sleep <b>once the bus is quie</b> | e. Select <u>"When Message Not</u><br>, " <u>On Expression</u> " will sleep <b>once</b><br>et. Note: <b>Any bus activity will</b> |
| Never   | <ul> <li>When Message Not Present</li> </ul>   | On Expression  | On No Bus Activity  |
| Vehicle Spy can generat<br>currently have loaded. I<br>to SD card" or just "Save                          | e a database that will <b>decode all the da</b><br><b>If you want this</b> , check " <u>Generate decod</u><br><u>e to local file system</u> ". Then, click " <u>Generat</u><br>e decoding database<br>O Save to local<br>O Save to local | ta logged by this setup. It does thi<br><u>inq database</u> " and then choose if yo<br><u>e</u> " and go!<br>SD card<br>al file system           | is by combining the databases you<br>u'd like the database to " <u>Transfer</u><br>11   |
| Enable v  | ideo logging   | (12) Advance   | e for Wireless Devices d Options Clean  |
|   |  |  |   |

Figure 6.b: Settings for Script 2

- Restart option (Figure 6.a (6)): Select 'Restart' option. So it will restart logging after completion of collection. It will again check for start collection expression for restart logging.
- Collection upload (Figure 6.a (7)): Check box of 3G and WiFi option as per availability or select both.
- Reporting option (Figure 6.b (8)): Check box of 'LED' option. It will give process status on LEDs of device. Also check box of 'Beep on wakeup (PLASMA only)' option. So it will give beep when device will wakeup.
- Power management (Figure 6.b (9)):
  - Select 'On no bus activity' option. So device will go in sleep mode when no message present on bus.
  - Wake mode: Select 'Normal' option.
  - Check box of start new file on waking up.
- Generate (Figure 6.b (10)): Select 'Generate decoding database',
   'Transfer to SD card' and check box of 'generate for wireless device'(Figure 6.b (11))

2.2.2.5 Create Function block:

- Go in scripting and automation  $\rightarrow$  Function blocks.

| 🍄 Ap     | 🍄 Application Signals 🔯 🚸 VehicleScape DAQ 🔯 🚍 Function Blocks 🕰 |                        |                       |               |  |  |  |  |
|----------|--|------------------------|-----------------------|---------------|--|--|--|--|
| $+)\tau$ | +) () & 🖻 💼 🗠 🔗 🗐 🛎 🖬 🔍  |                        |                       |               |  |  |  |  |
| Key      | 🤕 Description 🕜 Type Start Type Running 🕨 🛑 🖍 🖶 Status           |                        |                       |               |  |  |  |  |
|          | _  | Y                      | 7                     | 3 7           |  |  |  |  |
| tst0     | Timer C  | ontrol and Clear state | Script Immediate      | stopped       | Function Block has not started         |  |  |  |
|          | -  |                        |                       |               |  |  |  |  |
|          |  |                        |                       |               |  |  |  |  |
| Script   | Start  |                        | Time                  | r Control and | l Clear state                          |  |  |  |
| + Af     | fter   | 🕈 Before 🗧 🗎 🛍         | 1 No Errors           |               |  |  |  |  |
|          | Step   | Description            | Value                 |               | Comment                                |  |  |  |
|          | 1  |                        |                       |               |  |  |  |  |
|          | 2  | 📑 If                   | {TCUMessage (Pre      | sent) :in1-0} | // Check for message present           |  |  |  |
|          | 3  | -• Set Value           | {Timer :sig0-index(0) | } = 60000     | // Set timer value to 1 min=60000ms    |  |  |  |
|          | 4  | 🖄 Wait For             | 0.100000 sec          |               |  |  |  |  |
|          | 5  | 🗸 🛛 Clear Stats        | TCU_Message           |               | // Clear message preent state from bus |  |  |  |
|          | 6  | 🚰 End If               |                       |               |  |  |  |  |
|          | 7  |                        |                       |               |  |  |  |  |

Figure 8: Function block script for Collection.

- Click on '+' option on left side (Fig. 8 (1)). Select 'script' option.
- It will create new function block. Write description in description box (Fig. 8 (2)).
- Make sure start type is Immediate (Fig. 8 (3)).
- In script,
  - Write logic to check for TCU\_message present.
    - Click on Description column. It will show list of commands.
       Select 'If' Command.
    - Double click on Value column in same row. It will pop up 'Enter Expression' window. Select 'Rx Messages' → 'Present' property from property list and then double click on 'TCU\_message' from message list. So it will display '{TCU\_message (Present): in1-0}'in expression block.
    - Press OK.
    - You can add comment in comment column.
    - So here if condition will check for TCU\_message is present on bus or not.

- Set timer value to 1 min i.e. 60000ms.
  - On next line, same way Select 'set value' from command list.
  - Select app signal in 'Enter expression' window. Set value to 60000.
  - So here as TCU\_Message will come on bus 'If' condition become true and then it will set timer value to 6000ms.
  - Here we have selected Timer application signal as down timer so it will start decreasing.
- Clear stats of 'TCU\_message' present
  - Select 'Clear stats' command in next line. Select 'Clear stats for one message' in 'Select message to clear stats' window. Select TCU\_message in message list.
  - Press Ok.
  - It is used to clear message present status on bus. So when 'if' loop condition will recheck then it will not get previous message present status. So it will check for next TCU\_message present on bus.
- Here we have not used 'Stop' condition at last, so it will continuously run the function block.

#### **2.2.2.6** Create .wivi file:

- To create .wivi file of all configuration click on Here tab (Figure 6.b (12)) on VehicleScape DAQ.
- It will open the 'Coremini Executable Generator' window.
- Make sure that coremini compiled without error (Figure 9 (1)).
- Click on 'Export wivi file' option (Figure 9 (2)). It will export wivi file then Save file in destination folder
- Upload the file to device from wivi server and check output.

| CoreMini Executable Generator   | ×                                 |
|---|-----------------------------------|
| Build SD Card Partition Advanced Settings   | Help                              |
| <ul> <li>Capturing EMS_Message_ (copied from DB)</li> <li>Capturing TCU_Message (copied from DB)</li> <li>Capturing TGU_Message (copied from DB)</li> <li>Using standard filtered circular capture block for Collection 1.</li> <li>Generating message database for reading data files</li> <li>Message database file: C:\IntrepidCS\Vehide Spy 3(60045)\Data Directory\Loggere Scipts\DAQ 1_MessageD</li> <li>File added to SD Card image: C:\IntrepidCS\Vehide Spy 3(60045)\Data Directory\Loggere Scipts\DAQ 1_MessageD</li> <li>Exe Compressed to 5120 bytes (31.4 % of original size)</li> <li>CoreMini Header (270 bytes)</li> <li>Network Objects Compiled: 217 (868 bytes)</li> <li>Function Block Scripts Compiled: 155 (310 bytes)</li> <li>Function Block Script Steps Compiled: 15 (180 bytes)</li> <li>Function Block Captures Compiled: 158 (28 bytes)</li> <li>Function Block Captures Compiled: 3 (24 bytes)</li> <li>Function Block Size: 63 (504 bytes)</li> <li>Function Stack Size: 188 (1504 bytes)</li> </ul> | A<br>Database.vsdl<br>sageDatabas |
| Expression Stack Size: 188 (1504 bytes)   | >                                 |
| Compile CoreMini<br>CoreMini Compiled with Warnings at UTC 2014/04/30 05:50:28. Compile   | py to Clipboar                    |
| Device Configuration and Download   |                                   |
| neoVI (USB) neoECU (CAN) RS232/UART/Bluetooth<br>Configure Device (Bit Rates, Initial Pin States, Resistor Enables, etc): ### Hardware Setup<br>Download CoreMini :<br>Device neoVI FIRE 52567 	V Storage SD Card 	V  |                                   |
| Send Extract Clear Run CoreMini After Download Advanced Settin  | ngs                               |

Figure 9: Settings for complied Wivi file

**Note:** Refer Script2.vs3zip file for details. You can Create .wivi file and upload in to device. Check output with simulator (<u>Simulator.vs3zip</u> file similar to simulator file for Script 1).

#### 2.3 Script 3: Log DTC and DID Response. Update collection on server periodically.

#### 2.3.1 Objective:

- To log DTC and DID response for selected ECU.
- Updated collection on server for every ECU after 1 hour

#### 2.3.2 Procedure:

Launch Vspy and create new Platform or select existing from current platform list. (Please refer the <u>Script 3.vs3zip</u> file)

#### 2.3.2.1 Upload DBC and .gmd file:

- Select 'setup' tab  $\rightarrow$  Network database.
- Select HS CAN from Networks column on the left.
- Add .dbc file to this network, from a tab 'Add' on the right.
- Save the Platform Changes.
- Same way you can add DBC and VS3 database for other network.
- To import .gmd , Select 'Diagnostic' tab and browse the .gmd file.

**2.3.2.2** To Import .odx or .cdd file.

- Select 'setup' tab  $\rightarrow$  ECU diagnostic database.
- Click on 'Import' button (Figure 10 (1)).
- To Import .odx file: Select Import → ISO 14229(UDS) → ODX/PDX
   → Select the ODX file from PC.
- To Import .cdd file:Select Import → KWP 2000 → CDD → select the .cdd file from PC. For this example script 3 we don't have .cdd file.



Figure 10: Settings to Import a .odx or .cdd file

- **2.3.2.3** Create Application signal to trigger collection start for all ECUs.
  - Go in Scripting and automation  $\rightarrow$  Application signal.
  - Click on '+' button (Figure 11(1)). It will create new application signal.
  - Write description to application signal. Here write description as
     'DID and DTC request trigger'
  - Select signal type as analog.
  - Select value type as 'This value is used as a general purpose variable'. Check box Initial value to zero.

| 🔤 New Spy Setup - Vehicle Spy                    |                                       |                                  |   |          | - • ×            |
|--|---------------------------------------|----------------------------------|---|----------|------------------|
| <u>File Setup Spy Networks Measurement Ember</u> | lded Tools Scripting and <u>A</u> uto | mation <u>R</u> un <u>T</u> ools | <u>H</u> elp                            |          |                  |
| Offline 📑 📴 Platform: U                          | 22xMY15_CANSim_v0.6                   | - 🔋 😫 🔧                          | Desktop 1                               |          | 🔍 <u>D</u> ata 🔻 |
| 📽 Application Signals                            |                                       |                                  |   |          | <b>• </b>        |
| + - 🔏 🖻 🖻 🖻                                      | Description                           |                                  |   |          |                  |
| Filter   | DID and DTC request trigg             | er                               |   |          |                  |
| DID and DTC request trigger (sig0)               | Signal Type Analog                    | Format                           | •                                       |          |                  |
|  | Units 2                               | Min 0.0000000                    | Max                                     | 0.000000 |                  |
| U  | The Application Signal i              | s an array Array                 | Size 8                                  |          | •                |
|  | Value Type                            |                                  | ,                                       |          |                  |
|  | This value is used a                  | s a general purpose va           | ariable                                 |          | •                |
|  |                                       | Initial Value                    | 0                                       |          |                  |
|  | C This value is calcula               | ited on an interval              |   |          | •                |
|  |                                       | Interval (sec                    | 0.01                                    | Cotup    |                  |
|  | C This value is a time                | r                                | , | Jemb     | •                |
|  |                                       | Timer Type                       | Timer Resol                             | ution    |                  |
|  |                                       | Up Timer                         | <u> </u>                                | <u>~</u> |                  |
|  | Enable Text API Save A                | nd Restore                       | Panels                                  | Handlers |                  |
|  | 🗖 Persistent Signal - Will            | be saved to SD Card v            | vhen used in CoreMini.                  |          |                  |
|  | Never optimize signal o               | ut even if unused                |   |          |                  |
|  |                                       |                                  |   |          |                  |
|  |                                       |                                  |   |          |                  |
|  |                                       |                                  |   |          |                  |
| • (edit) • (edit)                                | • (edit)                              | • (edit)                         | • (edit)                                | • (edit) | No Bus Errors    |

Figure 11: Application signal for the trigger.

**2.3.2.4** Configure VehicleScape DAQ:

Select Measurements  $\rightarrow$  VehicleScape DAQ  $\rightarrow$  Standalone Logging

- Create 1 more collection by pressing '+' button.

| 🤨 Network Databases 🖾 🐻 ECUs 🔯 🚸 VehicleScape DAQ 🖾 😭 Application Signals 🖾 🚍 Function Blocks 🛽   | 3 💁 Me   |
|---|----------|
| Database/Hardware Setup Channels PC Logging Standalone Logging Gateway Online 🗸 Auto Start PC DAQ   | DAQ 1    |
| 🕫 Vo Messages Histogram   | :<br>5   |
| Setup the format of the file name. This is also your <b>collection name</b> .   | MSIO     |
| EMS Identification  | lentif   |
| Bus Query Options   | icati    |
| The Data control in the second DTD and DTC. for some different COULTY we have detailed a financial different COULTY and the second state of the |          |
| state of ECUs.  | 2. 10    |
| First, select a "Base ECU". The ECU should contain the definitions of the DIDs you'd like to read.  | 티        |
| Base ECU Base ECU   | entifi   |
| Would you like to "Celect ECUs" to read from or simply "Specify a Pappe" of diagnostic IDs?   | catio    |
| Contraction of simply <u>specify a range</u> of diagnosic tose  | <b>S</b> |
| Select ECUs 2 O Specify a Range   |          |
| Select ECUs   |          |
| Which <b>DIDs</b> should we read from each ECU?   |          |
| Select DYD (\$F111) EMS SERIAL NUMBER ♥ (\$F112) EMS VIN NUMBER ♥ (\$F113) EMS SYSTEM   |          |
| Select DIDS SUPPLIER PART NUMBER V (\$F114) EMS MANUFACTURING DATE  |          |
| Should we read DTCs? and iffso, which status mask should be used?   |          |
| Read DTCs Read Extended/Snapshot records Status Mask \$8F   |          |
| Clear DTCs  |          |
| Should we read the OBD-II VIN and Calibration ID from the engine control module? OBD-II PIDs \$09 \$02 and \$09 \$04 will<br>be read.           |          |
| Read OBD-II VIN and Calibration ID  |          |
|   |          |

Figure 12.a: VehicleSpy configuration for Bus Query

- Write Collection name as 'EMS Identification' for first collection and 'TCU Identification' for second collection and 'TCU Identification'. Check the box on its side to append Date & Time to the file name.
- Bus Query option:
  - Base ECU (Figure 12.a (1)): Select base ECU which contains the definition of DID's. Here select EMS ECU for first collection and TCU ECU for second collection.

- Select ECU (Figure 12.a (2)): Select ECU To which have to send DID request. Here select EMS ECU for first collection. TCU ECU for second collection.
- Select DID (Figure 12.a (3)): Select DID requests from DID list of ECU.
- Check box of 'Read DTCs' for both collections.

#### AN-ICSI-1008

| 🐱 New Spy Setup - Vehicle Spy  | • X    |
|--|--------|
| <u>File</u> <u>Setup</u> Spy <u>N</u> etworks <u>M</u> easurement <u>E</u> mbedded Tools Scripting and <u>A</u> utomation <u>Run</u> <u>T</u> ools <u>H</u> elp  |        |
| 🚺 🗸 Offline 📰 📴 Platform: U22xMY15_CANSim_v0.6 🔽 🧃 🍓 🔧 🕩   | Data 🔻 |
| 🙄 Application Signals 🔯 🚸 VehicleScape DAQ 🔀   | 6      |
| Database/Hardware Setup Channels PC Logging Standalone Logging Gateway Online DAQ 1  |        |
| Collection Start Options   | · ·    |
| When should we start collecting? Select " <u>Immediately</u> " to <b>start when the logger is powered</b> . Select " <u>When Expression is True</u> "<br>for a <b>conditional start</b> . Select " <u>Using Trigger</u> " is used to <b>capture before and after an event</b> .  |        |
|  |        |
| C Using Trigger  | •      |
| How should the above expression be used to <b>determine a trigger event</b> ? Select " <u>When expression is true</u> " to trigger when the expression changes from zero to non-zero. Otherwise, choose one of the derivative options.   | -      |
| ○ Trigger when expression is true  |        |
| C Trigger when expression changes Intervention Provided Intervention Provided Intervention Intervention Intervention Intervention  |        |
| C Trigger when expression increases v by more than v 1   |        |
| Should we always collect the DAQ channels even if we're not logging? (Needed if start expression/trigger is a DAQ item.)   |        |
| Collection Configuration   |        |
| What kind of collection should we use for this trigger? Select " <u>Pre/Post Collection</u> " for a collection of the <b>vehicle bus before</b><br>and after the trigger event. Select " <u>One-shot Report</u> " for a <b>report of the values</b> of all the channels at the time of the<br>event  | ·      |
| C Pre/Post Collection 5 One-shot Report  |        |
| How long after the trigger event should we wait before actually collecting the one-shot?   |        |
| After trigger delay (s) 0.0  | -   ·  |
| Restart Options Collection Upload (wireless only)  |        |
| Select "Restart" to start capturing again.<br>At the end of the collection, upload the data to<br>Wireless neoVI? Select both for fastest network.   | •      |
| C Stop 6 C Restart □ 3G □ WFI  |        |
| Reporting Options  |        |
| How would you like feedback about the status of the logger? Select "LED" to <b>flash the LEDs</b> based on the status of the capture.<br>Select " <u>neoVI MOTE</u> " to <b>use the neoVI MOTE</b> to display status and manually trigger. Select " <u>Wireless NeoVI</u> " to choose signals that<br>will display on the Wireless NeoVI website | •      |
| (edit)     (edit)     (edit)     (edit)     No Bus   | Errors |

Figure 12.b: VehicleSpy configuration for collection options

- Collection start option (Figure 12.b (1)):
  - Select 'Using Trigger Expression' option(Figure 12.b (2)).
  - Click on *f* tab(Figure 12.b (3)). It will open 'Select start Expression' window.
  - Click on 'App signals' and double click on 'DID and DTC request trigger' application signal. It will come in Expression box on top of window.
  - Set value '{DID & DTC request trigger :sig0-index(0)}=1' in expression box for first collection. Click on OK option. Select 'Trigger when expression is true'

- Same way set value '{DID & DTC request trigger :sig0index(0)}=2' for second collection.
- It will start collection for 'EMS Identification' when application signal value becomes 1 and for 'TCU Identification' when application signal value becomes 2.
- Collection configuration (Figure 12.b (4)): Select 'One shot report' for all collections. (Figure 12.b (5))
- Restart option (Figure 12.b(6)): Select 'Restart' option for all collections. So it will restart logging after completion of collection. It will again check for start collection expression for restart logging.
- Collection Upload (Figure 12.b (7)): Check box of 3G and WiFi option as per availability or select both for all collections.

| Reporting Options 8<br>How would you like feedback about the status of the logger? Select "LED" to <b>flash the LEDs</b> based on the status of the capture.<br>Select " <u>neoVI MOTE</u> " to <b>use the neoVI MOTE</b> to display status and manually trigger.   |
|---|
| LEDS neoVI MOTE Beep on Wakeup (PLASMA only)  |
| When the logger is running, the <b>red</b> LED will blink at a fast rate. When data is captured to the SD card, the <b>green</b> LED will flash slowly. When capturing post-trigger data, the <b>green</b> LED will flash quickly.  |
| Power Management Disable all logger TX'ing on this: free Expression When should the neoVI go into low-current draw (sleep) mode? Select "Never" to always stay awake. Select "When Message Not. Present" to go to sleep once a message is not on the bus after a certain period of time. Likewise, "On Expression" will sleep once an expression is true for a period of time and "On No Bus Activity" will sleep once the bus is quiet. Note: Any bus activity will cause the peoVI to stay awake or wake or use und   |
| Never     When Message Not Present     On Expression     On No Bus Activity     Generate     Vehicle Spy can generate a database that will <b>decode all the data logged</b> by this setup. It does this by combining the databases you     currently have loaded. If you want this, check "Generate decoding database" and then choose if you'd like the database to " <u>Transfer</u> to SD card" or just "Save to local file system". Then, click "Generate" and go!   |
| ✓ Generate decoding database <ul> <li>Transfer to SD card</li> <li>Save to local file system</li> <li>✓ Generate for Wireless Devices</li> <li>12</li> <li>Advanced Options</li> <li>Clean</li> </ul> <ul> <li>Advanced Options</li> <li>Clean</li> </ul> <ul> <li>Image: Clean</li> <li>Image: Clean</li></ul> |
|   |

Figure 12.c: VehicleSpy configuration for Reporting and Generation

- Reporting option (Figure 1(8)): Check box of 'LED' option. It will give process status on LEDs of device.
- Power management (Figure 1 (6)): select 'Never' option.

- Here we have to send DID and DTC request to both ECUs after every 1 hour. So Need to write script to control DID and DTC requests
- **2.3.2.5** Create Function block:
  - Go in scripting and automation  $\rightarrow$  Function blocks.
  - Click on '+' option on left side (Fig. 7 (1)). Select 'script' option.
  - It will create new function block. Write description in description box as 'DID and DTC Trigger Control' (Fig. 7 (2))
  - Make sure start type is 'Immediate' (Fig. 7 (3)).
  - In script(Refer Figure 10),
    - Set 'DID and DTC request trigger' application signal value to 1(refer script 2 for set value command). So 'Start collection condition' for first collection will satisfy. So device will send DID and DTC request on bus.
    - Same way set 'DID and DTC request trigger' application signal value to 2 for other collection.
    - Set 'Wait for' command to 1 hour (3600 sec).
    - Then it will again jump on first step. So process will be continuing after every one hour.

| Script | Start |             | DID and DTC Trigger                              | control                           |
|--------|-------|-------------|--|-----------------------------------|
| + Af   | fter  | 🕈 Before 💻  | 🗈 🛍 🛛 🗊 No Errors                                |                                   |
|        | Step  | Description | Value  | Comment                           |
|        | 1     | - Set Value | {DID and DTC request trigger :sig0-index(0)} = 1 | // Trigger for EMS Identification |
|        | 2     | 🖄 Wait For  | 1.000000 sec                                     | // Delay                          |
|        | 3     | - Set Value | {DID and DTC request trigger :sig0-index(0)} = 2 | // Trigger for TCU Identification |
|        | 4     | 🖄 Wait For  | 3600.000000 sec                                  | // Wait for 1 Hrs                 |
|        | 5     | 🚺 Jump To   | Step 1   |                                   |
|        | 6     |             |  |                                   |

Figure 13. Function Block for the triggers.

#### **2.3.2.6** Create wivi file:

- To create .wivi file of all configuration click on Generate tab (Figure 12.c (12)) on VehicleScape DAQ.
- It will open the 'Coremini Executable Generator' window.
- Make sure that coremini compiled without error (Figure 14 (1)).
- Click on 'Export wivi file' option (Figure 14 (2)). It will export wivi file then Save file in destination folder
- Upload the file to device from wivi server and check output.
- Device will send request for DID and DTC for both ECU and it will upload on server after every one hour.

| CoreMini Executable Generator  | ×         |
|--|-----------|
| Build SD Card Partition Advanced Settings Help   |           |
| Capturing EMS_Message_(copied from DB) Capturing TCU_Message (copied from DB) Capturing TCU_Message (copied from DB) Capturing MBFM_Message (copied from DB) Capturing MBFM_Message (copied from DB) Using standard filtered circular capture block for Collection 1. Generating message database for reading data files Message database file: C:\IntrepidCS\Vehide Spy 3(60045)\Data Directory\Loggere Scipts\DAQ 1_MessageDatabase.v File added to SD Card image: C:\IntrepidCS\Vehide Spy 3(60045)\Data Directory\Loggere Scipts\DAQ 1_MessageDatabase.v Compressed to 5120 bytes (31.4 % of original size) CoreMini Header (270 bytes) CoreMini Header (270 bytes) Function Block Scripts Compiled: 1217 (868 bytes) Function Block Scripts Steps Compiled: 15 (310 bytes) Function Block Scripts Steps Compiled: 15 (310 bytes) Function Block Scripts Steps Compiled: 15 (310 bytes) Function Block Captures Compiled: 168 bytes) Function Block Capture Scompiled: 168 bytes) Function Block Capture Scompiled: 168 bytes) Function Block Captures Scompiled: 169 bytes) Compile CoreMini CoreMini Compiled with Warnings at UTC 2014/04/30 05:50:28. Compile Wireless neoVI | sdl<br>as |
| Export a .wivi package for use with Wireless neoVI Export WiVI Fil   | :         |
| Device Configuration and Download  |           |
| neoVI (USB)       neoECU (CAN)       RS232/UART/Bluetooth         Configure Device (Bit Rates, Initial Pin States, Resistor Enables, etc):       Imp Hardware Setup         Download CoreMini :       Device       neoVI FIRE 52567         Device       neoVI FIRE 52567       Storage       SD Card         Send       Extract       Clear       Imp Run CoreMini After Download       Advanced Settings   |           |
| Connected to device. SD Card: 1.83 GB  |           |

Figure 14: Settings for complied Wivi file

**Note:** Refer Script3.vs3zip file for details. You can Create .wivi file and upload in to device. Check output with simulator (<u>Simulator.vs3zip</u> file similar to simulator file for Script 1).

#### 2.4 Script 4: Log selected messages before and after a trigger

#### 2.4.1 Objective:

- To start data logging as on EngSpd signal value becomes 150 Rpm.
- Collect selected messages.
- Collect 200 messages before and 200 messages after trigger.

#### 2.4.2 Procedure:

Launch Vspy and create new Platform or select existing from current platform list. (Please refer the <u>Script 4.vs3zip</u> file)

#### **2.4.2.1** Upload DBC file:

- Select 'Setup' tab  $\rightarrow$  Network database.
- Select HS CAN from Networks column on the left.
- Add .dbc file to this network, from a tab 'Add' on the right.
- Save the Platform Changes.
- Same way you can add DBC and VS3 database for other network.

**2.4.2.2** Copy messages from database to receive message list:

- Go in Spy networks → Messages Editor. It will open Message Editor Window.
- Select Database tab and Copy message from database message list.
- Select <u>receive</u> tab and paste. It will paste message in receive messages list.

#### **2.4.2.3** Configure VehicleScape DAQ:

Select Measurements  $\rightarrow$  VehicleScape DAQ

Select Channels:

Here we can select particular messages which you have to log.

- Go in VehicleScape DAQ  $\rightarrow$  channels.
- Double click on messages from message list at left side of window.
- It will display selected messages on 'Selected channels for test' block (Figure 15 (1))
- Keep polling setup as default.

| Database/Hardware S | Setup Channels PC Logging Stand | dalone Logging G | Gateway Online DAQ 1  |   |
|---------------------|---------------------------------|------------------|---|---|
| Search              |                                 |                  | Polling Setup         When polling for items, at what rate should we send requests? DTC Request Rate (ms)         (In milliseconds, smaller is faster)         Low 500       Normal         150       High         0       2000         Mask       Set Export Order |   |
|                     |                                 |                  | Selected Channels for Test  |   |
| Signal              | Source                          | Protocol         | Signal Priority Source 🔺 Type Network Condition Proto Expor Alarm   | s |
| EMS_Message_        | HS CAN (\$100)                  | Normal Mode . 🛧  | EMS_MessageHS CAN (\$100) Message HS CAN Always Norm 1  |   |
| 👯 EngSpd            | HS CAN EMS_Message_ (\$100)     | Normal Mode .    | TCU_Message HS CAN (\$200) Message HS CAN Always Norm 2   |   |
| 👫 Gear_status       | HS CAN EMS_Message_ (\$100)     | Normal Mode .    | MBFM_Message HS CAN (\$300) Message HS CAN Always Norm 3  |   |
| ✓ TCU_Message       | HS CAN (\$200)                  | Normal Mode .    |   |   |
| 🏹 Signal_0          | HS CAN TCUMessage (\$200)       | Normal Mode .    |   |   |
| MBFM_Message        | HS CAN (\$300)                  | Normal Mode .    |   |   |
| 🏷 Signal_0          | HS CAN MBFM_Message (\$300)     | Normal Mode .    |   |   |
| " IC_Message        | HS CAN (\$400)                  | Normal Mode .    |   |   |
| Signal_0            | HS CAN IC_Message (\$400)       | Normal Mode .    |   |   |
| "" RPAS_Message     | HS CAN (\$500)                  | Normal Mode .    |   |   |
| Signal_0            | HS CAN RPAS_Message (\$500)     | Normal Mode .    |   |   |

Figure 15: Selected messages.

Standalone Logging:

Go in VehicleScape DAQ  $\rightarrow$  Standalone Logging  $\rightarrow$  Messages.

| 🚾 Script 4.vs3zip - Vehicle Spy  |
|--|
| <u>File</u> <u>Setup</u> Spy <u>N</u> etworks <u>M</u> easurement <u>E</u> mbedded Tools Scripting and <u>A</u> utomation <u>R</u> un <u>T</u> ools <u>H</u> elp   |
| Offline 📰 📴 Platform: Internal VS3ZIP (Script4) 🔽 🧃 🔩 🔧 💭  |
| 🚸 VehicleScape DAQ 🖾 💀 Messages Editor 🖾 🧃 Network Databases 🖾 @ Messages 🖾  |
| Database/Hardware Setup Channels PC Logging Standalone Logging Gateway Online DAQ 1  |
| Messages Histogram   |
| Setup the format of the file name. This is also your collection name.  |
| EngSpd Signal trigger  |
| Message Collection Options   |
| Which messages should we collect? Select "Entire Bus" to capture all messages. Choose "Selected Messages" to only capture specific messages. You can select these messages on the "Channels" tab.  |
| C Entire Bus 2 C Selected Messages   |
| whenever they are on the bus).   |
| Enable decimation (from Channels list priority column)   |
| Store on change only (based on entire raw messsage)  |
| Collection Start Options   |
| When should we start collecting? Select " <u>Immediately</u> " to <b>start when the logger is powered</b> . Select " <u>When Expression is True</u> " for a <b>conditional start</b> . Select " <u>Using Trigger</u> " is used to <b>capture before and after an event</b> .   |
| C Immediately  |
| When Expression is True     Image: True       Image: Using Trigger     Image: True         Image: True     Image: True |
| How should the above expression be used to <b>determine a trigger event</b> ? Select " <u>When expression is true</u> " to trigger when the expression changes from zero to non-zero. Otherwise, choose one of the derivative options.   |
| ○ Trigger when expression is true  for at least 1.0  seconds   |
| C Trigger when expression changes 🗌 neoVI MOTE / Push Button Pendant on MISC 5   |
| C Trigger when expression increases v by more than 1   |
| Should we always collect the DAQ channels even if we're not logging? (Needed if start expression/trigger is a DAQ item.)   |

Figure 16.a Configuration for collection for data logging as on EngSpd signal value

- Collection name: Write a Collection name 'EngSpd signal Trigger' in collection name box (Figure 16.a (1)) (Default name will be Collection 1) and check the box on its side to append Date & Time to the file name.So it will create collection by selected name with date and time.
- Message collection option (Figure 16.a (2)): Select 'Selected channels' option. So it will log only that channels which is selected in channel tab.

- Collection start option:
  - Select 'Using Trigger Expression' (Figure 16.a (3)): Click on fx
     tab. It will open 'Select start Expression' window. (Figure 16.a (4))
  - Click on 'Rx Messages' (Figure 17 (1)) and select 'Value' property (Figure 17 (2)) from property list then double click on 'EngSpd' Signal (Figure 17(3)). It will come in Expression block on top of window then set expression as '{EngSpd (Value) :in14-sig0-0}=150' (Figure 167(4)).
  - Click on OK option.
  - Select 'Trigger when expression is true'

| <b>Ξ</b>                     |                                  | Select Start Expression               |              | - 🗆 🗙   |
|------------------------------|----------------------------------|---------------------------------------|--------------|---|
| Enter Expression for signa   | al                               |                                       |              | Help OK Cancel                                |
| Description England (value)  | <u> </u>                         |                                       |              | m Format                                      |
| Erigspu (value)              | (4)                              |                                       | Forma        | at 🗸 Min                                      |
| Expression {EngSpd (Value) : | in 14-sig0-0}=150                |                                       | Units        | Max   |
| Clear                        | Evaluate as text                 |                                       |              | Discrete Values                               |
| f* Expression Builder        | <u> </u>                         |                                       |              |   |
| •••Rx Messages               | Solt By: Networks 🗸              | Add To Expression                     | Add Operator | Calculator Panel                              |
| Database                     | Find (                           |                                       |              | ^ / * -                                       |
| Tx Messages                  | Find                             | Clear                                 |              |   |
| Signal Groups                | 🖃 🛼 HS CAN                       | <u>^</u>                              | >            | / 0 9 +                                       |
| <b>₽</b> DAO                 | □ e\n EH0_Message_ (199) 3       | /                                     | <            | 4 5 6   |
| Sim Jobs                     | EngSpd (Analog)                  |                                       | <>           | 1 2 3 )                                       |
| *Ann Signals                 | Gear_status (Analog)             |                                       | <=           | 0.(   |
| Proposition States           | E BVB IC_Message (400)           |                                       | >=           |   |
| Place                        | Bighar MBEM Message (300)        |                                       | i.           | PARAmetica Canala                             |
| T-NOUES                      | Signal 0 (Analog)                |                                       | bit0(        | Name  |
|                              | PYO RPAS Message (500)           |                                       | bit1(        |   |
| EFUNCTION BLOCKS             | Signal_0 (Analog)                |                                       | bit2(        |   |
| Physical 10                  | TCU_Message (200)                |                                       | bit3(        | Add   |
| Logger                       | Signal ((Analog)                 | ×                                     | bit4(        |   |
|                              | 0                                |                                       | bit6(        | <ul> <li>Hide auto-generated items</li> </ul> |
|                              | Nelve                            |                                       | bit7(        | Test  |
|                              | Present                          | <u>^</u>                              | byte0(       |   |
|                              | Signal Valid                     |                                       | byte1        | Hesult:                                       |
|                              | Gradient<br>New Value Time (abs) |                                       | byte2(       |   |
|                              | Start Value                      |                                       | byte3(       |   |
|                              | Min Value                        | ~                                     | one( v       |   |
|                              | Indx value                       | · · · · · · · · · · · · · · · · · · · |              |   |

Figure 17: Selection of expression for start logging option

| Collection Configuration<br>What kind of collection should we use for this trigger? Select " <u>Pre/Post Collection</u> " for a collection of the <b>vehicle bus</b><br>and after the trigger event. Select " <u>One-shot Report</u> " for a <b>report of the values</b> of all the channels at the time | <b>before</b><br>of the                       |
|--|---|
| Vent.     Pre/Post Collection     Stop on expression      changing     f*  | 00 🖵 Timeout sec                              |
| Configure " <u>Pre-Trigger</u> " mode. First, select your <u>card size</u> . Then, choose the way you'd like to define your buffer. Fir your " <u>Messages Before Trigger</u> " and " <u>Messages After Trigger</u> ". All of these will <b>affect your maximum number o</b>                             | nally, select<br>f captures.                  |
| Card Size       32 GB       # of Trigger Events (Maximum): 1728         By time       Messages Before Trigger       200         By number of messages       Messages After Trigger       200   |   |
| Restart Options<br>Once we collect 1728 trigger events, what should we do? Select " <u>Stop</u> " to end<br>the capture. Select " <u>Restart</u> " to start capturing again (will erase oldest<br>capture). Wireless neoVI? Select both for  | y)<br>bload the data to<br>r fastest network. |
| C Stop 8 € Restart   | /iFi  |

Figure 16.b: Configuration for Collection of messages

- Collection configuration
  - Select 'Pre/Post collection' option (Figure 16.b (5))
  - Select card size 32GB for plasma (Figure 16.b (6))
  - Select 'By number of messages' (Figure 16.b (7)) and Set Messages before Trigger to 200 and Messages after Trigger to 200.So it will log 200 messages before trigger and 200 messages after trigger condition.
- Restart option (Figure 16.b (8)): Select 'Restart' option. So it will restart logging after completion of collection. It will again check for start collection expression for restart logging.
- Collection upload (Figure 16.b (9)): Check box of 3G and WiFi option as per availability or select both.

| 🚸 VehicleScape DAQ 🖾 吶 Messages Ed   | itor 🔀 🟮 Network Databases 💈  | 🕄 🕒 Messages 🔀   |   |
|--|---|--|---|
| Database/Hardware Setup   Channels   PC Loggi  | ng Standalone Logging Gateway   | Online DAQ 1   | •   |
| Reporting Options  |   |  |   |
| How would you like feedback about the status<br>Select " <u>neoVI MOTE</u> " to <b>use the neoVI MOTE</b><br>will display on the Wireless NeoVI website.                               | of the logger? Select " <u>LED</u> " to <b>flash t</b><br>E to display status and manually trigge                                 | he LEDs based on the status of the<br>er. Select " <u>Wireless NeoVI</u> " to choose               | capture.<br>signals that                    |
| LEDs neoVI MOTE  | 🦳 Wireless NeoVI (PLASMA Only   | ) 📃 Beep on Wakeup (PLASN  | 1A only)                                    |
| When the logger is running, the <b>red</b> LED will slowly. When capturing post-trigger data, the  | olink at a fast rate. When data is capt<br>e <b>green</b> LED will flash quickly.   | ured to the SD card, the green LED   | will flash                                  |
| Power Management   | ression   |  |   |
|  |   |  |   |
| When should the neoVI go into low-current draw<br>Present <sup>*</sup> to go to <b>sleep once a message is r</b>   | v (sleep) mode? Select " <u>Never</u> " to <b>alv</b><br>n <b>ot on the bus</b> after a certain period                            | <b>vays stay awake</b> . Select " <u>When M</u><br>d of time. Likewise, " <u>On Expression</u> " ( | <u>essage Not</u><br>will sleep <b>once</b> |
| an expression is true for a period of time an<br>cause the neoVI to stay awake or wake   | d " <u>On No Bus Activity</u> " will sleep <b>once</b>  | the bus is quiet. Note: Any bus  | activity will                               |
| 3 • Never • When Messag  | e Not Present O On E  | xpression C On No Bu   | s Activity                                  |
| Generate   |   |  |   |
| Vehicle Spy can generate a database that will <b>d</b><br>currently have loaded. <b>If you want this</b> , chec<br><u>to SD card</u> " or just " <u>Save to local file system</u> ". T | ecode all the data logged by this<br>k " <u>Generate decoding database</u> " and<br>hen, click " <u>Generate</u> " and <b>go!</b> | setup. It does this by combining the<br>then choose if you'd like the databas                      | databases you<br>se to " <u>Transfer</u>    |
| 4 Generate decoding database   | <ul> <li>Transfer to SD card</li> <li>Save to local file system</li> </ul>  | 5 Generate for CoreMini  | ]   |
| Save setup as VS3ZIP   | Save in existing setup  | Generate for wireless Devi   | Ces   |
| (Needed for WirelessNeoVI.com)   | C Save in data directory  | Advanced Options   | Clean                                       |
|  | 6 🔠 Generate  |  |   |

Figure 16.c: Configuration for Reporting and Generation

- Reporting option (Figure 16.c (1)): Check box of 'LED' option. It will give process status on LEDs of device. Also check box of 'Beep on wakeup (PLASMA only)' option. So it will give beep when device will wake up.
- Power management (Figure 16.c (2)): Select 'Never' option (Figure 16.c (3)). So it will never go in sleep mode.
- Generate (Figure 16.c (4)): Select 'Generate decoding database',
   'Transfer to SD card' and check box of 'generate for wireless device'
   (Figure 16.c (5))

**2.4.2.4** Create wivi file:

- To create .wivi file of all configuration click on Higgs Generate tab (Figure 16.c (6)) on VehicleScape DAQ.
- It will open the 'Coremini Executable Generator' window.
- Make sure that coremini compiled without error (Figure 18 (1)).
- Click on 'Export wivi file' option (Figure 18 (2)). It will export wivi file then Save file in destination folder
- Upload the file to device from wivi server and check output.

| CoreMini Executable Generator   | ×                 |
|---|-------------------|
| Build SD Card Partition Advanced Settings   | Help              |
| Capturing EMS_Message_(copied from DB)     Capturing TCU_Message(copied from DB)     Capturing MBFM_Message (copied from DB)     Capturing MBFM_Message (copied from DB)     Capturing MBFM_Message (copied from DB)     Using standard filtered circular capture block for Collection 1.     Generating message database file: C:\IntrepidCS\Vehicle Spy 3(60045)\Data Directory\Loggere Scipts\DAQ 1_MessageDataba     File added to SD Card image: C:\IntrepidCS\Vehicle Spy 3(60045)\Data Directory\Loggere Scipts\DAQ 1_MessageDataba     File added to SD Card image: C:\IntrepidCS\Vehicle Spy 3(60045)\Data Directory\Loggere Scipts\DAQ 1_MessageDataba     File added to SD Card image: C:\IntrepidCS\Vehicle Spy 3(60045)\Data Directory\Loggere Scipts\DAQ 1_MessageDataba     File added to SD Card image: C:\IntrepidCS\Vehicle Spy 3(60045)\Data Directory\Loggere Scipts\DAQ 1_MessageDataba     File added to SD Card image: C:\IntrepidCS\Vehicle Spy 3(60045)\Data Directory\Loggere Scipts\DAQ 1_MessageD     Exe Compressed to 5120 bytes (31.4 % of original size)     CoreMini Header (270 bytes)     Process Lists: 28 (28 bytes)     Process Lists: 28 (28 bytes)     Function Block Script Compiled: 5 (168 bytes)     Function Block Script Steps Compiled: 10 (180 bytes)     Function Block Script Steps Compiled: 10 (180 bytes)     Function Block Captures Compiled: 1 (58 bytes)     Function Block Capture Static Filters Compiled: 3 (24 bytes)     Argument Stack Size: 63 (504 bytes)     Argument Stack Size: 63 (88 (1504 bytes))     Expression Stack Size: 63 (80 tytes)     Expression Stack Size: 63 (bytes)     Expression Stack Size: 63 (bytes)     Captures Directory Stack Size: 63 (byte | se.vsdl<br>atabas |
| Expression Stack Size: 188 (1504 bytes)   | >                 |
| Compile CoreMini<br>CoreMini Compiled with Warnings at UTC 2014/04/30 05:50:28. Compile Copy to C<br>Wireless neoVI<br>Export a .wivi package for use with Wireless neoVI Export Wiveless neoVI   | Clipboar<br>2     |
| Device Configuration and Download   |                   |
| neoVI (USB) neoECU (CAN) RS232/UART/Bluetooth   |                   |
| Configure Device (Bit Rates, Initial Pin States, Resistor Enables, etc): Here Hardware Setup Download CoreMini : Device neoVI FIRE 52567 V Storage SD Card V  |                   |
| Send Extract Clear Run CoreMini After Download Advanced Settings  |                   |
| Connected to device. SD Card: 1.83 GB   |                   |

Figure 18: Settings for complied Wivi file

**Note:** Refer Script4.vs3zip file for details. You can Create .wivi file and upload in to device. Check output with simulator (<u>Simulator.vs3zip</u> file similar to simulator file for Script 1).

#### 3. Contact Us:



Intrepid Control Systems, Inc. Email: <u>icsindia@intrepidcs.com</u> Website: <u>www.intrepidcs.com</u>