Communication Control (0x28)

Purpose: This Communication control is a process of controlling the communication between one or more ECUs and one other network

Introduction

- How can we control communication?

  Simple way of controlling communication is to control either **transmitter or receiver**

- This can be achieved by this service by Controlling the communication is not only the purpose but also can **communication type.**

Sub-functions

Enum Values	Description
0x00	Enable **Rx** & **Tx**
0x01	Enable **Rx** & Disable **Tx**
0x02	Disable **Rx** & Enable **Tx**
0x03	Disable **Rx** & **Tx**
0x04	Enable **Rx** & Disable **Tx** with Enhanced Address information
0x05	Enable **Rx** & **Tx** with Enhanced Address information
0x06 to 0x3F	SAE Reserved
0x40 to 0x5F	OEM Specific
0X60 to 0x7E	Supplier specific
0x7F	SAE Reserved

Request Frame:

1. **Service Id**
2. **Sub-function**
3. **Communication Type (If necessary)**
4. **Node Identification number (If necessary)**

Positive Response Frame:

1. **Service Id**
2. **Sub-function**

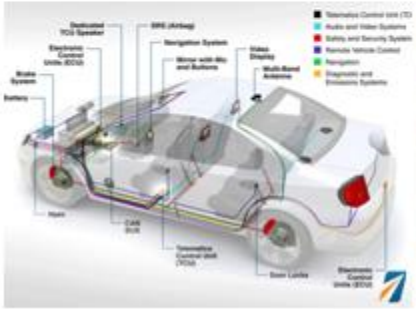
Negative Response Frame:

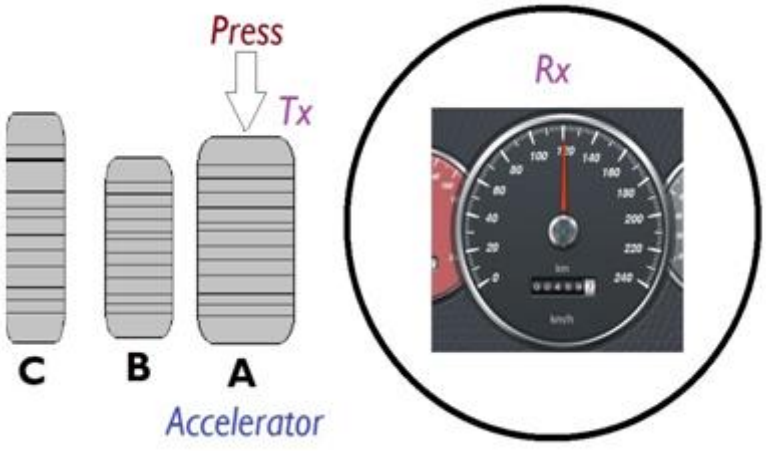
1. **Negative Response (7F)**
2. **Service Id**
3. **NRC Code**

Understanding on Communication Control !!

Communication Type

- There are two types of communication can happen in ECU
  - Normal Communication
  - Network Communication



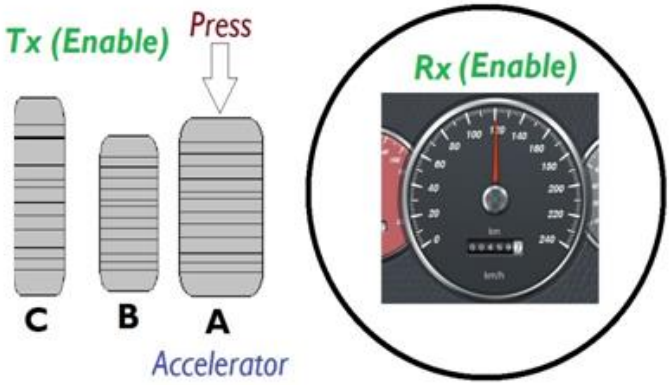


Assumption scenario:

- **Tester wants to control the communication between two nodes, he can control it by following ways**

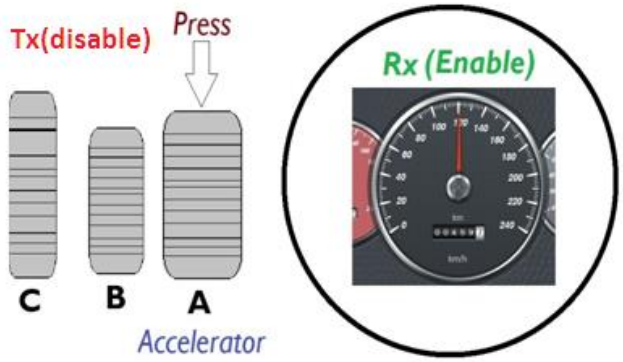
Sub-functions for Communication Control

- **Enable Rx & Tx - 0x00:** This sub-function is used to enable the **reception and transmission** of messages for the **specified communication Type** (Note)



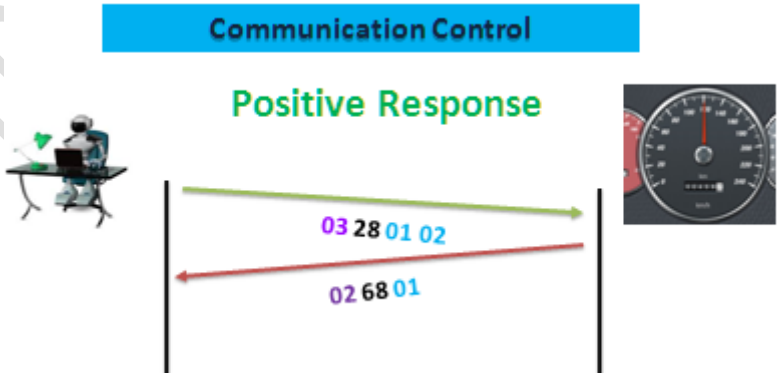
Request Response frame

- **Enable Rx & Disable Tx**: This sub-function is used to enable the **reception and disable the transmission** messages for the specified

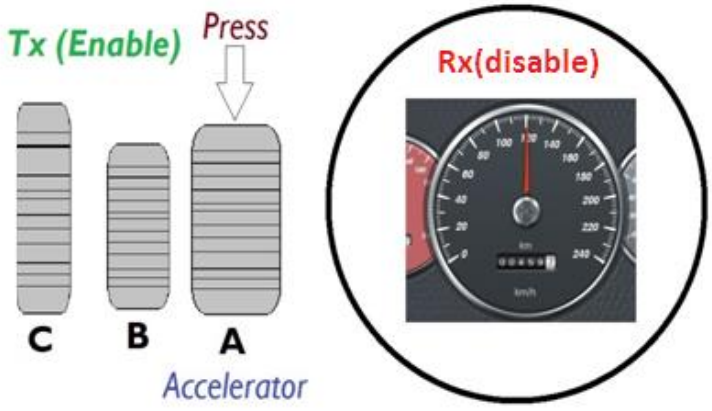


communication Ty pe

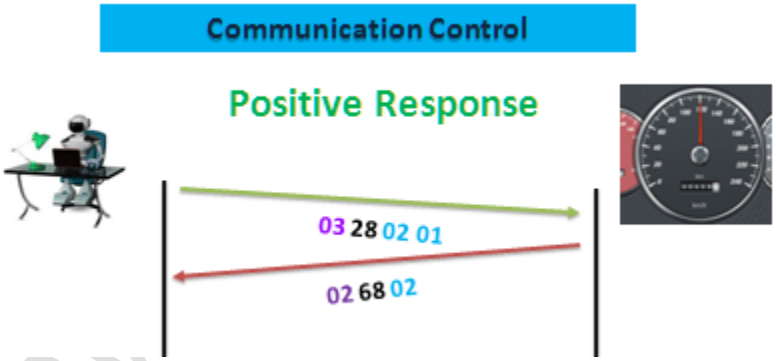
Request Response frame



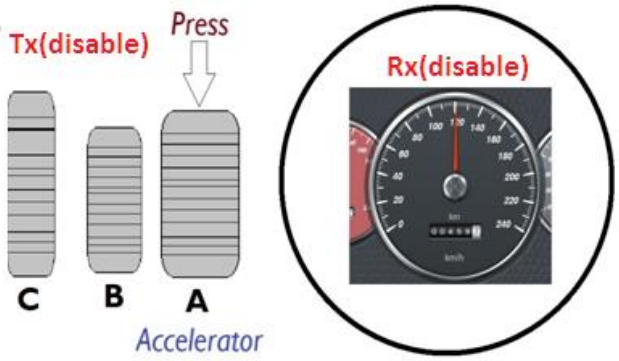
- **Disable Rx & Enable Tx:** This sub-function is used to disable the **reception and enable the transmission** messages for the specified communication Type



Request Response Frame



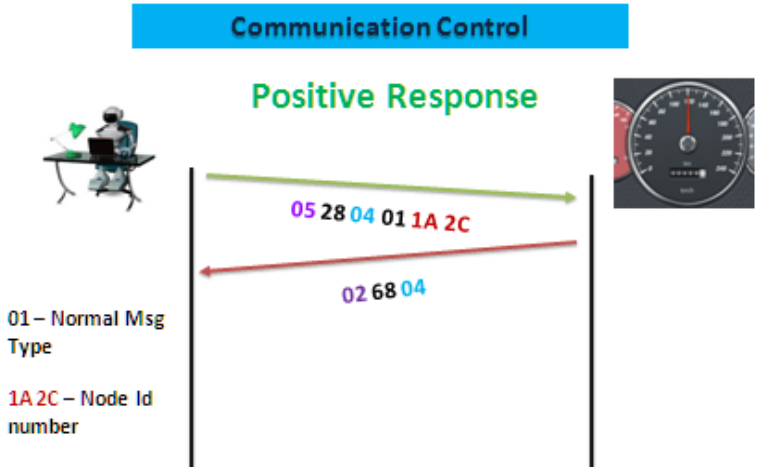
- **Disable Rx & Tx:** This sub-function is used to disable both **reception and transmission** of messages



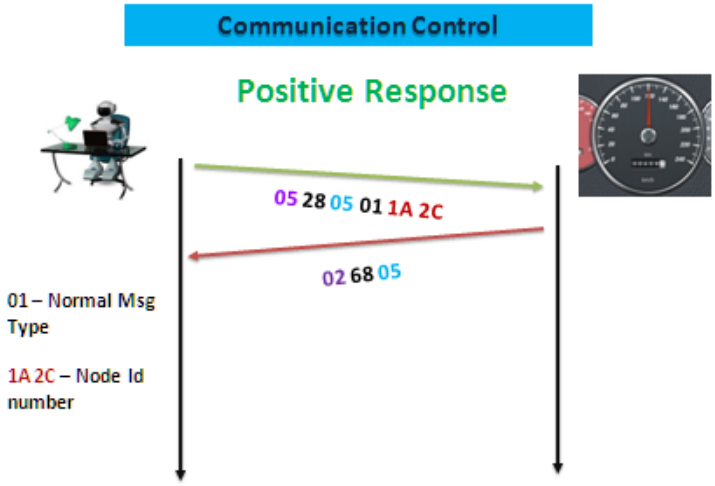
for the specified communication Type.

Request Response Frame

- **Enable Rx & Disable Tx with Enhanced Address information:** This sub-function is used to enable the **reception and disable the transmission** messages for the specified communication Type with enhanced information that is used in the request frame



- **Enable Rx & Tx with Enhanced Address information:** This sub-function is used to enable the **reception and transmission** of messages for the specified communication Type enhanced information that is used in the request frame



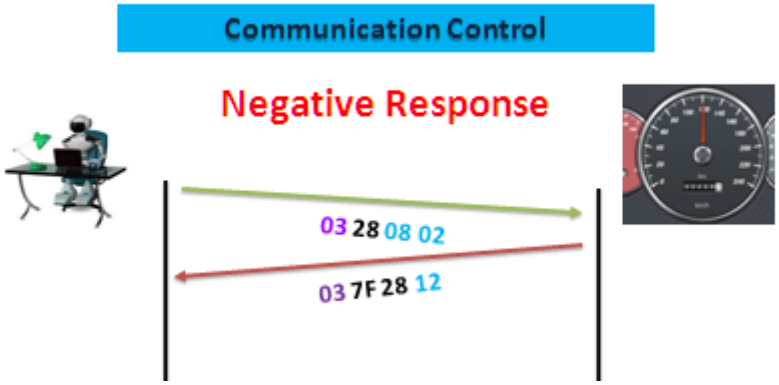
The Enabling and Disabling is done, But why its required

List of NRCs Supported

1. **0x12 Sub-function Not Supported**
2. **0x13 Incorrect Message Length**
3. **0x22 Conditions Not Correct**
4. **0x31 Request Out of Range**

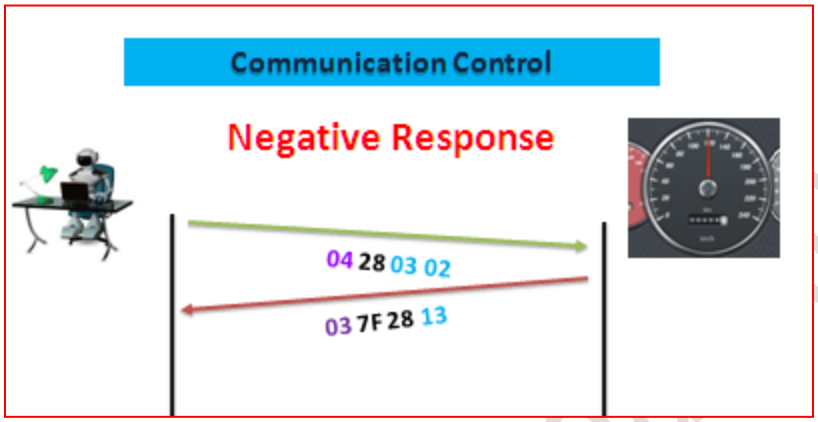
Sub-function Not Supported (0x12)

ECU responds with **NRC 12** if tester tries to request with unsupported sub-function and the **sub function is not supported** as per requirement



Incorrect Message Length (0x13)

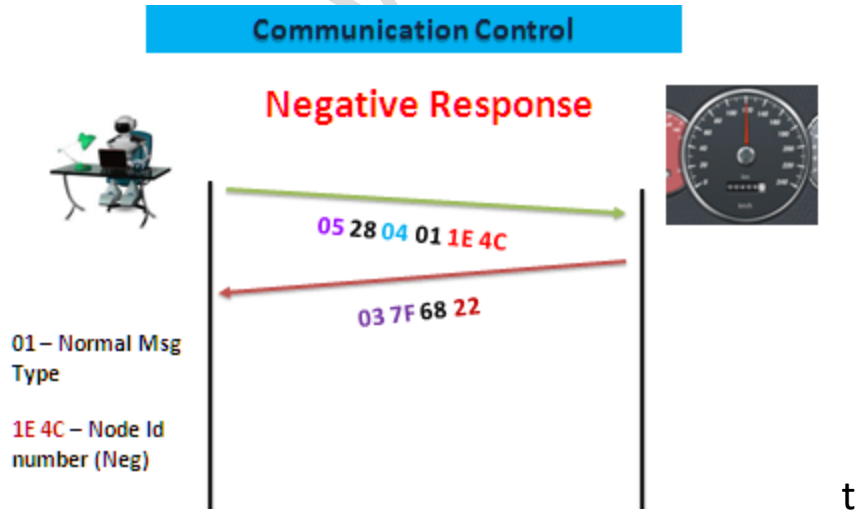
ECU responds with **NRC 13** if tester tries to request with **incorrect message length**



Sub-function Not Supported (0x22)

ECU responds with **NRC 22** if tester tries to request this service when the **conditions are not met**.

Assumption Requirement says, **Node Id 1E 4C** is been communicating but by the same time tester requests to enable/disable, let's see the response for the reques

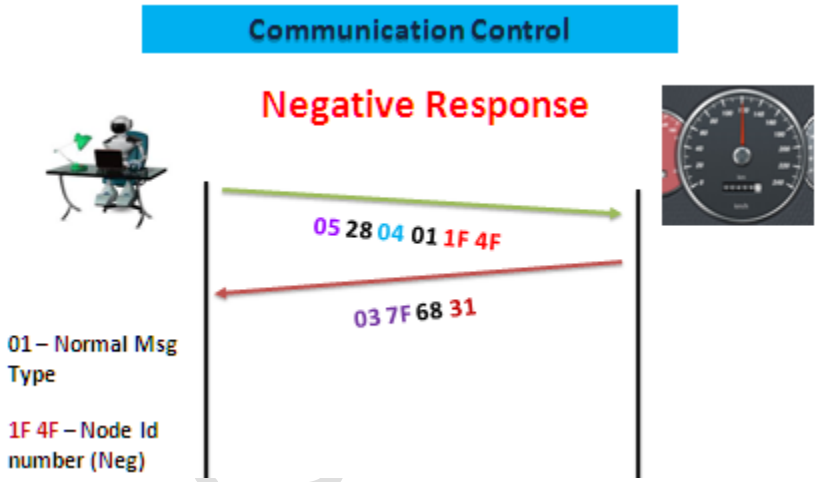


t

Sub-function Not Supported (0x31)

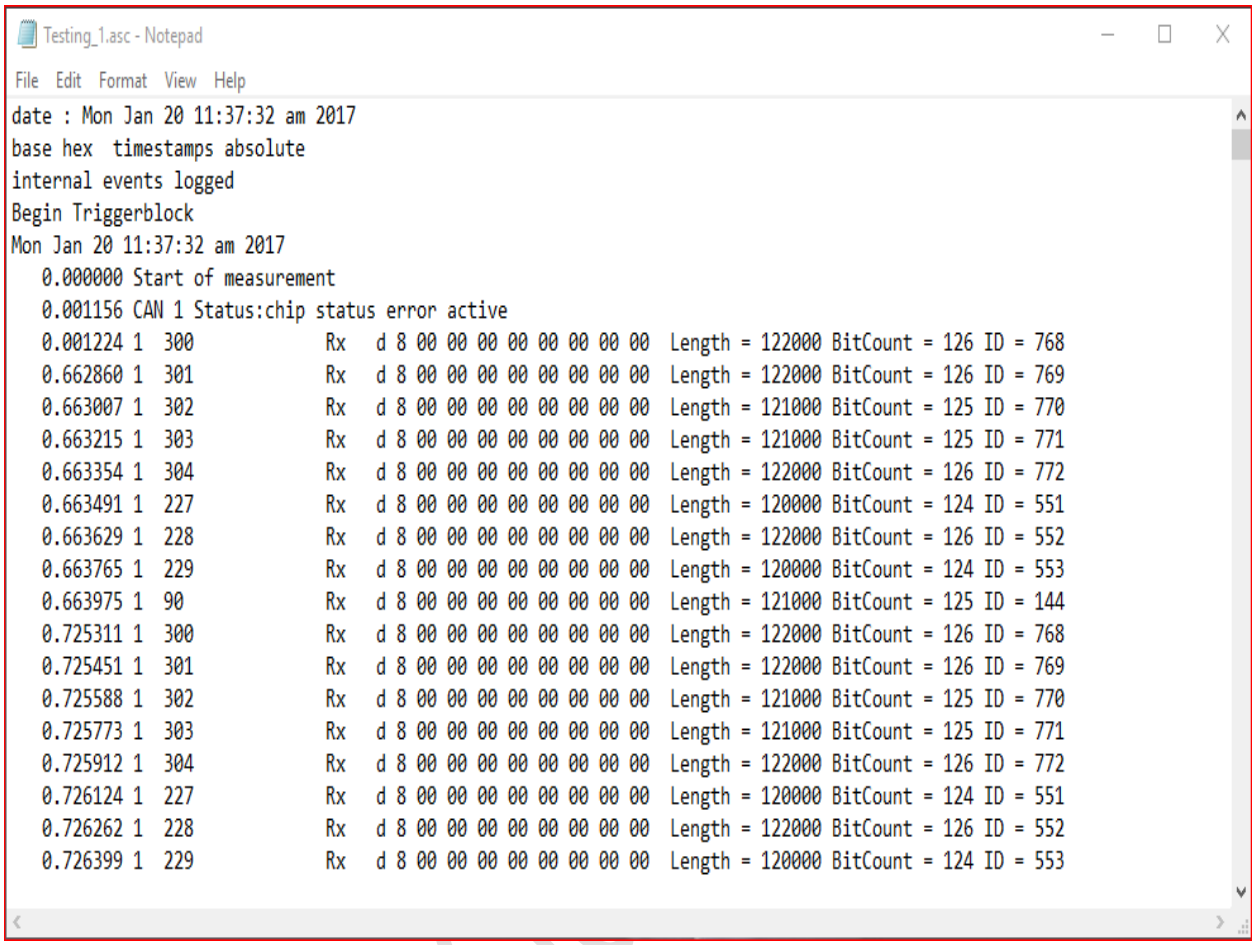
ECU responds with **NRC 31** if tester tries to request this service with DTC that is **out of range**.

Assumption Requirement says, **Node Id 1F 4F** is not in the range and not valid IDs but tester requests to enable/disable the invalid node, let's see the response for the request



More to Know!!

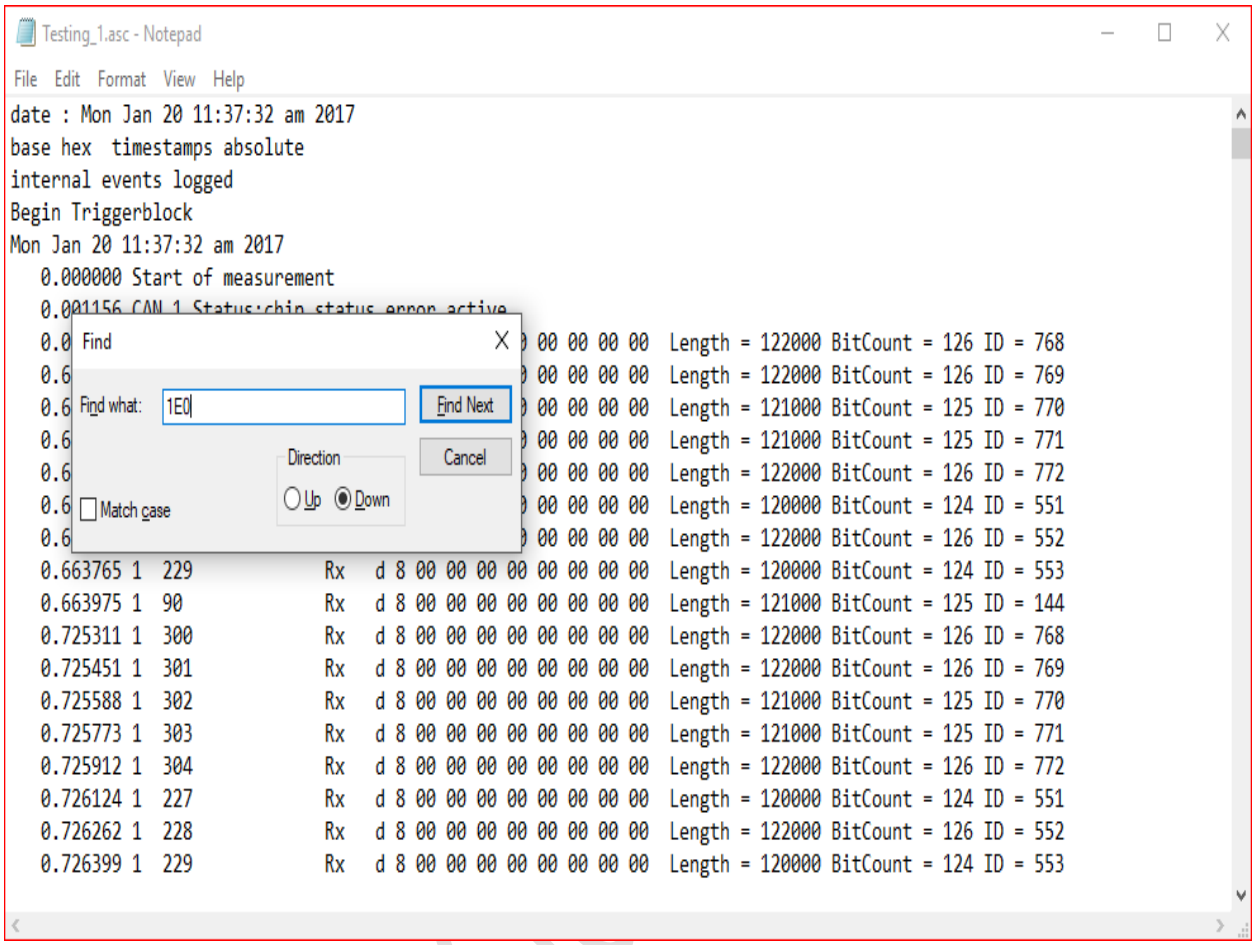
Logs

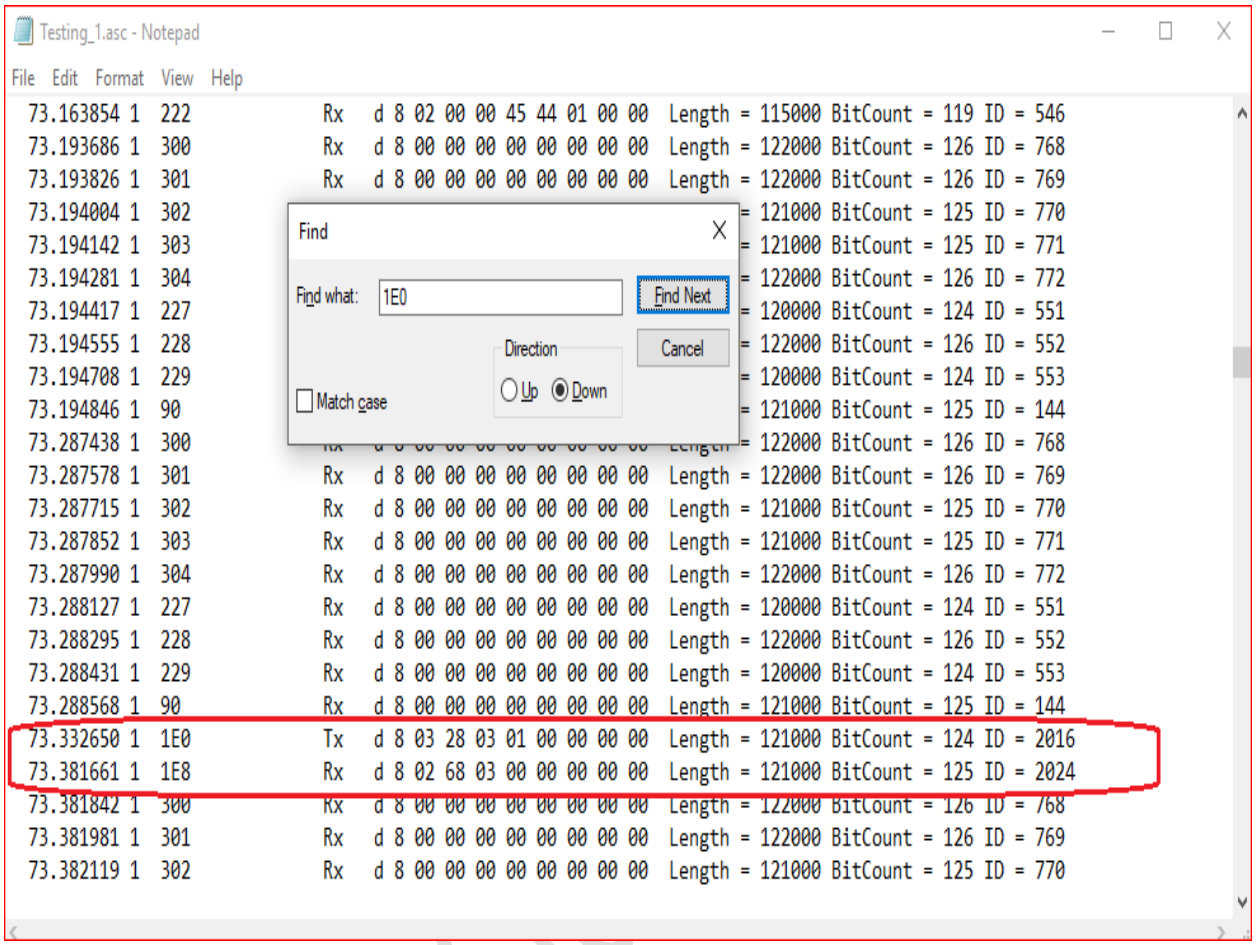


Testing_1.asc - Notepad

File Edit Format View Help

```
date : Mon Jan 20 11:37:32 am 2017
base hex  timestamps absolute
internal events logged
Begin Triggerblock
Mon Jan 20 11:37:32 am 2017
   0.000000 Start of measurement
   0.001156 CAN 1 Status:chip status error active
   0.001224 1  300             Rx   d 8 00 00 00 00 00 00 00 00  Length = 122000 BitCount = 126 ID = 768
   0.662860 1  301             Rx   d 8 00 00 00 00 00 00 00 00  Length = 122000 BitCount = 126 ID = 769
   0.663007 1  302             Rx   d 8 00 00 00 00 00 00 00 00  Length = 121000 BitCount = 125 ID = 770
   0.663215 1  303             Rx   d 8 00 00 00 00 00 00 00 00  Length = 121000 BitCount = 125 ID = 771
   0.663354 1  304             Rx   d 8 00 00 00 00 00 00 00 00  Length = 122000 BitCount = 126 ID = 772
   0.663491 1  227             Rx   d 8 00 00 00 00 00 00 00 00  Length = 120000 BitCount = 124 ID = 551
   0.663629 1  228             Rx   d 8 00 00 00 00 00 00 00 00  Length = 122000 BitCount = 126 ID = 552
   0.663765 1  229             Rx   d 8 00 00 00 00 00 00 00 00  Length = 120000 BitCount = 124 ID = 553
   0.663975 1  90              Rx   d 8 00 00 00 00 00 00 00 00  Length = 121000 BitCount = 125 ID = 144
   0.725311 1  300             Rx   d 8 00 00 00 00 00 00 00 00  Length = 122000 BitCount = 126 ID = 768
   0.725451 1  301             Rx   d 8 00 00 00 00 00 00 00 00  Length = 122000 BitCount = 126 ID = 769
   0.725588 1  302             Rx   d 8 00 00 00 00 00 00 00 00  Length = 121000 BitCount = 125 ID = 770
   0.725773 1  303             Rx   d 8 00 00 00 00 00 00 00 00  Length = 121000 BitCount = 125 ID = 771
   0.725912 1  304             Rx   d 8 00 00 00 00 00 00 00 00  Length = 122000 BitCount = 126 ID = 772
   0.726124 1  227             Rx   d 8 00 00 00 00 00 00 00 00  Length = 120000 BitCount = 124 ID = 551
   0.726262 1  228             Rx   d 8 00 00 00 00 00 00 00 00  Length = 122000 BitCount = 126 ID = 552
   0.726399 1  229             Rx   d 8 00 00 00 00 00 00 00 00  Length = 120000 BitCount = 124 ID = 553
```


Testing_1.asc - Notepad
File Edit Format View Help
date : Mon Jan 20 11:37:32 am 2017
base hex timestamps absolute
internal events logged
Begin Triggerblock
Mon Jan 20 11:37:32 am 2017
0.000000 Start of measurement
Find
Find what: 1E0
Find Next
Cancel
Direction
Up
Down
Match case
0.663765 1 229 Rx d 8 00 00 00 00 00 00 00 00 Length = 120000 BitCount = 124 ID = 553
0.663975 1 90 Rx d 8 00 00 00 00 00 00 00 00 Length = 121000 BitCount = 125 ID = 144
0.725311 1 300 Rx d 8 00 00 00 00 00 00 00 00 Length = 122000 BitCount = 126 ID = 768
0.725451 1 301 Rx d 8 00 00 00 00 00 00 00 00 Length = 122000 BitCount = 126 ID = 769
0.725588 1 302 Rx d 8 00 00 00 00 00 00 00 00 Length = 121000 BitCount = 125 ID = 770
0.725773 1 303 Rx d 8 00 00 00 00 00 00 00 00 Length = 121000 BitCount = 125 ID = 771
0.725912 1 304 Rx d 8 00 00 00 00 00 00 00 00 Length = 122000 BitCount = 126 ID = 772
0.726124 1 227 Rx d 8 00 00 00 00 00 00 00 00 Length = 120000 BitCount = 124 ID = 551
0.726262 1 228 Rx d 8 00 00 00 00 00 00 00 00 Length = 122000 BitCount = 126 ID = 552
0.726399 1 229 Rx d 8 00 00 00 00 00 00 00 00 Length = 120000 BitCount = 124 ID = 553


Testing_1.asc - Notepad
File Edit Format View Help
73.163854 1 222 Rx d 8 02 00 00 45 44 01 00 00 Length = 115000 BitCount = 119 ID = 546
73.193686 1 300 Rx d 8 00 00 00 00 00 00 00 00 Length = 122000 BitCount = 126 ID = 768
73.193826 1 301 Rx d 8 00 00 00 00 00 00 00 00 Length = 122000 BitCount = 126 ID = 769
73.194004 1 302 = 121000 BitCount = 125 ID = 770
73.194142 1 303 = 121000 BitCount = 125 ID = 771
73.194281 1 304 = 122000 BitCount = 126 ID = 772
73.194417 1 227 = 120000 BitCount = 124 ID = 551
73.194555 1 228 = 122000 BitCount = 126 ID = 552
73.194708 1 229 = 120000 BitCount = 124 ID = 553
73.194846 1 90 = 121000 BitCount = 125 ID = 144
73.287438 1 300 = 122000 BitCount = 126 ID = 768
73.287578 1 301 Rx d 8 00 00 00 00 00 00 00 00 Length = 122000 BitCount = 126 ID = 769
73.287715 1 302 Rx d 8 00 00 00 00 00 00 00 00 Length = 121000 BitCount = 125 ID = 770
73.287852 1 303 Rx d 8 00 00 00 00 00 00 00 00 Length = 121000 BitCount = 125 ID = 771
73.287990 1 304 Rx d 8 00 00 00 00 00 00 00 00 Length = 122000 BitCount = 126 ID = 772
73.288127 1 227 Rx d 8 00 00 00 00 00 00 00 00 Length = 120000 BitCount = 124 ID = 551
73.288295 1 228 Rx d 8 00 00 00 00 00 00 00 00 Length = 122000 BitCount = 126 ID = 552
73.288431 1 229 Rx d 8 00 00 00 00 00 00 00 00 Length = 120000 BitCount = 124 ID = 553
73.288568 1 90 Rx d 8 00 00 00 00 00 00 00 00 Length = 121000 BitCount = 125 ID = 144
73.332650 1 1E0 Tx d 8 03 28 03 01 00 00 00 00 Length = 121000 BitCount = 124 ID = 2016
73.381661 1 1E8 Rx d 8 02 68 03 00 00 00 00 00 Length = 121000 BitCount = 125 ID = 2024
73.381842 1 300 Rx d 8 00 00 00 00 00 00 00 00 Length = 122000 BitCount = 126 ID = 768
73.381981 1 301 Rx d 8 00 00 00 00 00 00 00 00 Length = 122000 BitCount = 126 ID = 769
73.382119 1 302 Rx d 8 00 00 00 00 00 00 00 00 Length = 121000 BitCount = 125 ID = 770
Find
Find what: 1E0
Find Next
Cancel
Direction
Up
Down
Match case