# Research & Vehicle Technology
## "Infotainment Systems Product Development"

# Feature – Track Apps

# Infotainment Subsystem Part Specific Specification (SPSS)

Version 1.1
**UNCONTROLLED COPY IF PRINTED**

**Version Date: September 23, 2022**

**FORD CONFIDENTIAL**

# <u>Revision History</u>

| Date | Version | Notes | |
|---|---|---|---|
| **August 11, 2021** | **1.0** | **Initial Release** | |
| | | | |
| **September 23, 2022** | **1.1** | | |
| | | 875796/B-General Requirements | tmertiri:add new req |
| | | TRAP-REQ-524797/A-Missing Signals | tmertiri: new requirement |
| | | 889433/B-Requirements | tmertiri: update structure req |
| | | TRAP-REQ-416339/B-Activate / Deactivate Line Lock | tmertiri: update requirement |
| | | TRAP-REQ-416341/B-Line Lock Status | tmertiri: update parameters |

# Table of Contents

# 1 Architectural Design

## 1.1 Overview

Track Apps is a set of mini features or functions that are first made available in a performance vehicle. Main components, which this SPSS has further details below are Acceleration Timer, Brake Performance, Lap Timer, Launch Control, Line Lock. Due to the nature of operation and driver safety, the features are typically initialized in Client then main feature operation with the user is moved on to Server 1. Server 2 represents the various electrical modules that control the features.

**In some of these functions, server 1 will not notify the client of feature status. Feature status could be available to the user through other vehicle modules. A good candidate could be Server 1 for feature status display.**

Any function/feature operation is considered a test, since the user is "testing" the vehicle performance depending on the particular operation being executed.

## 1.2 Class Description Mapping

This is a mapping of hardware modules involved in this feature. We use logical names to protect the spec from any potential changes in case a module gets updated or renamed.

| Logical Class | Physical Module (ECU) |
|---|---|
| Client | APIM |
| Server 1 | IPC |
| Server 2 | Various other modules, such as BCM, PCM, ABS |

## 1.3 TRAP-CLD-REQ-411000/A-Track Apps Client

Client provides information to the user through a graphical interface. It also may provide a way to interface back with the feature, such as to enable/disable etc.

## 1.4 TRAP-CLD-REQ-411001/A-Track Apps Server 1

Server 1 interprets the user input and in many cases, it displays test results. Server 1 also generates the test data that is sent to Client through TP for storage and long term display.

## 1.5 TRAP-CLD-REQ-414061/A-Track Apps Server 2

Server 2 are the various modules that provide feature control such as brake status or supporting information such as ignition status.

## 1.6 Logical Signal Mapping

The CAN signals mentioned throughout this document shall refer to the CAN signal's logical name. The logical names shall be mapped to their actual CAN signal names. Please use the table below to perform the mapping. The InfoCAN database file is the master file for the actual CAN signal names. Note: some CAN signals referenced throughout this document may use the logical name while some may use the actual CAN signal name.

| Logical Name | CAN Signal Name |
|---|---|
| IgnSt | Ignition_Status |
| LnchRq | LnchCtl_D_Rq2 |
| LRpmRq | LnchCtl_N_Rq2 |
| LTaMnuRq | TrakAppl_D_RqMnu |
| LTaSetRq | TrakApplActvSet_D_Rq |
| LTaSt | TrakAppl_D_Stat |

| LUnits | Mc_VehUntTrpCoUsrSel_St |
|---|---|
| LLineLckSt | BrkLineLck_D_Stat |

## 1.7 TRAP-IIR-REQ-410998/A-Track Apps Client Tx

### 1.7.1 MD-REQ-410995/A-LnchRq

LnchRq: This signal is sent from the client to the Server 2 to indicate feature change request.

| Value | Value Description |
|---|---|
| 0x0 | Off. Request Launch Control Off. |
| 0x1 | On. Request Launch Control On. |
| 0x2 | Not Used |
| 0x3 | Not Used |

### 1.7.2 MD-REQ-410996/A-LRpmRq

LRpmRq: This signal is sent from client to the Server 2. It is used to indicate user request for a specific RPM. Unit is rpm.

| Value | Value Description |
|---|---|
| 0x0 | 0 |
| 0x1 | 50 |
| .. | |
| … | |
| … | |
| 0xFF | 12750 |

### 1.7.3 MD-REQ-413529/A-LTaMnuRq

LTaMnuRq: This signal is used by many of the track app functions to allow the user to request one of the available options of track test drive. The applicable option is determined based on the available track app. Further details will be added to the functions. Signal is transmitted from client.

| Detail | State Encoded |
|---|---|
| | |
| Deactivate_TrackApp | 0x0 |
| Activate_60-0mph | 0x1 |
| Activate_100-0mph | 0x2 |
| Activate_100-0kph | 0x3 |
| Activate_200-0kph | 0x4 |
| Activate_0-30mph | 0x5 |
| Activate_0-60mph | 0x6 |
| Activate_0-100mph | 0x7 |
| Activate_0-50kph | 0x8 |
| Activate_0-100kph | 0x9 |
| Activate_0-200kph | 0xA |
| Activate_1/8mile | 0xB |
| Activate_1/4mile | 0xC |
| Activate_200m | 0xD |

| Activate_400m | 0xE |
|---------------|-----|
| Activate_LapTimer | 0xF |
| Activate_LineLock | 0x10 |
| *NotUsed1* | 0x11 |
| … | … |
| *NotUsed14* | 0x1E |
| *NotUsed15* | 0x1F |

### 1.7.4   MD-REQ-413530/A-LTaSetRq

LTaSetRq: This signal is used to send particular configuration values to Acceleration timer per user request. Signal is transmitted from client.

| Detail | State Encoded |
|--------|---------------|
|  |  |
| Null | 0x0 |
| AccTimer_Start_Automatic | 0x1 |
| AccTimer_Start_Drag | 0x2 |
| AccTimer_Start_Track | 0x3 |
| *NotUsed1* | 0x4 |
| … | … |
| *NotUsed12* | 0xF |

**NULL**: this parameter is sent when there is no user request to execute the test.
**Automatic, Drag** and **Track** are test options that the user can choose. One of this must be choose for the feature to work in addition to other signals.
**Not Used**: These are values that should not be seen in the bus. If client receives any of them, the data should be ignored.

**Automatic**       Acceleration Timer begins automatically when the vehicle starts to move. Reaction Time is not available in this setting.
**Drag**     Acceleration Timer uses a Drag Race style of lamp sequences and begins when lamp sequence finishes. Reaction Time is available in this setting.
**Track**     Acceleration Timer uses a Track Race style of lamp sequences and begins when lamp sequence finishes. Reaction Time is available in this setting.

## 1.8   TRAP-IIR-REQ-410999/A-Track Apps Client Rx

### 1.8.1   MD-REQ-410994/A-IgnSt

IgnSt: This signal is sent from server 2 to client to indicate ignition state.

| Value | Interpretation |
|-------|----------------|
| 0x0 | Unknown |
| 0x1 | Off |
| 0x2 | Accessory |
| 0x4 | Run |
| 0x8 | Start |
| 0xF | Invalid |

| **FILE: TRACK APPS SPSS V1.1 SEP 23, 2022** | **FORD MOTOR COMPANY CONFIDENTIAL**<br>*The information contained in this document is Proprietary to Ford Motor Company.* | *Page* 7 *of* 37 |
|-----------------------------------------------|---------------------------------------------------------------------------------------------------------------|------------------|

### 1.8.2    MD-REQ-413531/A-LTaSt

LTaSt: This signal is sent from the server 1 to the client to indicate applicable track app status.

| Detail | State Encoded |
|---|---|
| | |
| Null | 0x0 |
| Cancelled | 0x1 |
| Deactivated | 0x2 |
| NotUsed1 | 0x3 |
| NotUsed2 | 0x4 |
| NotUsed3 | 0x5 |
| NotUsed4 | 0x6 |
| NotUsed5 | 0x7 |

### 1.8.3    MD-REQ-416164/A-LUnits

LUnits: This signal is sent from Server 1 to client. It is used to indicate what units should the client use, Metric or English.

| Value | Value Description |
|---|---|
| 0x0 | Metric. Client will display Metric system units |
| 0x1 | English. Client will display English units. |

### 1.8.4    MD-REQ-416337/A-LLineLckSt

This signal is sent from Server 2 to the client to indicate the state of Line Lock feature.

| Value | Value Description |
|---|---|
| 0x0 | Available: Line lock function is available and can be initiated |
| 0x1 | Initiated: Line lock is initiated |
| 0x2 | Engaged: Line lock is engaged |
| 0x3 | Off: Line lock is off (was previously engaged) |
| 0x4 | Not Available: Line lock is not available to be selected |
| 0x5 | Not Used: Parameter is not to be sent in the bus |
| 0x6 | Not Used: Parameter is not to be sent in the bus |
| 0x7 | Not Used: Parameter is not to be sent in the bus |

# 2   General Requirements

## 2.1   TRAP-REQ-411002/A-Power Mode Operation

The feature should only be accessible for interaction to the user while signal IgnSt is 0x4 (Run) or 0x8 (Start).

## 2.2   TRAP-REQ-416165/A-Units Being Displayed

Client shall display the units and data acquired in the units provided by the signal LUnits. There is no need to process any data by converting it between units.

## 2.3   TRAP-REQ-524797/A-Missing Signals

If the CAN message of signal LTaSt is missing, than the application will show the same feature that exist before missing the CAN message

if the CAN message of signal LUnits is missed, than the application will show the same units that exist before missing the CAN message

if the CAN message of signal LLineLckSt is missed, than the application will show OFF(0x3) for the Line Lock feature.

# 3 Functional Definition

## 3.1 TRAP-FUN-REQ-410993/A-Launch Control

### 3.1.1 Overview

Launch Control is a feature that controls the vehicle launch. It also sets the engine revolution per minute (RPM) which must be reached before vehicle launches.
**Client will not be aware of feature status. Feature status could be available to the user through other vehicle modules.**

### 3.1.2 Requirements

#### 3.1.2.1 _TRAP-REQ-411070/A-RPM Value_

Client shall contain the RPM value within boundaries provided from configuration values.

#### 3.1.2.2 _TRAP-REQ-411004/A-RPM User Control_

The client shall transmit the RPM signal (LRpmRq) with the requested RPM value selected from the HMI. The value will continuously be transmitted until the user changes the value.

#### 3.1.2.3 _TRAP-REQ-411005/A-Launch Control User Request_

Client shall send LnchRq to the server with the request ON or OFF values. The value will continuously be transmitted until the user changes the value.

#### 3.1.2.4 _TRAP-REQ-411006/A-State Recall_

Client will retransmit the last ignition cycle RPM until the user changes the value.

### 3.1.3 Use Cases

#### 3.1.3.1 _TRAP-UC-REQ-410983/A-Enable Launch Control_

| Actors | User |
|---|---|
| Pre-conditions | Vehicle is in Ignition State of Run Vehicle at a standstill (0 mphs) Launch Control is Disabled Traction Control is On Drive Mode is Normal |
| Scenario Description | Navigating to the available HMI Menu user selects Launch Control. User changes RPM value. User toggles Launch Control state to Enable. |
| Post-conditions | Launch Control Enabled Launch Control settings saved to memory |
| List of Exception Use Cases | |
| Interfaces | Client HMI |

#### 3.1.3.2 _TRAP-UC-REQ-410984/A-Disable Launch Control_

| Actors | User |
|---|---|
| Pre-conditions | Vehicle is in Ignition State of Run |

| | Vehicle at a 0 mph Launch Control is Enabled Launch Control is Available Drive Mode is Normal |
|---|---|
| **Scenario Description** | Driver toggles the state of Launch Control to disable. |
| **Post-conditions** | Launch Control Disabled Launch Control settings saved to memory |
| **List of Exception Use Cases** | |
| **Interfaces** | Client HMI |

### 3.1.4    White Box Views

#### *3.1.4.1    Sequence Diagram*

##### 3.1.4.1.1    TRAP-SD-REQ-411300/A-Launch Control Operation
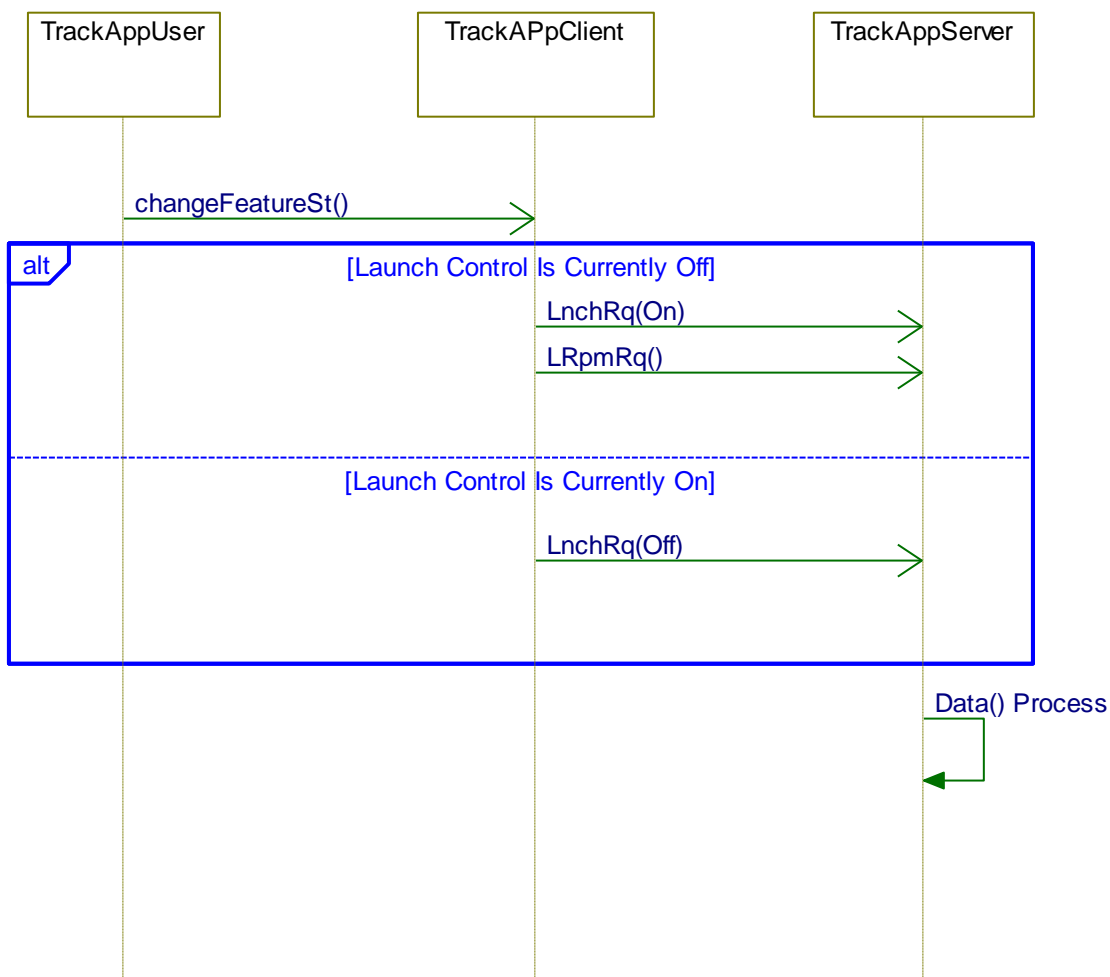
### *3.1.4.2  Activity Diagram*

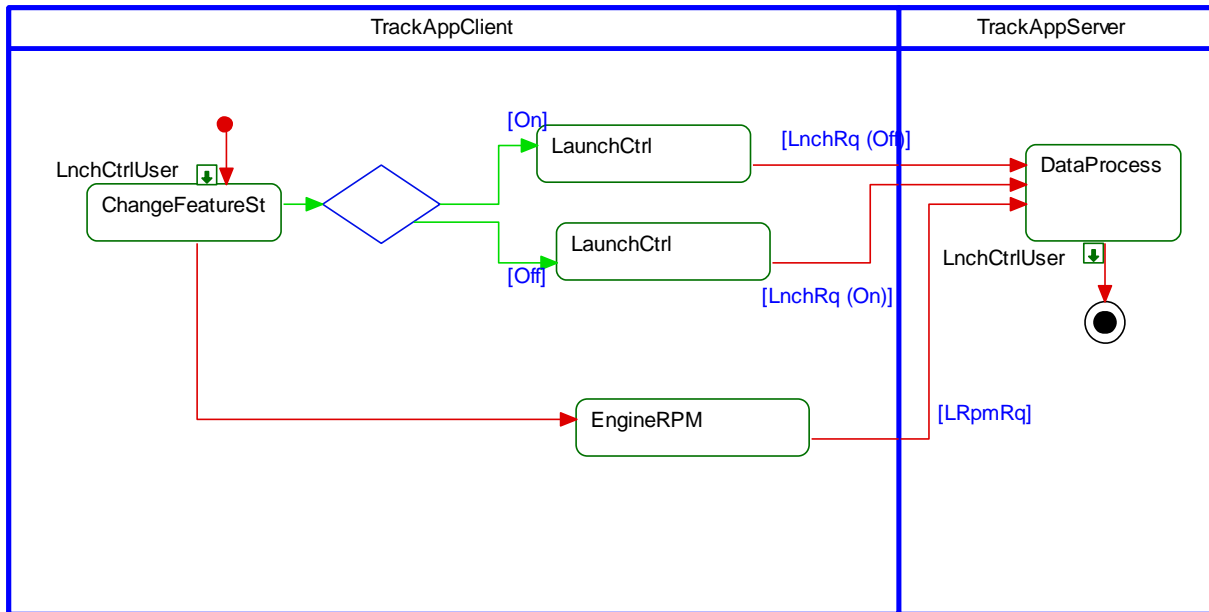#### 3.1.4.2.1  **TRAP-SD-REQ-411301/A-Launch Control Operation**



## 3.2  **TRAP-FUN-REQ-413528/A-Acceleration Timer**

### 3.2.1  **Overview**

This function provides a convenient way for the customer to test and record acceleration of the vehicle in a drag strip environment. The user request is done from the client with continued instructions provided in server 1.

The client will not display an Active or Engaged feature status, but will be updated to a canceled status if a user cancels.

### 3.2.2  **Requirements**

#### *3.2.2.1  TRAP-REQ-413544/A-Unit Convention*

The client module shall display the Acceleration Timer test event options and results in the same unit convention (English or Metric) that is specified from the Vehicle Settings menu. Distance shall be measured/displayed in units of miles or feet (English) and meters (Metric), and speed shall be in measured/displayed in mph (English) and kph (Metric).

#### *3.2.2.2  TRAP-REQ-413545/A-Tests Options Available*

The feature allows for both time based and speed based options. As mentioned earlier units are displayed in the same units that vehicle settings have been configured to be displayed. For the two units available, the below values are possible:

**English Units**
0-30 mph
0-60 mph
0-100 mph
1/8 mile
¼ mile

**Metric Units**
0-50 kph

0-100 kph
0-200 kph
200 meters
400 meters

### 3.2.2.3   TRAP-REQ-414455/A-Acceleration Timer Availability

Acceleration Timer and its functional requirements are valid when the vehicle has been configured to carry the function. If the configuration is Off or Disabled, no requirements of this functions should be executed. Client shall not send this feature's signals in the CAN bus.

### 3.2.2.4   TRAP-REQ-413673/A-Acceleration Timer Applicable Parameters

Applicable parameters for Acceleration timer function in LTaMnuRq method are:
0x0 and the values 0x5-0xE.

| Deactivate_TrackApp | 0x0 |
| --- | --- |
| | |
| Activate_0-30mph | 0x5 |
| Activate_0-60mph | 0x6 |
| Activate_0-100mph | 0x7 |
| Activate_0-50kph | 0x8 |
| Activate_0-100kph | 0x9 |
| Activate_0-200kph | 0xA |
| Activate_1/8mile | 0xB |
| Activate_1/4mile | 0xC |
| Activate_200m | 0xD |
| Activate_400m | 0xE |

### 3.2.2.5   TRAP-REQ-416162/A-Speed vs Distance Based Tests

From the available test in acceleration timer, as seen in requirement 413545, tests can be separated into two logical groups.

**Speed Based:**
0-30 mph
0-60 mph
0-100 mph
0-50 kph
0-100 kph
0-200 kph

**Distance Based:**
1/8 mile
¼ mile
200 meters
400 meters

The type of test being executed, speed or distance, controls the looks and how the data gets organized in client HMI. For examples of how that data looks like refer to HMI documentation.

### *3.2.2.6 Distance Based Requirements*

#### 3.2.2.6.1 Overview

Data being generated by the server 1 and transmitted to the client depend on the test type that the user choose. Data delivered is formatted differently for the viewer to see the data based on distance based or speed based test. The requirements in this section are applicable to distance based test choices.

#### 3.2.2.6.2 TRAP-REQ-414453/A-Distance Based Data Sessions

Upon successful feature operation completion, client will be provided with test data through TP. Refer to Transport Protocol function in this SPSS for further details.
Client shall be able to store 250 sessions. These sessions are test results that server 1 transmits to client once data is collected by server 1.
When maximum memory allocation is reached, the oldest data needs to be overwritten with newly acquired data. If user deletes data points manually, that freed internal memory needs to be utilized before data gets overwritten.

#### 3.2.2.6.3 TRAP-REQ-414489/A-Distance Based Time Data

This field displays the amount of time taken for the vehicle to travel the distance identified by each applicable end-point for a given Session. This data shall be stored in permanent memory until the Session data is cleared/deleted.

#### 3.2.2.6.4 TRAP-REQ-414490/A-Distance Based Max Speed Data

This field displays the vehicle speed achieved at the moment when the vehicle crosses each applicable end-point for a given Session. This data shall be stored in permanent memory until the Session data is cleared/deleted.

#### 3.2.2.6.5 TRAP-REQ-414495/A-Distance Based All Time Best Run Data

All time best run data is generated for each test type, ex one for 1/8 mile, 1/4 mile, 200m or 400m and it is the shortest time the test run. This data gets updated at every test if need be (new data is better than old data) and it is stored permanently by the client.

#### 3.2.2.6.6 TRAP-REQ-414486/A-Distance Based Best Data Clearing

When clearing best data, only the selected best data needs to be cleared. For example: if best 200m test data is selected, only that data is cleared, not the 400m, 1/8 mi or the ¼ mi.

#### 3.2.2.6.7 TRAP-REQ-434021/A-Distance Based All Time Best Data Clearing

When clearing All Time Best Data, only that data (distance based all time best data) will need to be cleared. The other data, Best Data content, needs to stay unchanged.

### *3.2.2.7 Speed Based Requirements*

#### 3.2.2.7.1 Overview

Data being generated by the server 1 and transmitted to the client depend on the test type that the user choose. Data delivered is formatted differently for the viewer to see the data based on distance based or speed based test. The requirements in this section are applicable to speed based test choices.

#### 3.2.2.7.2 TRAP-REQ-429819/A-Speed Based Data Sessions

Upon successful feature operation completion, client will be provided with test data through TP. Refer to Transport Protocol function in this SPSS for further details.
Client shall be able to store 250 sessions. These sessions are test results that server 1 transmits to client once data is collected by server 1.
When maximum memory allocation is reached, the oldest data needs to be overwritten with newly acquired data. If user deletes data points manually, that freed internal memory needs to be utilized before data gets overwritten.

#### 3.2.2.7.3 TRAP-REQ-416159/A-Best

Best data is the shortest time.

### 3.2.2.7.4  TRAP-REQ-416160/A-Last Data

It shows the most recent time taken for each test event since the user last reset or cleared the data.

### 3.2.2.7.5  TRAP-REQ-416161/A-Average Data

This field shows the average time taken for each test event since the user last reset or cleared the data. To calculate the average, need to add total test time accumulated and divide by the number of tests.

### 3.2.2.7.6  TRAP-REQ-434022/A-Speed Based Best Data Clearing

When clearing speed based best data, data points from all test categories from the active unit convention will be cleared/reset. For example, when clearing best data with the English unit convention active, the 0-30mph, 0-60mph, and 0-100mph data points will be cleared, but not the best data points for 0-50kph, 0-100kph, or 0-200kph.

### 3.2.2.7.7  TRAP-REQ-434023/A-Speed Based All Time Best Data Clearing

When clearing All Time Best Data, only that data (speed based all time best data) will need to be cleared. The other data, Best Data content, needs to stay unchanged.

### *3.2.2.8  TRAP-REQ-416163/A-All Time Best*

All time best data, either speed or distance based, is of the same units as Best data, however it is stored in a different memory location.

### *3.2.2.9  TRAP-REQ-415873/A-Best and All Time Best*

Some functions require the client to store Best and All Time Best Data.
What they mean is that is the data being received from Server 1 is better than what the client has, the client should store that data part in the Best or All Time Best.  The difference between the two raises because they have different reset options. The user could reset them at different time or simultaneously, so they could start their recording keeping at different timeframes, hence they could have different values stored in them. This requirements is strictly to differentiate the use or purpose of Best and All Time Best . The units of Best data are dependent on the test being executed. Refer to separate functions to know more of applicable Best data and units.

### ***3.2.2.10  Client And Server 1 Communication Situations***

### 3.2.2.10.1  TRAP-REQ-418393/A-Not Active

Not Active is the state when the feature has not been activated by the user from the client. The client should transmit the below:

> LTaMnuRq = 0x0 ('Deactivate_TrackApp')
> LTaSetRq = 0x0 ('Null')

### 3.2.2.10.2  TRAP-REQ-418394/A-Activate

Activate is the state when the user activated the feature from the client. At this time the client shall transit the below:
LTaMnuRq = applicable state of active feature. Refer to requirement 413673 for a list fo options.
LTaSetRq = applicable state of active feature. Refer to requirement 413530 for a list of options.

### 3.2.2.10.3  TRAP-REQ-418395/A-Deactivate

If the feature is deactivated via the Client, the Client should transmit the below signals:
LTaMnuRq = 0x0 ('Deactivate_TrackApp')
LTaSetRq = 0x0 ('Null')

3.2.2.10.4  TRAP-REQ-418396/A-Cancel

If the feature is cancelled, as indicated by Server 1 with the signal states below,
LTaSt ('Cancelled') or LTaSt ('Deactivated')

the client shall stranmit the signals like below:
LTaMnuRq = 0x0 ('Deactivate_TrackApp')
LTaSetRq = 0x0 ('Null')

## 3.2.3   Use Cases

### *3.2.3.1   TRAP-UC-REQ-414497/A-Start Acceleration Timer*

| **Actors** | User |
|---|---|
| **Pre-conditions** | Acceleration Timer equipped vehicle.<br>Vehicle is on |
| **Scenario Description** | User starts acceleration timer feature from client.<br>User select a test case. |
| **Post-conditions** | Acceleration timer follow up instructions show up in Server 1.<br>User proceeds to follow instructions.<br>Acceleration timer gets active. |
| **List of Exception Use Cases** | |
| **Interfaces** | Client HMI, Server1. |

### *3.2.3.2   TRAP-UC-REQ-414498/A-Stop Acceleration Timer*

| **Actors** | User |
|---|---|
| **Pre-conditions** | Acceleration timer test is in progress |
| **Scenario Description** | User selects to cancel acceleration timer in client HMI. |
| **Post-conditions** | Acceleration timer test gets interrupted and does not complete.<br>No data is sent from Server 1 to Client. |
| **List of Exception Use Cases** | |
| **Interfaces** | Client HMI |

### *3.2.3.3   TRAP-UC-REQ-414499/A-All Time Best Data Update*

| **Actors** | User |
|---|---|
| **Pre-conditions** | Acceleration timer in progress |
| **Scenario Description** | Acceleration timer test ends. A new best timer is generated.<br>Server1 sends the test data to client. |
| **Post-conditions** | Client displays the new timer as Best for the test category. |
| **List of Exception Use Cases** | |
| **Interfaces** | Client HMI |

### 3.2.4    White Box Views

#### *3.2.4.1    Activity Diagram*

#### 3.2.4.1.1    TRAP-ACT-REQ-415938/A-Activate Acceleration Timer
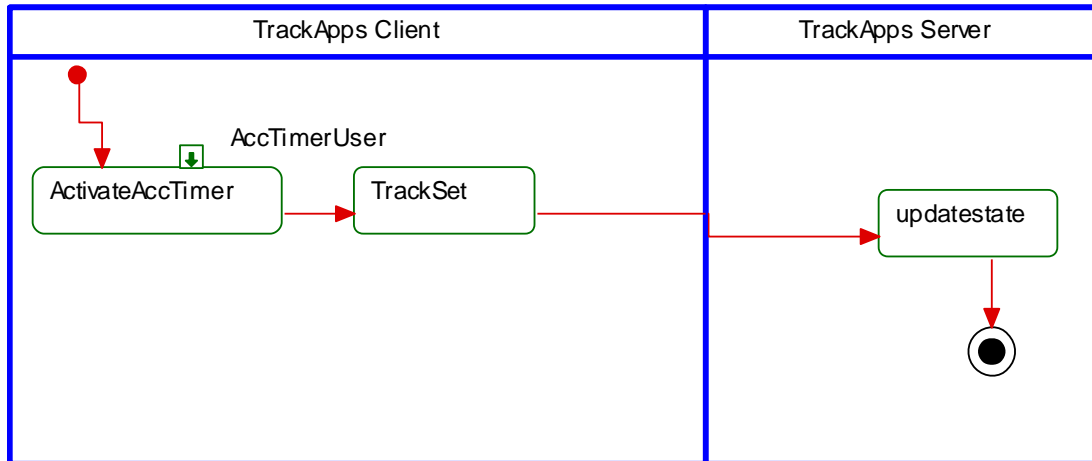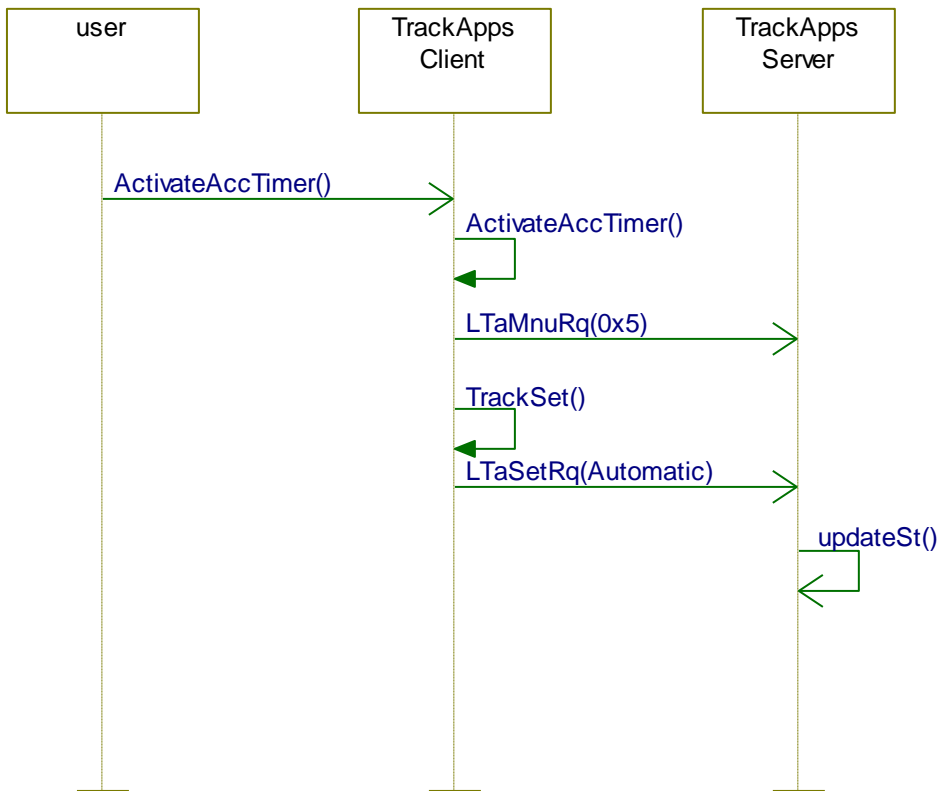


#### 3.2.4.1.2    TRAP-ACT-REQ-415986/A-Deactivate Acceleration Timer
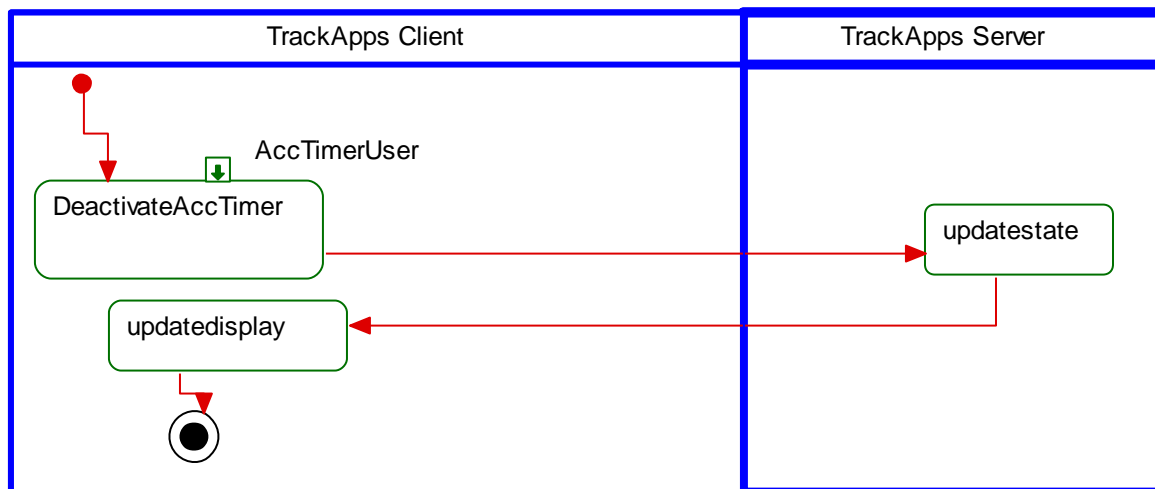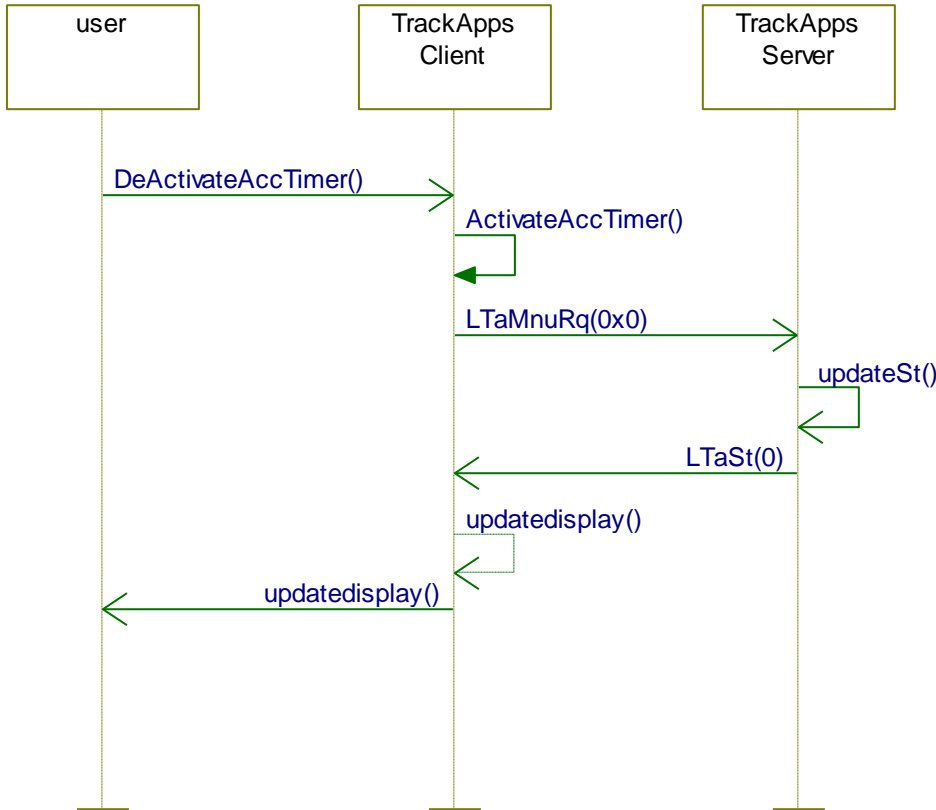
### 3.2.4.2 *Sequence Diagram*

### 3.2.4.2.1 TRAP-SD-REQ-415942/A-Activate Acceleration Timer

### 3.2.4.2.2    TRAP-SD-REQ-415982/A-Deactivate Acceleration Timer



## 3.3    TRAP-FUN-REQ-413675/A-Brake Performance

### 3.3.1    Overview

The purpose of Brake performance is to provide a convenient way for the vehicle owner to test and record their brake performance by following a few instructions. The user request is done from the client and the user proceeds to follow applicable instructions in Server 1.

The server will Not notify the client of active/engaged feature status. The client shall get update when user cancels through the server 1 and will be able to indicate canceled status.

### 3.3.2    Requirements

#### 3.3.2.1    *TRAP-REQ-413676/A-Brake Performance Applicable Parameters*

Applicable parameters for Brake Performance function in LTaMnuRq method are:
0x0 and the values 0x1-0x4.

| **Deactivate_TrackApp** | 0x0 |
|---|---|
| Activate_60-0mph | 0x1 |
| Activate_100-0mph | 0x2 |
| Activate_100-0kph | 0x3 |
| Activate_200-0kph | 0x4 |

The client shall transmit any of the parameters 0x1- 0x4 upon user requesting that particular test. The value should be transmitted for as long as the feature is active.

### 3.3.2.2   TRAP-REQ-413692/A-Brake Performance Availability

Brake performance could have a diagnostics value that indicates if the feature is available or not. If the feature/function is not available, the function and its requirements are not to be executed by the client.

### 3.3.2.3   TRAP-REQ-413693/A-Unit Convention Brake Performance

Test event and results are tied to the unit system of the client. Distance shall be measured/displayed in units of feet (English) and meters (Metric), and speed shall be in measured/displayed in mph (English) and kph (Metric).

### 3.3.2.4   TRAP-REQ-413697/A-Brake Performance Test Availability

For the two display units available, the below test values are possible:


English Units
60-0 mph
100-0 mph


Metric Units
100-0 kph
200-0 kph

### 3.3.2.5   TRAP-REQ-413698/A-Deactivate Brake Performance

User should be able to deactivate the brake performance operation at any stage of the feature. To do this, client should allow the user to send LTaMnuRq with the value of 0x0.

Client transmits the LTaMnuRq ( 0x0 ) when the user hasn't requested to run brake performance operation yet.

### 3.3.2.6   TRAP-REQ-414524/A-Brake Performance Data

Client shall store brake performance relevant data that server 1 provides, such as Last data set, Average, Best and All Time Best.

### 3.3.2.7   TRAP-REQ-416166/A-Server 2 Side Feature Termination

The signal LTaSt is used to provide the client with the feature status on Server 2 side. The values LTaSt (0x1) and LTaSt (0x2 )  (Cancelled and Deactivated) mean that the feature has been terminated by the user.

The client should set the values of LTaMnuRq to 0x0 and LTaSetRq to 0x0 .

### 3.3.2.8   TRAP-REQ-416167/A-Best Data

Best data is the shortest distance. This means the brakes can stop the vehicle at shorter distance, which means better brakes. Al time best also stores distance value.

### 3.3.2.9   TRAP-REQ-416168/A-Average

Average data is found by adding total number of sample data values and diving by the number of sample data.

### **3.3.2.10  Client And Server 1 Communication Situations**

3.3.2.10.1  TRAP-REQ-418393/A-Not Active

Not Active is the state when the feature has not been activated by the user from the client. The client should transmit the below:

       LTaMnuRq = 0x0 ('Deactivate_TrackApp')
       LTaSetRq = 0x0 ('Null')

3.3.2.10.2  <u>TRAP-REQ-418394/A-Activate</u>

Activate is the state when the user activated the feature from the client. At this time the client shall transit the below:
LTaMnuRq = applicable state of active feature. Refer to requirement 413673 for a list fo options.
LTaSetRq = applicable state of active feature. Refer to requirement 413530 for a list of options.


3.3.2.10.3  <u>TRAP-REQ-418395/A-Deactivate</u>

If the feature is deactivated via the Client, the Client should transmit the below signals:
LTaMnuRq = 0x0 ('Deactivate_TrackApp')
LTaSetRq = 0x0 ('Null')


3.3.2.10.4  <u>TRAP-REQ-418396/A-Cancel</u>

If the feature is cancelled, as indicated by Server 1 with the signal states below,
LTaSt ('Cancelled') or LTaSt ('Deactivated')

the client shall stranmit the signals like below:
LTaMnuRq = 0x0 ('Deactivate_TrackApp')
LTaSetRq = 0x0 ('Null')


## 3.3.3  Use Cases

### *3.3.3.1  TRAP-UC-REQ-414526/A-Start Brake Performance*

| Actors | User |
|---|---|
| Pre-conditions | Brake Performance feature equipped vehicle.<br>Vehicle is on |
| Scenario Description | User starts brake performance feature from client.<br>User select a test case. |
| Post-conditions | Brake Performance follow up instructions show up in Server 1.<br>User proceeds to follow instructions.<br>Brake Performance gets active. |
| List of Exception Use Cases | |
| Interfaces | Client HMI, Server1. |


### *3.3.3.2  TRAP-UC-REQ-414527/A-Stop Brake Performance*

| Actors | User |
|---|---|
| Pre-conditions | Brake Performance test is in progress |
| Scenario Description | User selects to cancel brake performance in client HMI. |
| Post-conditions | Brake Performance test gets interrupted and does not complete.<br>No data is sent from Server 1 to Client. |
| List of Exception Use Cases | |
| Interfaces | Client HMI |

### 3.3.3.3   *TRAP-UC-REQ-414528/A-All Time Best Brake Performance*

| Actors | User |
| --- | --- |
| **Pre-conditions** | Brake Performance in progress |
| **Scenario Description** | Brake Performance test ends. A new best data set is generated. Server1 sends the test data to client. |
| **Post-conditions** | Client displays the new timer as Best for the test category. |
| **List of Exception Use Cases** | |
| **Interfaces** | Client HMI |

## 3.3.4   White Box Views

### 3.3.4.1   *Activity Diagram*

### 3.3.4.1.1   TRAP-ACT-REQ-415939/A-Brake Performance Operation
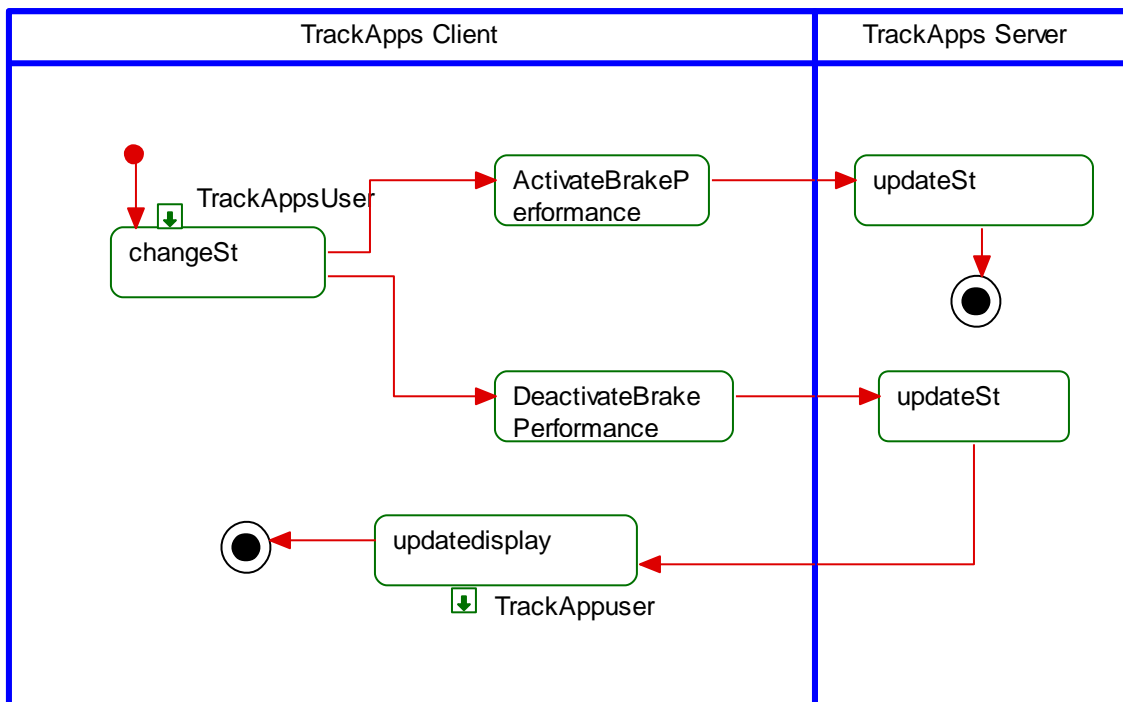
### 3.3.4.2   Sequence Diagram

#### 3.3.4.2.1   TRAP-SD-REQ-415943/A-Brake Performance Operation
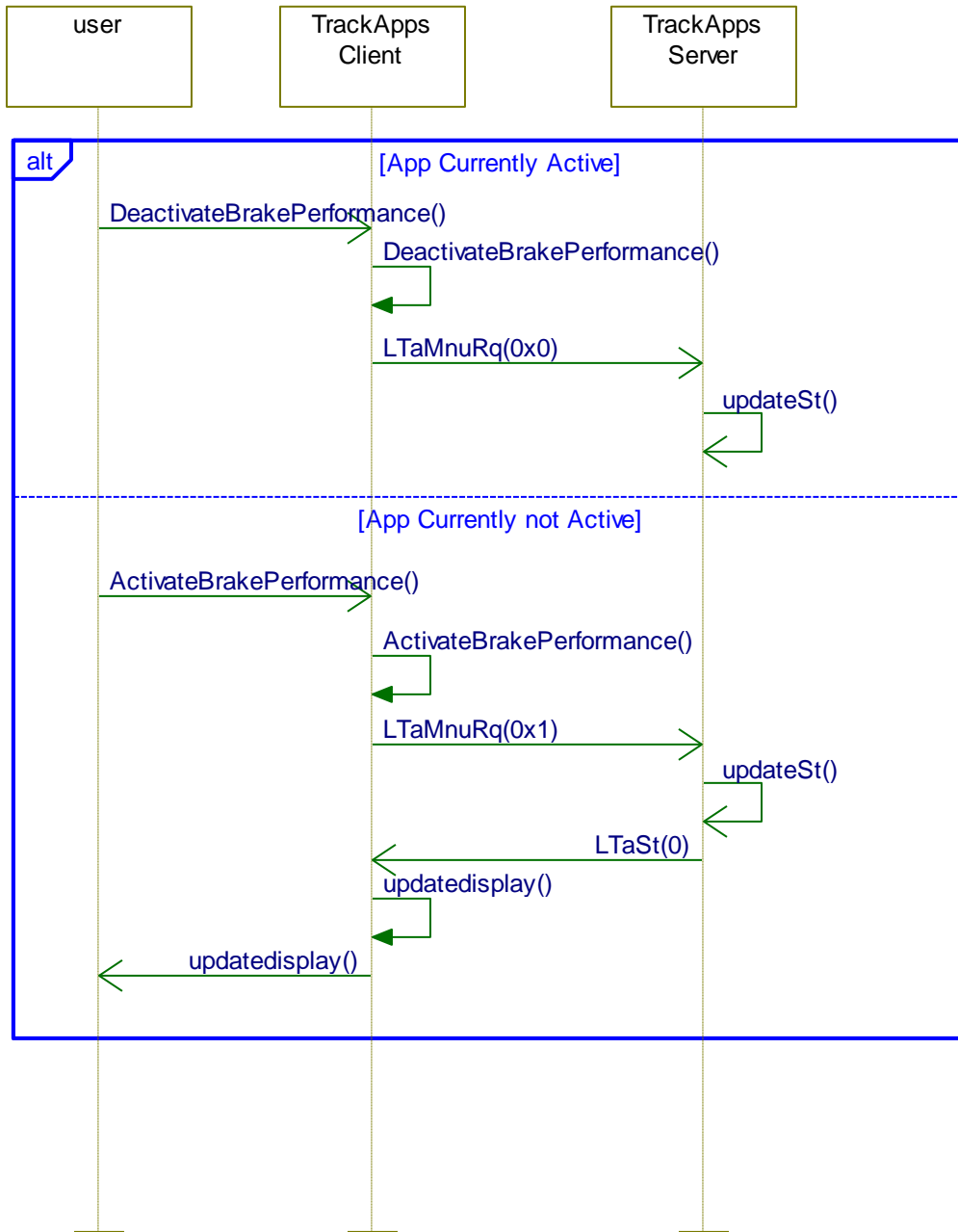


## 3.4   TRAP-FUN-REQ-413702/A-Line Lock

### 3.4.1   Overview

Line lock is a performance feature controlled by the AdvanceTrac system that allows the vehicles front brakes to be applied while the rear brakes are free (no brake pressure) without driver brake input.  This feature would be initiated by the driver and allow rear wheel spin while the front brakes are holding the vehicle stationary. This feature has been installed in performance RWD vehicles for decades using a solenoid on the front brake circuit. Traditionally, line lock is used to heat the rear tires for racing.

**3.4.2 Requirements**

*3.4.2.1 TRAP-REQ-413691/A-Line Lock Availability*

Line Lock could have a configuration bit, properly recorded in diagnostics specs of client, indicating if a vehicle has this feature active or not.

*3.4.2.2 TRAP-REQ-416338/A-Line Lock Applicable Parameters*

Applicable parameters for Line Lock function in LTaMnuRq method are:

| | |
|---|---|
| **Deactivate_TrackApp** | 0x0 |
| **Activate_LineLock** | 0x10 |
| | |

*3.4.2.3 TRAP-REQ-416339/B-Activate / Deactivate Line Lock*

Client needs to send out TrakAppl_D_RqMnu(Line Lock state) to 0x10 as long as BrkLineLck_D_Stat is Initiated and engaged(BrkLineLck_D_Stat Is 0x1 or 0x2). For everything else of BrkLineLck_D_Stat(0x0, 0x3 , 0x4…etc), the Client should send out TrakAppl_D_RqMnu(Line Lock state) set to 0x0.

Client shall not transmit LTaMnuRq as 0x0 when LLineLckSt is 0x2,

*3.4.2.4 TRAP-REQ-416341/B-Line Lock Status*

The signal LLineLckSt provides the status of the feature. Its parameters control the feature like below.

When LLineLckSt is 0x0, the feature should be displayed as Off and the user should be able to request to turn feature On.

When LLineLckSt is 0x1, the feature should be displayed as On and the user should be able to request to turn feature Off.

When LLineLckSt is 0x2, the feature should be displayed as On but the user should not be able to turn the feature Off

When LLineLckSt is 0x3, the feature should be displayed as Off and unavailable to be enabled or interface with the user.

When LLineLckSt is 0x4 or 0x5 or 0x6 or 0x7, the feature should be displayed as Off and unavailable to be enabled or interface with the user.

States in power Up:
APIM should not save the state of Line Lock on power-off or battery disconnect, On boot expected to send out TrakAppl_D_RqMnu(Line Lock state) set to 0x0.

**3.4.3 Use Cases**

*3.4.3.1 TRAP-UC-REQ-416342/A-Activate Line Lock*

| **Actors** | Vehicle user |
|---|---|
| **Pre-conditions** | Vehicle equipped with Track apps. |
| | Track apps is Off and available for user to interface with it |

| **Scenario Description** | User requests Line Lock to be turned On. |
|---|---|
| **Post-conditions** | User follows the instructions in Server 1.<br>Line lock gets enabled and it is being executed. |
| **List of Exception Use Cases** | |
| **Interfaces** | Client HMI, Server 1 HMI |

### *3.4.3.2  TRAP-UC-REQ-416343/A-Deactivate Line Lock*

| **Actors** | Vehicle user |
|---|---|
| **Pre-conditions** | Vehicle equipped with Track apps.<br>Track apps is On and available for user to interface with it |
| **Scenario Description** | User requests Line Lock to be turned Off. |
| **Post-conditions** | Line Lock gets turned Off. |
| **List of Exception Use Cases** | |
| **Interfaces** | Client HMI, Server 1 HMI |

### 3.4.4 White Box Views

#### *3.4.4.1 Activity Diagram*

#### 3.4.4.1.1 TRAP-ACT-REQ-415940/A-Operate Line Lock
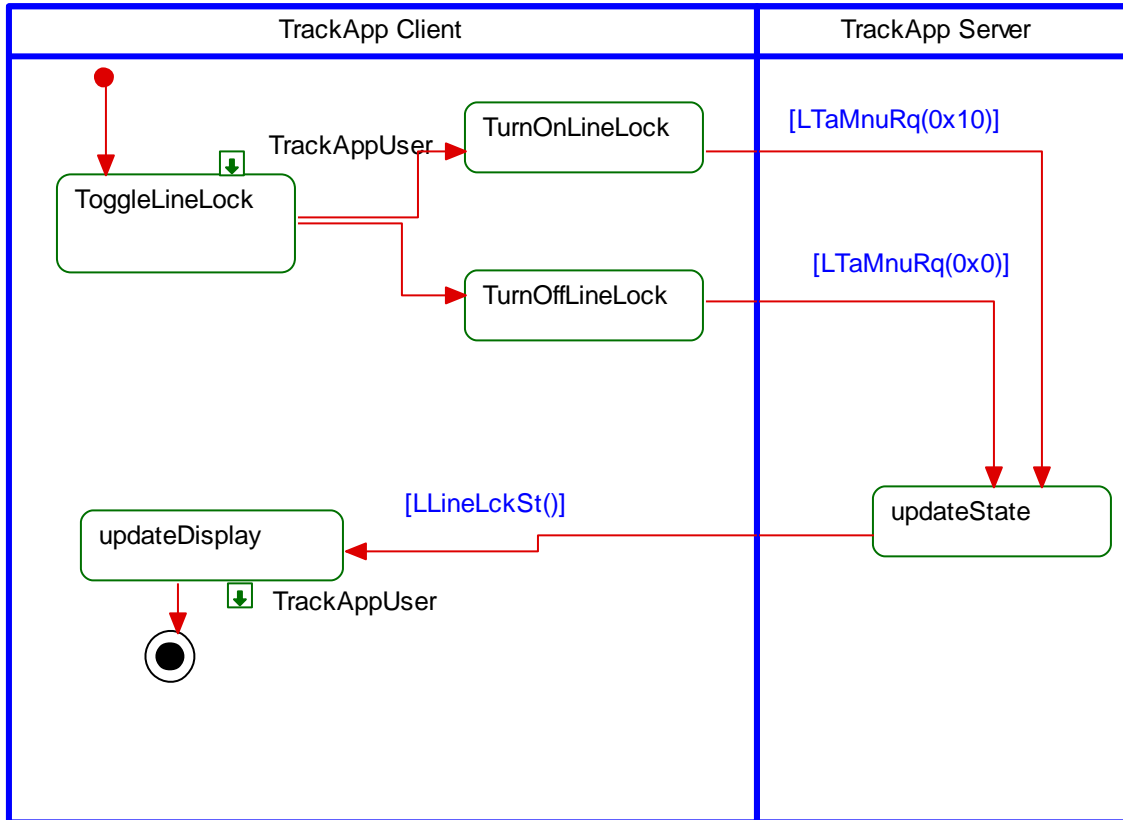
### 3.4.4.2    *Sequence Diagram*

### 3.4.4.2.1    TRAP-SD-REQ-415944/A-Operate Line Lock

## 3.5   TRAP-FUN-REQ-413703/A-Lap Timer

### 3.5.1   Overview

The purpose of this feature is to provide a convenient option for the customer to measure and record lap times on a track of their choice. The user request is done from the client and the user proceeds to follow applicable instructions in Server 1.

The server will Not notify the client of active/engaged feature status. The client shall get update when user cancels through the server1 and will be able to indicate canceled status.

### 3.5.2   Requirements

#### 3.5.2.1   *TRAP-REQ-413779/A-Lap Timer Applicable Parameters*
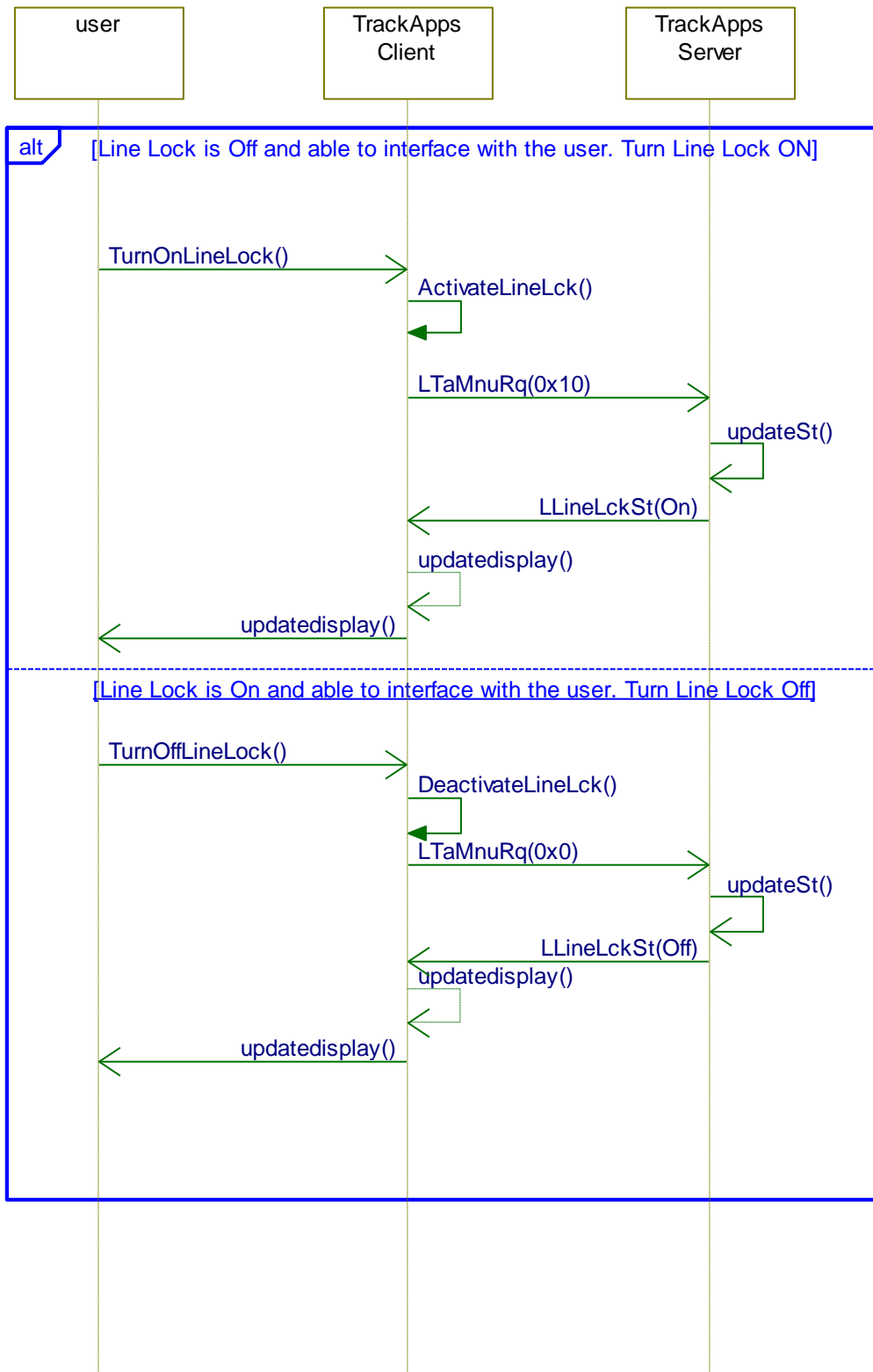
Applicable parameters for Lap Timer function in LTaMnuRq method are:
0x0 and the value 0xF.

| | |
|---|---|
| Deactivate_TrackApp | 0x0 |
| Activate_LapTimer | 0xF |

The client shall transmit the value 0xF upon user requesting that particular test. The value should be transmitted for as long as the feature is active.

#### 3.5.2.2   *TRAP-REQ-413780/A-Lap Timer Availability*

Lap Timer could have a configuration bit, properly recorded in diagnostics specs of client, indicating if a vehicle has this feature active or not.

#### 3.5.2.3   *TRAP-REQ-413781/A-Deactivate Lap Timer*

User should be able to deactivate the lap timer operation at any stage of the feature. To do this, client should allow the user to send LTaMnuRq with the value of 0x0.

The value 0x0 is also sent when the user has not made any request for lap timer execution.

#### 3.5.2.4   *TRAP-REQ-414582/A-Data Set And Content*

Each session can have up to 25 data sets, for a combined total of 100 data sessions. Among the data that it needs to store are Lap Time, Maximum Speed of Lap, Average Speed of Lap and Lap Timer.

#### 3.5.2.5   *TRAP-REQ-414597/A-Unit Convention Lap Timer*

Client should display units of Lap Timer based on the units that it is configured. Options of units display are in kph (Metric) or mph (English).

#### 3.5.2.6   *TRAP-REQ-414610/A-Test Event Selection*

As mentioned earlier, there are multiple sessions of Lap Timer. The user starts the feature by selecting one of the track. When test is over, the data set generated shall be stored in the memory location allocated to the user input session.

#### 3.5.2.7   *TRAP-REQ-414611/A-Data Set Clearance*

Once all dataset has been written, then the oldest data is cleared to make space for newer test data. If user removed data manually, the client should use the freed space before it overwrites in old data. User should be able to clear an entire data session.

#### 3.5.2.8   *TRAP-REQ-416166/A-Server 2 Side Feature Termination*

The signal LTaSt is used to provide the client with the feature status on Server 2 side. The values LTaSt (0x1) and LTaSt (0x2 )  (Cancelled and Deactivated) mean that the feature has been terminated by the user.

The client should set the values of LTaMnuRq to 0x0 and LTaSetRq to 0x0 .

### *3.5.2.9   TRAP-REQ-416169/A-Best Lap Timer*
Best data is the shortest test drive.

### *3.5.2.10   TRAP-REQ-416170/A-Average Speed*
Average Speed is calculated by dividing total distance traveled with total time traveled for the test.

### **3.5.2.11   Client And Server 1 Communication Situations**

#### 3.5.2.11.1   TRAP-REQ-418393/A-Not Active
Not Active is the state when the feature has not been activated by the user from the client. The client should transmit the below:

> LTaMnuRq = 0x0 ('Deactivate_TrackApp')
> LTaSetRq = 0x0 ('Null')

#### 3.5.2.11.2   TRAP-REQ-418394/A-Activate
Activate is the state when the user activated the feature from the client. At this time the client shall transit the below:
LTaMnuRq = applicable state of active feature. Refer to requirement 413673 for a list fo options.
LTaSetRq = applicable state of active feature. Refer to requirement 413530 for a list of options.

#### 3.5.2.11.3   TRAP-REQ-418395/A-Deactivate
If the feature is deactivated via the Client, the Client should transmit the below signals:
LTaMnuRq = 0x0 ('Deactivate_TrackApp')
LTaSetRq = 0x0 ('Null')

#### 3.5.2.11.4   TRAP-REQ-418396/A-Cancel
If the feature is cancelled, as indicated by Server 1 with the signal states below,
LTaSt ('Cancelled') or LTaSt ('Deactivated')

the client shall stranmit the signals like below:
LTaMnuRq = 0x0 ('Deactivate_TrackApp')
LTaSetRq = 0x0 ('Null')

### **3.5.3   Use Cases**

### *3.5.3.1   TRAP-UC-REQ-414631/A-Activate Lap Timer*

| **Actors** | User |
|---|---|
| **Pre-conditions** | Vehicle is equipped with Lap Timer |
| **Scenario Description** | User activates Lap Timer in Client screen |
| **Post-conditions** | Server1 takes over and follows up with instructions for the user. |
| **List of Exception Use Cases** | |
| **Interfaces** | Client HMI |

### 3.5.4 White Box Views

#### 3.5.4.1 Activity Diagram

#### 3.5.4.1.1 TRAP-ACT-REQ-415941/A-Lap Timer Activate/Deactivate

FILE: TRACK APPS SPSS V1.1 SEP 23, 2022

FORD MOTOR COMPANY CONFIDENTIAL
The information contained in this document is Proprietary to Ford Motor Company.

Page 30 of 37
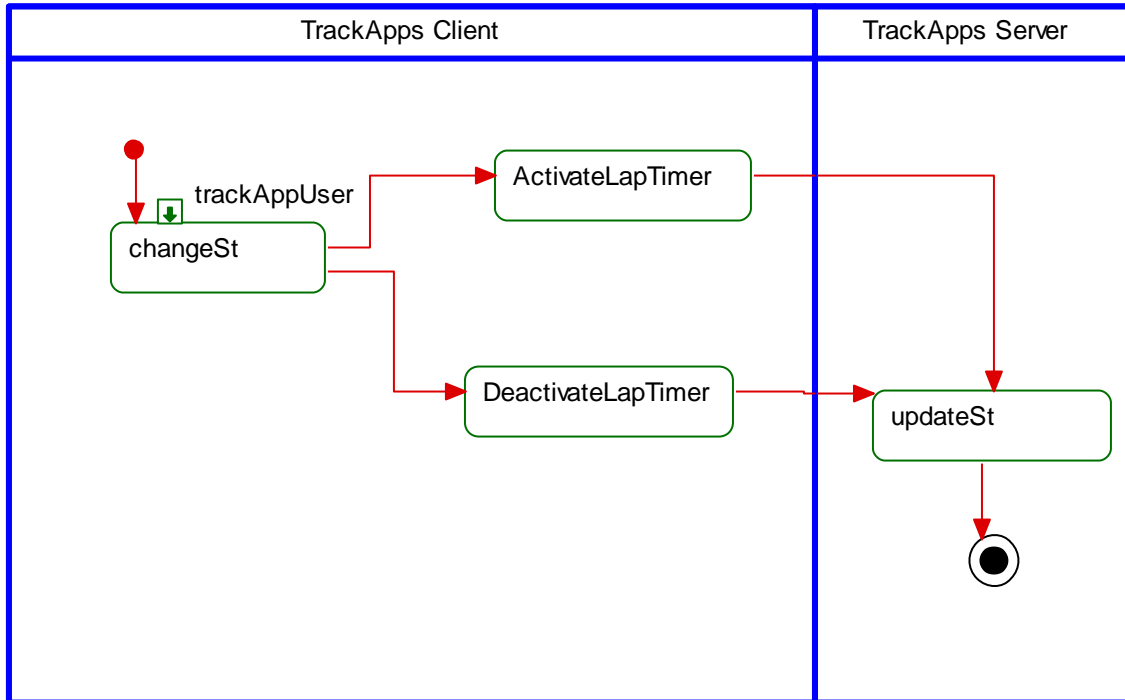
### *3.5.4.2 Sequence Diagram*

#### 3.5.4.2.1 TRAP-SD-REQ-415945/A-Lap Timer Activate/Deactivate
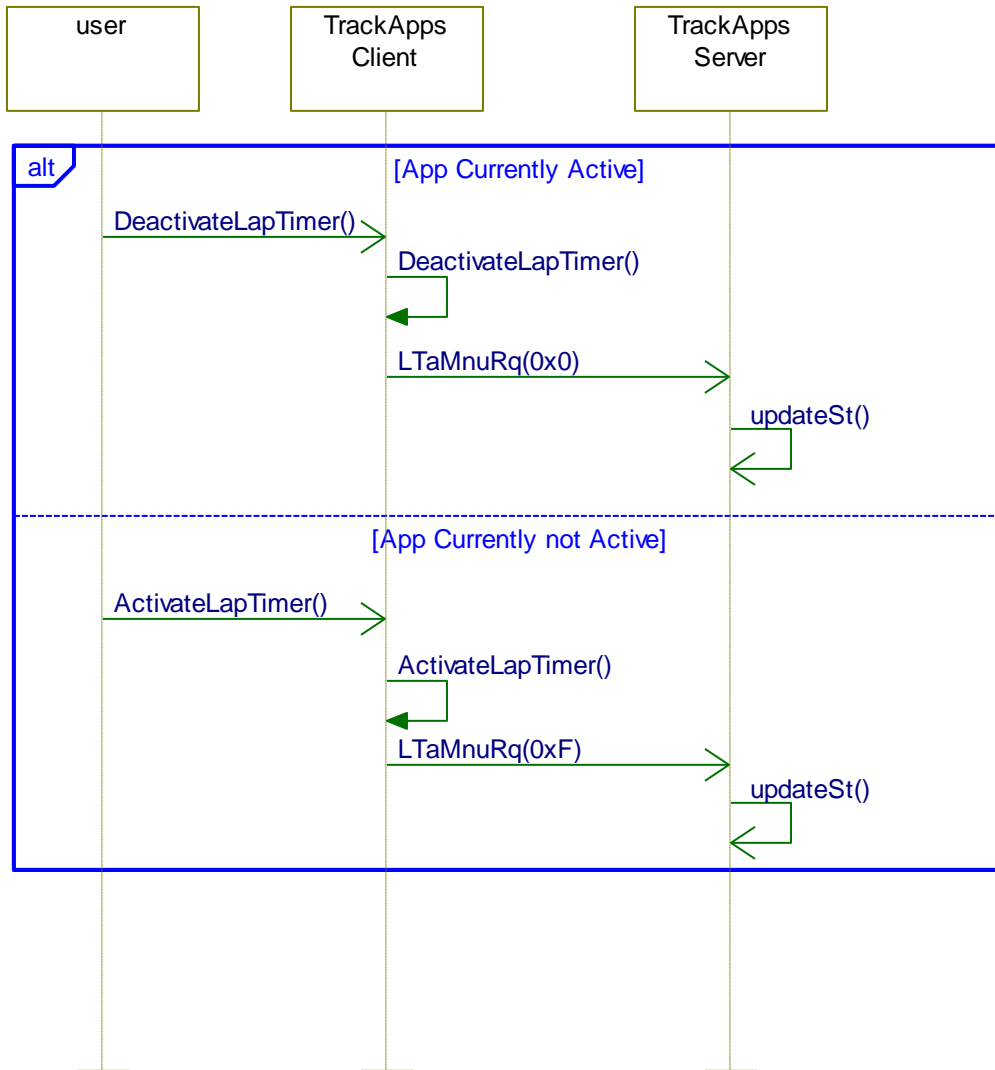


## 3.6 TRAP-FUN-REQ-413796/A-Transport Protocol Content

### 3.6.1 Overview

There is considerable data transfer between Client and Server1. When feature was designed Transport Protocol (TP) was chosen to be used to carry this data since it provided flexibility with data sizes, and convenience in approvals between the various module stakeholders.

The section below will include requirements about data definitions and data organization in the Transport protocol. Additional information is also added and provided through Transport Protocol SPSS. Refer to it for additional requirements and complete understanding of the requirements.

### 3.6.2     Requirements

#### *3.6.2.1     TRAP-REQ-413798/A-Data Types And Definitions*

Data being sent through TP is made of some basic structures. Structures include Time, Speed, Distance traveled etc. This section will try to define these structures, the resolution and bit length, so common documentation is shared between Server and Client, thus making operating on the same data reliable and error free.

#### 3.6.2.1.1   TRAP-REQ-413799/A-Timer

Timer is structure that is used to store time value. This time value when processed, provides the physical time it took for an event to happen.

There are 5 timers, Timer1 - Timer5.
Timer1 is 16 bits long.
Timer2 – Timer5 are 14 Bits long.
Timer resolution is 0.01s.

| | **Timer1** | **Timer2** | **Timer3** | **Timer4** | **Timer5** |
| --- | :---: | :---: | :---: | :---: | :---: |
| Resolution: | 0.01s | 0.01s | 0.01s | 0.01s | 0.01s |
| Size (bits) | 16 | 14 | 14 | 14 | 14 |
| Actual Range: | 0-655.35s | 0-163.83 | 0-163.83 | 0-163.83 | 0-163.83 |

#### 3.6.2.1.2   TRAP-REQ-413801/A-Speed

Speed is a structure that stores Speed value. When this value is processed properly, it indicates the speed value that was recorded while testing driving a particular feature.
There are 5 speed value. Speed1 to Speed5.
The resolution is 1. If vehicle units are Metric, then the smallest value means 1 Kph. If the units are English, the smallest unit is 1 mph.

All 5 speeds have a size of 9 bits long.

| | **Speed1** | **Speed2** | **Speed3** | **Speed4** | **Speed5** |
| --- | :---: | :---: | :---: | :---: | :---: |
| Resolution: | 1kph / mph | 1kph / mph | 1kph / mph | 1kph / mph | 1kph / mph |
| Size (bits) | 9 | 9 | 9 | 9 | 9 |
| Actual Range: | 0-511kph / mph | 0-511kph / mph | 0-511kph / mph | 0-511kph / mph | 0-511kph / mph |

#### 3.6.2.1.3   TRAP-REQ-413802/A-Distance

Distance is a structure that is used to provide distance value.
There are two distances, Distance1 and Distance2.

Resolution is 1, however if vehicle is in Metric system, it will mean 1 meter. If vehicle is in English unit, it will mean 1 ft.
Distance1 and Distance2 are 14 bits long.

| | **Distance1** | **Distance2** |
| --- | :---: | :---: |
| Resolution: | 1 m / ft | 1 m / ft |
| Size (bits) | 14 | 14 |
| Actual Range: | 0-16383 m / ft | 0-16383 m / ft |

### 3.6.2.1.4   TRAP-REQ-413803/A-Acceleration

Acceleration is a structure that is used to provide acceleration value. There are two accelerations, Accel1 and Accel2. Accel1 and Accel2 are 8 bits long.

|  | **Accel1** | **Accel2** |
|---|---|---|
| Resolution: | 0.01 G | 0.01 G |
| Size (bits) | 8 | 8 |
| Actual Range: | (0, 2.55) G | (0, 2.55) G |

### 3.6.2.1.5   TRAP-REQ-413804/A-Counter

Counter is a structure that is used to provide counter content. It provides an event counter.
There is only one counter, Counter, which has a resolution of 1 and a size of 5 bits.

|  | **Counter (Lap, or Session#)** |
|---|---|
| Resolution: | 1 |
| Size (bits) | 5 |
| Actual Range: | 0-31 |

### *3.6.2.2   REQ-413805/A-Test Types*

Depending on the test type, various combination of data structures could be generated and transmitted through TP. The section below will try to explain the possible test types and how to organize such data.

The test event comes from feature type being tested. Such information is conveniently described in Byte 4 of TP content.

### 3.6.2.2.1   TRAP-REQ-413806/A-Acceleration Timer Speed Based

Possible Acceleration Timer Speed Based data combination is provided below. Again keep in mind the option to use English or Metric units, is done based on the vehicle settings configuration.

| Test Event Selection | Test Event Data | Transport Protocol Data Assignment | Bits length |
|---|---|---|---|
| 0-30 mph | | | |
|  | 0x0 | Test Event Identifier | |
|  | 0-30mph Time | Timer1 | 16 |
|  | | | Total bits 16 |
| 0-50 kph | | | |
|  | 0x1 | Test Event Identifier | |
|  | 0-50kph Time | Timer1 | 16 |
|  | | | Total bits 16 |
| 0-60 mph | | | |
|  | 0x2 | Test Event Identifier | |
|  | 0-30mph Time | Timer1 | 16 |
|  | 0-60mph Time | Timer2 | 14 |
|  | | | Total bits 30 |
| 0-100 kph | | | |
|  | 0x3 | Test Event Identifier | |
|  | 0-50kph Time | Timer1 | 16 |
|  | 0-100kph Time | Timer2 | 14 |
|  | | | Total 30 bits |
| 0-100 mph | | | |
|  | 0x4 | Test Event Identifier | |
|  | 0-30mph Time | Timer1 | 16 |

| | | | |
| --- | --- | --- | --- |
| | 0-60mph Time | Timer2 | 14 |
| | 0-100mph Time | Timer3 | 14 |
| | | | Total 44 bits |
| **0-200 kph** | | | |
| | 0x5 | Test Event Identifier | |
| | 0-50kph Time | Timer1 | 16 |
| | 0-100kph Time | Timer2 | 14 |
| | 0-200kph Time | Timer3 | 14 |
| | | | Total 44 bits |

How to read the table:

If the user chooses to execute 0-200kph, the Server will collect 3 data points. The 1st will be stored in Timer1 and it will indicate the time it took for the vehicle to reach 50kph. The Timer2 indicates the time it took for the vehicle to reach 100kph and the Timer3 is used to indicate the time it took for the vehicle to reach 200kph. All these 3 timers , 16 bits, 14 bits and 14 bits are packed together in a serialized fashion and transmitted through TP with the other necessary bytes.

### 3.6.2.2.2   TRAP-REQ-413807/A-Acceleration Timer Distance Based

Acceleration timer distance based is another test type that the feature can execute. The table for these types of tests and necessary data set is provided below.

| **Test Event Selection** | **Test Event Data** | **Transport Protocol Data Assignment** | |
| --- | --- | --- | --- |
| **1/8 mile** | | | |
| | 0x6 | Test Event Identifier | |
| | 60ft Time | Timer1 | 16 |
| | 330ft Time | Timer2 | 14 |
| | 1/8mi Time | Timer3 | 14 |
| | 60ft Max Speed | Speed1 | 9 |
| | 330ft Max Speed | Speed2 | 9 |
| | 1/8mi Max Speed | Speed3 | 9 |
| | | | Total bits 71 |
| **200 meters** | | | |
| | 0x7 | Test Event Identifier | |
| | 25m Time | Timer1 | 16 |
| | 100mTime | Timer2 | 14 |
| | 200m Time | Timer3 | 14 |
| | 25m Max Speed | Speed1 | 9 |
| | 100m Max Speed | Speed2 | 9 |
| | 200m Max Speed | Speed3 | 9 |
| | | | Total bits 71 |
| **1/4 mile** | | | |
| | 0x8 | Test Event Identifier | |
| | 60ft Time | Timer1 | 16 |
| | 330ft Time | Timer2 | 14 |
| | 1/8mi Time | Timer3 | 14 |
| | 1000ft Time | Timer4 | 14 |
| | 1/4mi Time | Timer5 | 14 |
| | 60ft Max Speed | Speed1 | 9 |
| | 330ft Max Speed | Speed2 | 9 |
| | 1/8mi Max Speed | Speed3 | 9 |
| | 1000ft Max Speed | Speed4 | 9 |
| | 1/4mi Max Speed | Speed5 | 9 |
| | | | Total bits 117 |
| **400 meters** | | | |

| | | | |
|---|---|---|---|
| | 0x9 | Test Event Identifier | |
| | 25m Time | Timer1 | 16 |
| | 100mTime | Timer2 | 14 |
| | 200m Time | Timer3 | 14 |
| | 300m Time | Timer4 | 14 |
| | 400m Time | Timer5 | 14 |
| | 25m Max Speed | Speed1 | 9 |
| | 100m Max Speed | Speed2 | 9 |
| | 200m Max Speed | Speed3 | 9 |
| | 300m Max Speed | Speed4 | 9 |
| | 400m Max Speed | Speed5 | 9 |
| | | | Total bits 117 |

### 3.6.2.2.3   TRAP-REQ-413808/A-Brake Performance

Brake Performance provides a list of Brake Performances test and how the data is structured during such tests.

| Test Event Selection | Test Event Data | Transport Protocol Data Assignment | Total bits |
|---|---|---|---|
| 60-0 mph | | | |
| | 0xA | Test Event Identifier | |
| | 60-0mph Stopping Distance | Distance1 | 14 |
| | 60-0mph Peak Deceleration | Acceleration1 | 8 |
| | | | Total Bits 22 |
| 100-0 kph | | | |
| | 0xB | Test Event Identifier | |
| | 100-0kph Stopping Distance | Distance1 | 14 |
| | 100-0kph Peak Deceleration | Acceleration1 | 8 |
| | | | Total Bits 22 |
| 100-0 mph | | | |
| | 0xC | Test Event Identifier | |
| | 60-0mph Stopping Distance | Distance1 | 14 |
| | 100-0mph Stopping Distance | Distance2 | 14 |
| | 100-0mph Peak Deceleration | Acceleration2 | 8 |
| | | | Total Bits 36 |
| 200-0 kph | | | |
| | 0xD | Test Event Identifier | |
| | 100-0kph Stopping Distance | Distance1 | 14 |
| | 200-0kph Stopping Distance | Distance2 | 14 |
| | 200-0kph Peak Deceleration | Acceleration2 | 8 |
| | | | Total Bits 36 |

### 3.6.2.2.4   TRAP-REQ-413809/A-Lap Timer

Lap Timer provides the necessary details of how data is organized when Lap Timer test is done.

| Test Event Selection | Test Event Data | Transport Protocol Data Assignment | |
|---|---|---|---|
| Lap Timer (English) | | | |
| | 0xE | Test Event Identifier | |
| | Lap Time | Timer1 | 16 |
| | Maximum Speed of Lap | Speed1 [mph] | 9 |

| | Average Speed of Lap | Speed2 [mph] | 9 |
|---|---|---|---|
| | Lap Number | Counter | 5 |
| | | | Total bits 39 |
| Lap Timer (Metric) | | | |
| | 0xF | Test Event Identifier | |
| | Lap Time | Timer1 | 16 |
| | Maximum Speed of Lap | Speed1 [kph] | 9 |
| | Average Speed of Lap | Speed2 [kph] | 9 |
| | Lap Number | Counter | 5 |
| | | | Total bits 39 |

# 4   Appendix: Reference Documents

| | |
|---|---|
| 1 | Transport Protocol SPSS |
| 2 | HMI H31p |
| 3 | Infotainment Diagnostic Specification |