Function Specification (FncS)

(In Vehicle Software Update Vehicle FIS)

Document Type	Function Specification (FncS)	
Document ID	547918	
Document Location	VSEM Rich Client, VSEM Active Workspace	lgX5oHybx3NrTD
Document Owner	Tummepalli, Amareswar (atummepa)	
Document Version	F	
Document Status	Released	
Date Issued	09-Apr-2020 10:26	
Date Revised	24-Jul-2020 11:32	
Document Classification	GIS1 Item Number:	
	GIS2 Classification:	

Document Approval			
Person	Role	Email Confirmation	Date

Printed Copies are Uncontrolled

Content

List of Figures

1 Introduction

1.1 Purpose

The Feature Implementation Specification (FIS) specifies the deployment of the logical functions of a feature to an electrical architecture. The FIS specifies all interactions between the ECUs of the electrical architecture required for the feature including the technical signals and the interfaces. It also gives interface and integration requirements, which are specific to the feature for the electrical architecture.
To get more information about the concept of feature, function and component level abstraction refer to the Ford RE Wiki.

1.2 Scope

This FIS describes the deployment of the IN VEHICLE SOFTWARE UPDATE feature to the following electrical architecture(s):

Electrical Architecture Name	Owner	Reference
FNV2 – Fully Network Vehicle	Gwen Ald	<Add VSEM Link>
CGEA1.3C	Gwen Ald	

Table 1: Electrical Architecture(s) referenced in this document

The following functions from the Global Feature & Function List are referenced in this Feature Implementation Specification:

Function ID	Function (Group) Name	Owner	Reference
<Add VSEM ID>	CAVC Function Specification	Vijay Jayaraman	<Add VSEM Link>
	OVTP OTA Function Definition	Mohamad Nasser	

Table 2: Functions referenced in this document

1.3 Audience

The FIS is authored by CVS IVSU Team. All Stakeholders, i.e., all people who have a valid interest in the feature implementation should read and, if possible, review the FIS. It needs to be guaranteed, that all stakeholders have access to the currently valid version of the FIS.

1.3.1 Stakeholder List

For the latest list of the function stakeholders and their roles & responsibilities refer to <Put VSEM Link here>.

Name	CDSID	Responsibilities
Jim Weinfurther		Body Control Technical Specialist
Jeremy Rusell		PCM Technical Specialist
Gwen Ald		EE Architecture System Lead
Jason Miller		OVTP and Diagnostic Technical Specialist
Jennifer Shaw		ECG Supervisor
Aldi Caushi		Cyber Security Functional Owner
Bill Waldeck		NetCom Technical Specialist

Name	CDSID	Responsibilities
Scott Watkins		Cluster Technical Specialist

Table 3: Stackholder List

1.4 Document Organization

1.4.1 Document Context

Refer to the Specification Structure page in the Ford RE Wiki to understand how the FIS relates to other Ford Requirements Documents and Specifications.

1.4.2 Document Structure

The structure of this document is explained below:

Section 1 – Introduction – Giving an explanation how to use this document including responsibilities and the scope of the document. Additionally, it contains the revision history and a list of unsettled but known issues that have to be consolidated in future versions. It explains the terminology and gives a clarification of the definitions, concepts and abbreviations used in the document.

Section 2 – Platform Description – Giving an overview of the platform and listing assumptions, constraints or dependencies

Section 3 – Functional Architecture – Showing the logical architecture of functions

Section 4 – Function Deployment – Presenting the allocation of logical functions and signals to their electrical counter parts

Section 5 – Deployment Specific Modeling –Modeling techniques providing additional detail on e.g. interface behavior

Section 6 – Deployment Specific Requirements – Deployment specific requirements for ECUs, Network Communication, and Process

Section 7 – List of Open Concerns

Section 8 – Traceability Matrix

Section 9 – Revision History

Section 10 – Appendix - Presenting additional data mainly in a tabular form, e.g., a data dictionary

1.5 References

1.5.1 Ford Documents

Lst here all Ford internal documents, which are directly related.

Reference	Title	Doc. ID	Revision
1	OVTP OTA Function Definition		
2	OVTP Protocol Specification		
3	OTA Signed Commands		
4	Application Signing Requirements		
5	ESN Specification		
6	SWDL		
7	IVSU Feature Document		
8	IVSU_Vehicle_Function_Bare Metal Diff Updater		
9	IVSU FNV2 DFMEA	Please refer to LFMA documentId: 66689	
10	IVSU Functional safety requirements		
11	IVSU_Vehicle_Function_Diff Generator		

Table 4: Ford internal Documents

1.5.2 External Documents and Publications

The list of external documents could include books, reports and online sources.

Reference	Document / Publication

Table 5: External documents and publications

1.6 Terminology

1.6.1 Definitions

Definition	Description

Table 6: Definitions used in this document

1.6.2 Abbreviations

Abbr.	Stands for	Description
FS	Function Specification	
E/E	Electrical and Electronics	
FIS	Feature Implementation Specification	
IVSU	In Vehicle Software Update	
FESN	Ford Electronic Serial Number	Ford-specific ECU serial number used for OTA and security purposes
OTA	Over The Air	
DID	Diagnostic Data Identifier	

Table 7: Abbreviations used in this document.

2 Feature Implementation Description

2.1 Overview

Software updates for all vehicle's component is a way to reduce the warranty cost and improve the vehicle's functionality.
In Vehicle Software Updates feature provides the ability to re-flash the vehicle without the customer being required to go to a Ford dealer and service her car. There are two methods that the software gets to the vehicle: via OTA or via USB.
Ford owner's website or Ford Customer Service site are the only locations where the software can be downloaded into a USB.
OTA (Over the Air) will use the vehicle connectivity to download the software directly in the vehicle. The highest priority is Home Wi-Fi, then AppLink, then Cellular which is paid by Ford. However, the priorities can be modified based on Ford's requirement per each software update.
Once the software is present in the vehicle, the ECG module shall use Ethernet or CAN/CAN FD to distribute the software to the other entire vehicle ECUs.
To reduce the possibilities of permanent failure each component shall have double memory to keep the previous software in addition to the new software that is re-flashed with. The double memory is needed so that modules can revert back to the previous software in case of failures.

2.2 Input Requirements

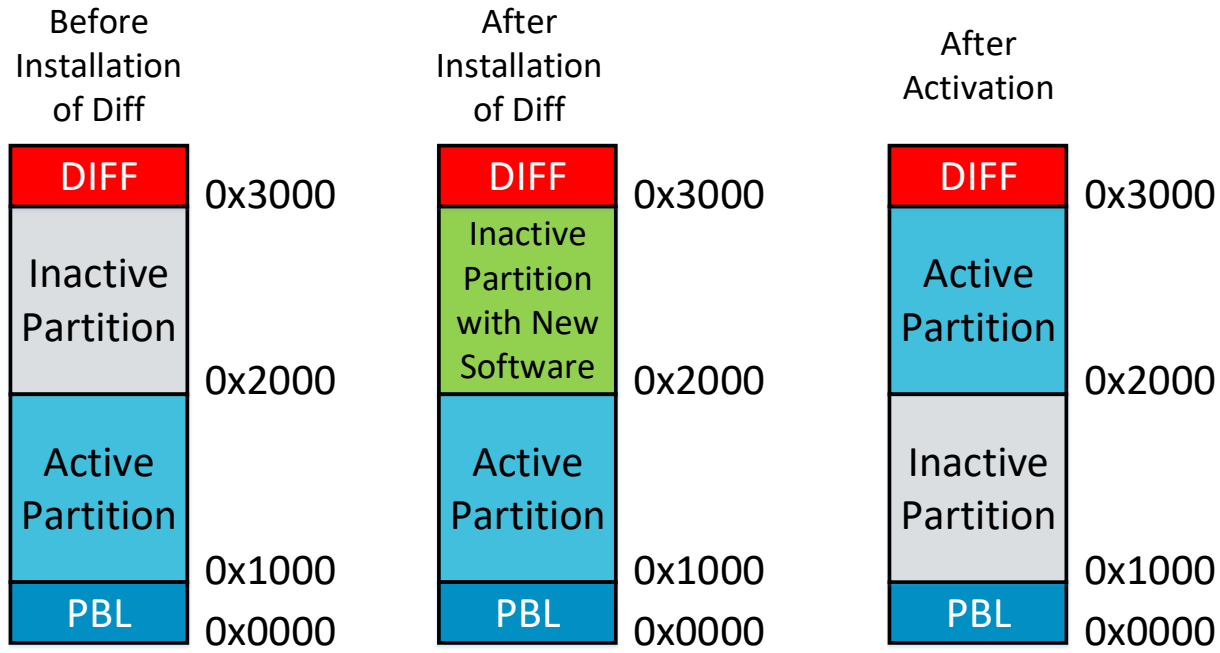
2.2.1 FRD-REQ-308047/B-###R_CMP_IVSU_V_00002### DIDs for OTA Command Signing Keys and Application Signing Keys

The hash of the OTA command signing key and the hash of the in-use application signing key shall be verified at Ford EOL by reading DID 0xD03E and 0xD03F.

2.2.2 FRD-REQ-308048/C-###R_CMP_IVSU_V_00003### Differential Updater

OTA supported ECUs shall support differential file updates If total programmable software size is larger than 1MB. (Ref 8, Ref 11)

Note: If an ECU supports differential files, the compiler settings shall be investigated to optimize the effectiveness of the differential generation.



In addition to the steps of a regular OVTP update, a Diff is programmed into a reserved region of memory. Then an additional step of installing the Diff is performed. Once the Diff installation is complete, the new software is present in the inactive memory and is ready to be remapped as described in OTA Function Definition Reference 1.

High Level Requirements:

- Hardware assisted memory remapping
- 2x internal flash to support storage of both A & B memory
- Additional reserved internal flash Diff Memory, which at least 30% of the size of A.
- Read-while-write capability to internal flash

2.2.3 FRD-REQ-308049/A-###R_CMP_IVSU_V_00004### Number of Software Updates

ECU shall support software update capability over the life of the vehicle (10yr/150K miles), assuming 5 re-flashes per year (50 total).

2.2.4 FRD-REQ-308050/B-###R_CMP_IVSU_V_00005### Temporary Vehicle Storage for Software Files

OTA client module shall support storage of 1 GB of OTA files for download.

2.2.5 FRD-REQ-308052/C-###R_CMP_IVSU_V_00007### Maximum ECU Activation Time

For OVTP OTA modules, the worst case allowed activationTime (Ref 1) for the initiateActivation command is 90s. For OVTP OTA modules, the worst case allowed rollbackTime (Ref 1) for the initiateRollback command is 90s. This is the worst cast time based upon nominal operating conditions and not the worst case time based upon extreme operating conditions, etc.

2.2.6 FRD-REQ-308053/B-###R_CMP_IVSU_V_00008### Component Hardware Review

Every OVTP OTA ECUs that requires an activationTime or rollbackTime (Ref 1) greater than 70s shall complete a deep dive review with the CVPP IVSU team. These components shall strive for technology improvements in their hardware to reduce the activation and rollback time.

2.2.7 FRD-REQ-308054/B-###R_CMP_IVSU_V_00009### Downloading in background

An ECU must be capable of downloading and storing a completely new set of all application software while the existing application software is running as normal. This background download shall not impact the ECU's normal application functionality performance requirements.

2.2.8 FRD-REQ-358773/A-Pause and Resume OTA Progress for OVTP OTA Modules

Every OVTP OTA ECU which supports an individual OVTP OTA function which cannot complete the request/response transaction within 5 minutes shall be capable of pausing the functionality when the network goes to sleep (if a sleep/wake ECU) or when the customer keys off (if a switched ECU). Furthermore, the OVTP OTA ECU shall be capable of resuming the functionality from where it was paused when the same OVTP OTA function is re-requested by the OTA client. For example, if the prepareActivation function for an ECU were to take 7 minutes as the worst case, this functionality must be capable of being paused as describe and resumed from that point (e.g., at the next key cycle). The intent of this is to ensure that progress toward completion of OTA will be made for customers who typically have short drive times of 5 minutes, Exceptions to this are possible but require review and explicit approval of the details of the design by the core IVSU OTA team.

2.2.9 FRD-REQ-308055/A-###R_CMP_IVSU_V_00010### Software Signing

All software downloaded via OTA shall be signed either by application signing, traditional signing or any other signing that is defined by Ford Security Team.

2.2.10 FRD-REQ-308056/B-###R_CMP_IVSU_V_00011### Vehicle Inhibit

The vehicle shall be inhibited for a maximum time of 30 minutes for any combination of non-interruptible OTA activity.
Note:

1. The vehicle shall be inhibited for a maximum time of 2 minutes for OTA over OVTP or Ethernet based SOA SFTP OTA activation.
2. The vehicle shall be inhibited for a maximum time of 4 minutes for DC Configuration.
3. The vehicle shall be inhibited for a maximum time of 15 minutes for SWDL OTA (E/R) Programming.

If Rollback, the timing may be double of mentioned above.

2.2.11 FRD-REQ-308057/B-###R_CMP_IVSU_V_00012### Preserve Data

Each ECU that is re-flashed via OTA or USB shall preserve all the direct ECU Configuration data, or previously learned data, adaptive factors, or other long-term adjustments, etc. Examples of information

that must not be lost after a reset include clock value, radio presets, correct fault gauge level, Bluetooth Pairing info, Seat settings etc
.

2.2.12 FRD-REQ-308060/B-ECUs that can download files from Cloud/USB shall be capable to have local wake up/stay awake

The ECUs (Sync/ECG) capable of downloading files from Cloud/USB on its own, shall have download capability with the location of files to download with the allowed time for the download activity even during Key Off, by keeping itself awake.

2.2.13 FRD-REQ-308061/B-OTA Client shall not request the OTA Run/Start active if ignition_status <> Off

The OTA client shall not request control of the Run/Start bus (e.g., VehOn_D_RqCld <> NoControl) when Ignition_Status <> Off or when VehOnSrc2_D_Stat <> Off.

2.2.14 FRD-REQ-308062/B-OTA Client shall NOT start any OTA Activity if it receives a load shedding signal.

OTA client shall not start any OTA activity if load shedding is active. In the case there is an OTA activity and load shedding transitions to active, the OTA client shall obey the following depending on the OTA activity stage:

1. If performing background download, it shall pause the download until load shedding is no longer active.
2. If the vehicle is inhibited due to an OTA activity, the non-interruptible OTA activity shall complete.

2.2.15 FRD-REQ-308065/B-OTA Client shall NOT initiate or process any OTA activity when Battery is in critical condition

The OTA client shall NOT initiate any OTA activity if the battery is in critical condition (KeyOffMde_D_Actl = Critical Battery). If the vehicle is already inhibited due to an OTA activity, the non-interruptible OTA activity shall complete.

2.2.16 FRD-REQ-324142/C-###R_CMP_IVSU_V_00022### DID for Entering in to OTA ProgrammingSession

DID $D04F shall be supported for all ECUs supporting diagnostics. If an ECU can always enter programmingSession upon request (and therefore has no preconditions), only bit 31 “No ProgrammingSession Preconditions Supported” shall be supported. If an ECU has any preconditions for entering programmingSession due to an OTA initiated event, then bit 31 shall not be supported and bits for each precondition which prevent the transition shall be supported. A reported DID value of all 0s shall always be used to indicate the ECU is able to transition into ProgrammingSession due to an OTA initiated request if asked at the present time. Conversely, a reported DID value with at least one bit not equal to zero requires the ECU to reject a request to transition to programmingSession due to an OTA initiated request.
Support of bits within DID $D04F (i.e., additional entry conditions for programmingSession and OTA activation) shall be kept to a minimum. Support of bits other than bit 31 requires explicit approval from CVPP core IVSU team. If a parameter is defined in the DID (e.g., hazards on) does NOT mean that an

ECU must use that as a precondition. Its presence is because at least one ECU presented a use case to CVPP demonstrating why their particular ECU needs to validate this condition. In other words, because an ECU can determine the state of a parameter in the DID does not mean it needs to implement that as a precondition to prevent OTA activation or programmingSession entry.
When a diagnostic programmingSession entry request is received, it can be recognized as an OTA initiated request by checking if VehOnSrc_D_Stat == OverTheAir OR VehOnSrc2_D_Stat == OverTheAir. DID $D04F shall always report the correct state of all supported bits each time the DID is read independent of the signal value of VehOnSrc_D_Stat and VehOnSrc2_D_Stat.

2.2.17 FRD-REQ-348263/A-Self Install ECU during Load shed

For ECUs with self-installation if they started the installation process and load shed transitioned to active, they shall complete the installation. For ECUs with self-installation if load shedding is active before starting the installation, they shall not start the installation and they shall install the next time the conditions are met (Next ignition cycle or when requested by the OTA client).

2.2.18 FRD-REQ-355940/A-Error Recovery During OVTP OTA Activation and Rollback

This requirement only applies to OTA OVTP ECUs which implement a hardware architecture in which the software is executed out of internal flash, and during OTA OVTP activation and rollback, the active software is temporarily erased and then programmed with new software (e.g., see hardware variants OTA Architecture Type 2 and Type 3).
When performing this activation and rollback, the application software jumps to the bootloader to initiate the erasure of the current software and the programming of the desired new software. This process shall be robust to resets and power interruption, so that if either occurs, the ECU shall be able to retry the erasing of internal flash and programming of the intended software. When performing a swap to different software initiated by the OTA OVTP client, if the ECU is unable to successfully fully program the intended software into internal flash, the ECU shall automatically erase the necessary areas of internal flash and attempt to program the original software stored in the appropriate backup in order to decrease the likelihood of the ECU from being stuck in the bootloader without a valid application.

2.2.19 FRD-REQ-358785/A-DID for Preventing OTA Update of Target ECU but Not Other ECUs

As described in requirement FRD-REQ-324142/C-###R_CMP_IVSU_V_00022###, DID $D04F is a standard DID that is implemented in all ECUs which prevents OTA SWDL programming of any ECU. DID $D031 is a standard DID that is only implemented if an ECU has additional preconditions above and beyond DID $D04F that must be checked in order to prevent the OTA programming of itself. To support DID $D031, underlying rationale for each supported bit must receive explicit approval in writing from the core OTA team within Ford CVPP.
DID $D04F will be read from all ECUs whenever an OTA SWDL event occurs on the vehicle. DID $D031 will only be read from ECUs which are undergoing an actual OTA update.
Examples:
DID $D04F prevents any ECU from being OTA programmed via SWDL
DID $D031 prevents the ECU reporting it from being OTA updated
DID $D04F means your ECU can’t be A/B swapped
DID $D04F means your ECU can’t go into programmingSession (which means no ECU can go into programmingSession).
DID $D031 means your ECU can’t be A/B swapped
DID $D031 means your ECU can go into programmingSession but does not want to be programmed. Onus is on the ECG to not program it.

2.2.20 FRD-REQ-376027/A-All Modules or Target ECUs that supports OTA Shall have the module PART II spec to be up to date in VSEM

All Modules or Target ECUs that supports OTA Shall have the module PART II spec to be up to date in VSEM

2.3 Assumptions & Constraints

1) The IVI components (ECG, SYNC and TCU) have more logical functions that are allocated to them. For easy of representation, the details between those modules are contained in its own FIS.
2) As a design considerations to improve performance of Wifi medium based OTA update following factor shall be considered
    a. Wifi Protocol supported (Say for example: 802.11a - aZ)
    b. Signal strength (Say for example -65dBm) and Proximity (Say for example 5 meters to 250 meters) to WiFi Access Point.

3 Functional Architecture

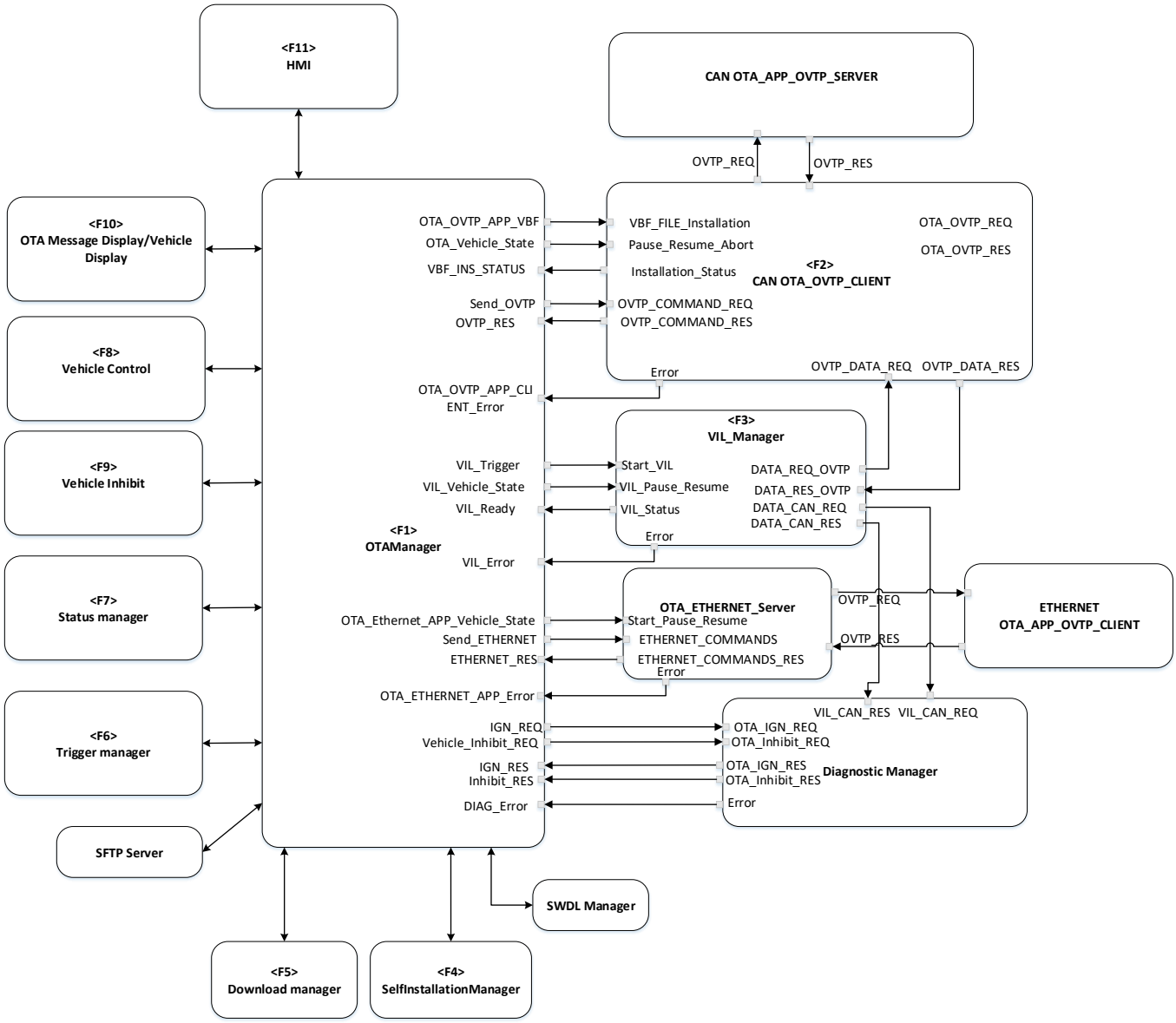


Figure 1: Functional Architecture

Vehicle components that are communicating in CAN, shall be updatable via CAN using the OVTP protocol.

Vehicle components that are communicating in Ethernet, shall be updatable via Ethernet using the OVTP protocol.

Note: L_ECG2ECU_x is many logical signals grouped together for easy presentation in this diagram. Each signal is represented as L_ ECG2ECU_001, L_ ECG2ECU_002,..and so on. L_ECU2ECG_x is many logical signals grouped together for easy presentation in this diagram. Each signal is represented as L_ECU2ECG_001, L_ECU2ECG_002,..and so on. Please refer Signal List Section for comprehensive list of logical signals.

3.1 Function List

Function ID	Function Name	Function Description
F1	IVSU_Vehicle_Function_OTAManager	Function in the ECG that is responsible for orchestrating the OTA update of the vehicle.
F2	IVSU_Vehicle_Function_OTA_OVTP_CLIENT	The function lives in the ECG module which is responsible for updating other ECUs in the vehicle.
F3	IVSU_Vehicle_Function_VILClient	The function lives in the ECG module and is responsible for doing all the diagnostic calls to read the part numbers or other DID information that are required to be reported to Ford's Cloud
F4	IVSU_Vehicle_Function_SelfInstallManager	This function is responsible for controlling, verifying and integrity check while installation, verification, activation, rollback of an installation file to respective Micros as per the Manifest.
F5	IVSU_Vehicle_Function_DownloadManager	This function is responsible for downloading binaries, control such as pause/resume of downloads, report Errors and reports Progress of downloads.
F6	IVSU_Vehicle_Function_TriggerManager	This function is responsible for categorize different type of IVSU related triggers and suggests error-handling mechanism for IVSU triggers
F7	IVSU_Vehicle_Function_StatusManager	Status Manager's responsibility is to periodically send logs to the IVSU cloud based on some configurable parameters, like every x days or when the log files gets to a certain size. Status Manager uses the Cloud Interface Manager to send the reports to the IVSU Cloud
F8	IVSU_Vehicle_Function_CAVC_Control	The function is responsible for starting the vehicle
F9	IVSU_Vehicle_Function_CAVC_Inhibit	The function that is responsible for inhibiting the start of the vehicle
F10	IVSU_Vehicle_Function_CAVC_Display	The function is responsible for displaying the correct messages to the customer within the required time.
F11	IVSU_Vehicle_Function_HMI	This function will define safer and reliable user experience with IVSU for OTA update.
F12	IVSU_Cloud_Function_StatusManager	This function is for monitoring, correcting, analyzing IVSU process from beginning to end. Thus, it is important to retrieve and dispatch status from vehicle side to corresponding micro service in cloud side.
F13	IVSU_Cloud_Function_SignedCommands	This function defines group of functions in IVSU feature which are Signed commands for OTA Program, Erase and DiffUpdate, Prepare, Activate, Rollback, Vehicle Inhibit and Vehicle De-inhibit

F14	IVSU_Cloud_Function_LoggingMonitoring	This function group specifies high-level requirements to log and monitor cloud operations.
F15	IVSU_Cloud_Function_Consumer_and_Service_Website	This function specification is to provide all requirements and flows for consumer and dealer website.
F16	IVSU_APP_Function_FMC_Brand_HMI	The function specific will cover all the Phone HMI flows, default values, and requirements for consumer FMC Brand App
F17	IVSU_Interface_Function_USB_Software_Updates	The purpose of this interface function is to provide requirements for all the vehicles that are capable to update thru USB port
F18	IVSU_Interface_Spec_Applink	The purpose of this Interface is to provide a middleware between IVSU Application and Applink Interface to avoid any code change dependency between IVSU Application and Applink Interface.
F19	IVSU_Interface_Spec_Connectivity	The purpose of this Interface is to provide a middleware between IVSU Application and Wireless Router Interface to avoid any code change dependency between IVSU Application and Wireless Router Interface.
F20	OVTP OTA FUNCTION Defination	This Function describes the functional use case of OTA for OVTP and is the controlling document for OTA Function IDs
F21	OTA Cloud Interface Specification	This Function describes interface requirements between Client device module and Ford back end OTA cloud infrastructure

Table 8: List of Functions

3.2 Signal List

Signal ID	Signal Name	Description
LS_ECG2ECU_00001	OVTP OTA session number	
LS_ECG2ECU_00002	SUCounter	Software Update Counter. For detailed description please refer to OVTP OTA FID spec
LS_ECG2ECU_00003	Number of blocks to erase	Parameters associated with Authorize Erase and Erase commands.
LS_ECG2ECU_00004	Start address of each block to be erased	Parameters associated with Authorize Erase and Erase commands
LS_ECG2ECU_00005	Size of each block to be erased	Parameters associated with Authorize Erase and Erase commands
LS_ECG2ECU_00006	Erase Memory Authorize command Signature	Parameters associated with Authorize Erase and Erase commands
LS_ECG2ECU_00007	Force Sync Counter Authorize command Signature	Parameters associated with Authorize Force Sync Counter command
LS_ECG2ECU_00008	Force Sync Counter after Successful Erase	

LS_ECG2ECU_00009	Number of blocks to program	Parameters associated with Authorize Program and Initiate Program commands
LS_ECG2ECU_00010	Start address of each block to be programmed	Parameters associated with Authorize Program and Program commands
LS_ECG2ECU_00011	Size of each block to be programmed	Parameters associated with Authorize Program and Program commands
LS_ECG2ECU_00012	Program memory Authorize command signature	Parameters associated with Authorize Program and Program commands
LS_ECG2ECU_00013	Block Sequence counter	
LS_ECG2ECU_00014	Force Sync Counter after Successful logic program (0x16)	

Signal ID	Signal Name	Description
LS_ECU2ECG_00001	ESN	ECU Serial number. (DID – New DID to be defined)
LS_ECU2ECG_00002	ECU Core Assembly number	ECU Core Assembly number (DID – 0xF111)
LS_ECU2ECG_00003	ECU delivery Assembly number	ECU delivery Assembly number (DID – 0xF113)
LS_ECU2ECG_00004	SWDL specification version	SWDL specification version (DID – 0xF162)
LS_ECU2ECG_00005	Diagnostic specification version	Diagnostic specification version (DID – 0xF163)
LS_ECU2ECG_00006	Vehicle Manufacturer ECU SW number	Vehicle Manufacturer ECU SW number (DID – 0xF188)
LS_ECU2ECG_00007	DID Configuration DID DE00 - DEFF	DID Configuration DID DE00-DEFF
LS_ECU2ECG_00008	DID Configuration DID DE01	DID Configuration DID DE01
LS_ECU2ECG_00009	DID Configuration DID DE02	DID Configuration DID DE02
LS_ECU2ECG_00010	DID Configuration DID DE03	DID Configuration DID DE03
LS_ECU2ECG_00011	DID Configuration DID DE04	DID Configuration DID DE04
LS_ECU2ECG_00012	DID Configuration DID DE05	DID Configuration DID DE05
LS_ECU2ECG_00013	DID Configuration DID DE06	DID Configuration DID DE06
LS_ECU2ECG_00014	DID Configuration DID DE07	DID Configuration DID DE07
LS_ECU2ECG_00015	DID Configuration DID DE08	DID Configuration DID DE08
LS_ECU2ECG_00016	DID Configuration DID DE09	DID Configuration DID DE09
LS_ECU2ECG_00017	DID Configuration DID DE0A	DID Configuration DID DE0A
LS_ECU2ECG_00018	DID Configuration DID DE0B	DID Configuration DID DE0B
LS_ECU2ECG_00019	DID Configuration DID DE0C	DID Configuration DID DE0C

LS_ECU2ECG_00020	DID Configuration DID DE0D	DID Configuration DID DE0D
LS_ECU2ECG_00021	DID Configuration DID DE0E	DID Configuration DID DE0E
LS_ECU2ECG_00022	DID Configuration DID DE1B	DID Configuration DID DE1B
LS_ECU2ECG_00023	ECU Cal-Config Part Number	ECU Cal-Config Part Number" didValue="F10A"
LS_ECU2ECG_00024	On-line Diagnostic Database Reference Number	On-line Diagnostic Database Reference Number" didValue="F110"
LS_ECU2ECG_00025	OTA session response	Positive Response/Negative response NRC(13/22/31)

LS_ECU2ECG_00026	Target ECU Internal OTA State	DID $D022 .
		• Byte 1:
		  ○ One byte Hex:
		  ▪ Last FID received
		• Byte 2:
		  ○ One byte SED (Internal OTA State)
		  ▪ Maybe 2 bytes if we really think we could have more than 256 states for future expansion
		• Byte 3 - 6
		  ○ 4 byte Hex: OTA Expected Address to Write and address to Erase
		  ▪ Note this would only be valid if Internal OTA State was a set of values indicating that a download is in progress, etc.
		  ▪ Otherwise, would simply be reported as all $00s
		Block sequence counter may be added in here as well???

LS_ECU2ECG_00027	Authorize Erase Memory response	Positive Response/Negative response NRC(11/13/15/16/17/31)
LS_ECU2ECG_00028	Erase Memory response	Positive Response/Negative response NRC(11/13/33/31)
LS_ECU2ECG_00029	Erase Successful Force Sync Counter response	Positive Response/Negative response NRC(11/13/17)
LS_ECU2ECG_00030	Authorize Program response	Positive Response/Negative response NRC(11/13/15/16/31/33)
LS_ECU2ECG_00031	maxNumberOfBlockLength	For Flash Write, Maximum block length accepted by Target ECU.
LS_ECU2ECG_00032	Initiate download response	Positive Response/Negative response NRC(11/13/15/16/17/31/33) For 33, Send Auth again.
LS_ECU2ECG_00033	Transfer data reponse	Positive Response/Negative response NRC(11/13/24/31/73) NRC Data bigger than requested block size(maxNumberOfBlockLength) – 0x31
LS_ECU2ECG_00034	Block Sequence counter echo	Positive response (Transfer data reponse).
LS_ECU2ECG_00035	Calculated Hash of all logical blocks root hashes (Swash)	
LS_ECU2ECG_00036	Complete data Transfer response	Positive and negative response(11/13/22/24)
LS_ECU2ECG_00037	Force Sync Counter after Successful logic program response	Positive and negative response

LS_ECU2ECG_00038		

4 Function Deployment

4.1 E/E Architecture Variant 1

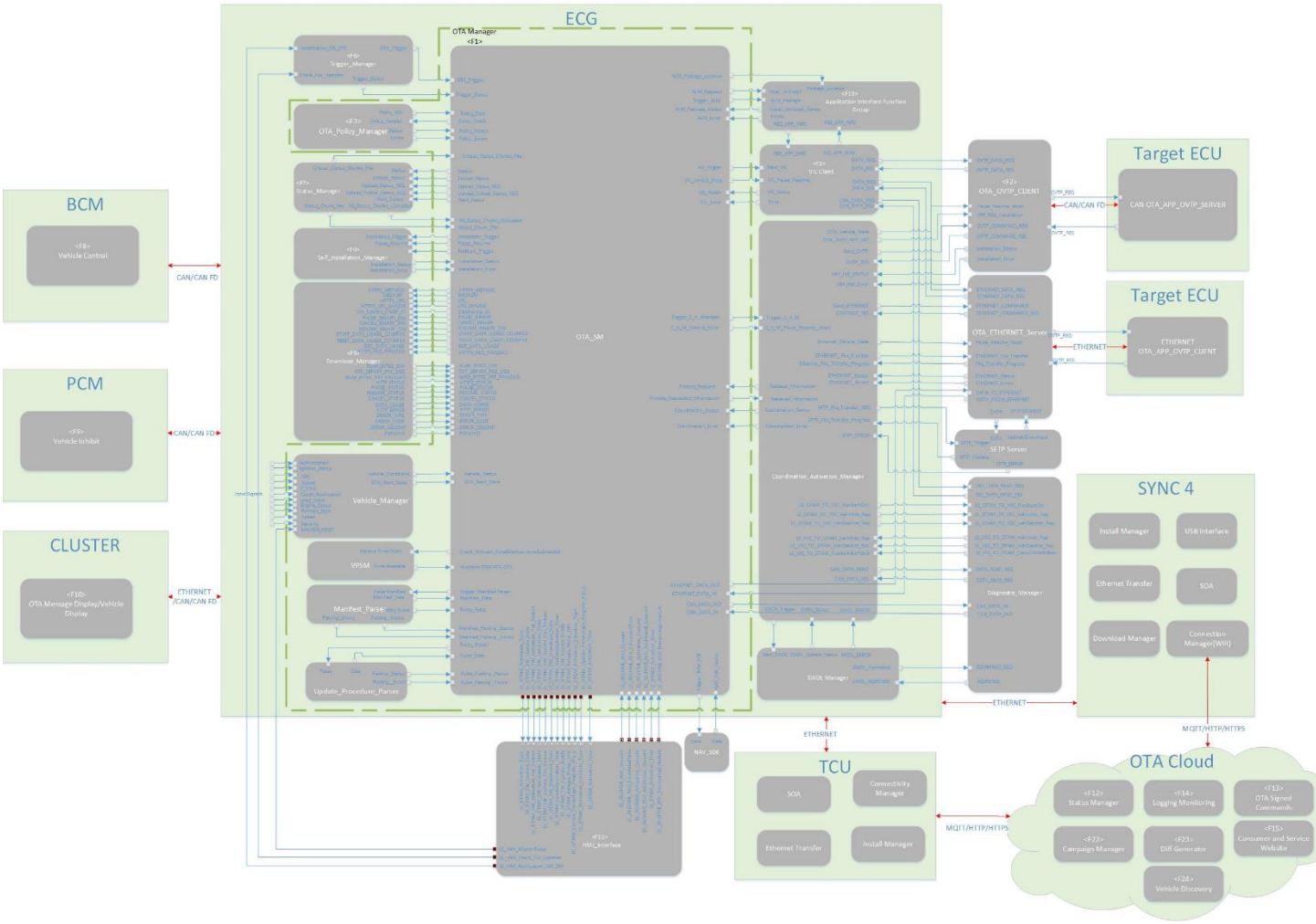


Figure 2: E/E Architecture, Variant 1

The first variant is the full connected vehicle where:

1. ECG is the OTA Master and a tester tool. It will download other ECUs files, will program modules in CAN or Ethernet and provide the Activation command to all ECUs in the vehicle. ECG is connected via Ethernet to TCU and SYNC therefore it will use any connectivity available to download software from the cloud.
2. TCU provides cellular connectivity and WiFi connectivity during ignition ON and OFF. The software files are first downloaded into the ECG then transferred to TCU.
3. SYNC4 provides WiFi connectivity during Ignition OFF that will be used for software file downloads requested from the ECG. The software files, are downloaded directly in the SYNC cache because if its size.
4. BCM is the module that will arbitrate all the requests to start the vehicle, including the OTA request
5. PCM is the module that will inhibit the vehicle when OTA requests it thru ECG
6. Cluster is the module that will display all time critical messages such as vehicle inhibit

4.1.1 E/E Components

All components that are OTA updatable shall utilize the CAN OVTP or Ethernet OVTP Server application. The exception is the SYNC4 and TCU module.

4.1.1.1 FRD-REQ-308756/C-###R_CMP_IVSU_V_00025### Capacitance Requirement Availability in case of Power OFF While OTA Update

For target ECUs which are powered at all times (or have the capability to latch power at key off), when the vehicle is shut down due to normal usage (e.g., customer keys off, remote start ends, etc.) the ECU must ensure that the OTA server component is correctly shut down and all information (e.g. DID $D022) to ensure the OTA client can resume the OTA transfer from the point where it was interrupted is written prior to module sleep or power down. The ECU is not explicitly required to ensure OTA resumption due to unexpected power removal (e.g., customer fuse pull or disconnection of the battery).

For target ECUs which are not powered at all times but, for example, rely upon power from the switched run/start bus, the target ECU shall have enough capacitance to ensure the OTA server component can correctly shutdown and accurately store all information (e.g. DID $D022) to ensure the OTA client can resume the OTA transfer from the point where it was interrupted. This is required even for unexpected removal of power. Exceptions to this are possible but require review and approval of the details of the design by the core IVSU OTA team.

4.1.2 E/E Connections

NA

4.1.3 Function Allocation

FunID	Function Name	Reference	VSEM ID	Allocated to (Element)
F1	OTA Manager	IVSU_Vehicle_Function_OTAManager	547911	ECG
F2	CAN OTA OVTP CLIENT	IVSU_Vehicle_Function_OTA_OVTP_CLIENT	547910	ECG
F3	VIL Manager	IVSU_Vehicle_Function_VILClient	547912	ECG
F4	Self-Install Manager	IVSU_Vehicle_Function_SelfInstallManager	547922	ECG, SYNC, TCU
F5	Download Manager	IVSU_Vehicle_Function_DownloadManager	547923	ECG, SYNC, TCU
F6	Trigger Manager	IVSU_Vehicle_Function_TriggerManager	547921	ECG
F7	Status Manager	IVSU_Vehicle_Function_StatusManager	548480	ECG
F8	Vehicle Control	IVSU_Vehicle_Function_CAVC_Control	546767	BCM
F9	Vehicle Inhibit	IVSU_Vehicle_Function_CAVC_Inhibit	546768	PCM
F10	Vehicle Display	IVSU_Vehicle_Function_CAVC_Display	527515	CLUSTER
F11	HMI Interface	IVSU_Vehicle_Function_HMI	548171	SYNC
F12	Cloud Status Manager	IVSU_Cloud_Function_StatusManager	547915	CLOUD
F13	Cloud Signed Commands	IVSU_Cloud_Function_SignedCommands	547916	CLOUD
F14	Cloud Logging Monitoring	IVSU_Cloud_Function_LoggingMonitoring	547917	CLOUD
F15	Cloud Consumer and Service Website	IVSU_Cloud_Function_Consumer_and_Service_Website	545839	CLOUD
F16	IVSU_APP_Function_FMC_Brand_HMI	IVSU_APP_Function_FMC_Brand_HMI	547920	Customer Mobile App

F17	IVSU_Interface_Function_USB_Software_Updates	IVSU_Interface_Funcation_USB_Software_Updates	547914	Module which has USB Interface and Ethernet interface with ECG
F18	IVSU_Interface_Spec_Applink	IVSU_Interface_Spec_Applink	547927	SYNC
F19	IVSU_Interface_Spec_Connectivity	IVSU_Interface_Spec_Connectivity	547925	ECG
F20	OVTP OTA FUNCTION Definition	OVTP OTA FUNCTION Definition	547919	Fast OTA ECUs
F21	OTA Cloud Interface Specification	OTA Cloud Interface Specification	546616	CLOUD, ECG
F22	Campaign Manager	IVSU_Cloud_Function_CampaignManager	583938	CLOUD
F23	Diff Generator	IVSU_Cloud_Function_Diff_Generator	584144	CLOUD
F24	Diff Updater	IVSU_Vehicle_Function_Diff Updater	583758	Any module that can be updated via OTA

Table 9: Function Allocation

Architectural Component/Interface	Overall Component ASIL	Req IDs	Req ASIL	Function/Behaviour	Req IDs	Req ASIL
Component 1	C(D)	Req a	B	Function 1	Req d	
		Req b	QM		Req e	B(C)
		Req c	C(C)	Function 3	Req f	C(D)
				Function 4	Req g	B(D)
Component 2	B(C)	Req b	QM	Function 1	Req d	
		Req h	B(C)			

Proposed Allocation Table

4.1.4 Signal / Parameter Mapping

ID	Logical Signal Name	Logical Signal Values	Mapped to Physical Signal Name	Physical Signal Values	Description
1	LS_OTAM_Update_Percentage_Progress_APP_x	Value {percentage}			Check for Update progress

2	LS_ASUHMI_ASU_ReoccuringSchedule	Value{ 01 - FALSE; 02 - TRUE}			Input to OTA Manager

3	LS_ASUHMI_ASU_CheckUpdate	Value{
		01 - True
		02 - False
		}
					One time consent

4	LS_OTAM_TriggerExpiration_Time	Values{
		01 - Not_expired
		02 - Expire
		}
					Software update expired clear all HMIs

5	LS_OTAM_UpdateReminder_Time	Values{
		01 - Bytes - date/time
		}
					SW Activation Reminder

6	LS_OTAM_UpdateExpiration_Time	Value { date/time }			Max time shown in the schdeule screen, if expire time is 3days from now then HMI shall only show 3days to activation the software bacause 4th day SW is not available.
7	LS_ASUHMI_Manage_Notification	Value{TRUE, FALSE}			

8	LS_ASUHMI_ASU_FeatureStatus	Values{
		01 - Enable
		02 - Disable
		}
					HMI Automatic software updates enable or disable OTA After Master reset or default values change

9	LS_ASUHMI_ASU_Consent	Value{TRUE, FALSE}			CCS settings True or False

10	LS_OTAM_SW_Update_Notify	Value {
		01 - PII_UPDATE;
		02 - Additional
		}
					HMI to display additional/pii consent

11	LS_OTAM_ECU_App_reside	Value {
		01 - APP_ECU_Updating
		}
					Customer check for update when App ECU is updating then HMI shall prompt the customer try later

12	LS_ASUHMI_ASU_Additional_Consent	Value{
		00 - NONE;
		01 - ONE_TIME;
		02 - PII_UPDATE;
		}
					OTA: One time skip additional but may need PII

13	LS_ASUHMI_ASU_ScheduleTime	Values{
		00 – Null
		01 – Bytes - date/time
		}
					Signal identify scheduled time/day for activation

14	LS_OTAM_Update_Time	Values{
		Bytes - date/time
		}
					OTA manager Last SW update time and date. Update HMI after every activation

15	LS_OTAHMI_Master_Reset_Status*	Value{
		00 - NONE
		01 - MasterReset
		02 - NoMasterReset
		}
					HMI shall notify OTA Manager for Master Reset

16	LS_ PARSERUSB_Conn_Status*	Values{
		01 - USB_Plug
		02 - USB_unPlug (download)
		}
					USB device status

17	LS_PARSER_USBSW_Update_Detected*	Values{
		01 - False
		02 - True
		}
					True: Processing Update…transient message

18	LS_PARSER_USBSW_Update_URL	Values {
		URLs/VIL Folder Location
		}
					if LS_PARSER_USBSW_Update_Detected = true, then Set IVSU trigger with content

19	LS_USBOTA_System_Updating*	Values{
		01 - Older_Software
		02 - Valid Manifest
		03 - Redownload Files
		04 - Sys_to_update_date
		}
					Determine if USB device is with vaild software

20	LS_USBOTA_SW_Update_Status*	Values{
		01 - Updating (Downloading/Installing/Resumed)
		02 - Failed
		03 - PENDING_Activation,
		04 - SUCCESSFUL,
		05 - Paused
		}
					If updating (download/install) failed then use "Failed" for USB Software update Status

21	LS_ASUHMI_Activation_Consent	Value{
		01 - NOW;
		02 - DATETIME;
		}
					One time schedule and NOW

22	LS_OTAM_Update_Percentage_OverallProgress	Value {percentage}			OTA/USB overall progress bar

23	LS_OTAM_OTAUSB_Number_of_Files	Value {
		01 - file remaining
		02 - total files
		}
					Total number of files in the manifest

24	LS_OTAM_Activation_Schedule_Type	Value {
		01 - WEEKLY;
		02 - DAY;
		}
					Schedule weekly or daily share with OTA Manager

25	LS_OTAM_SW_Activation_Fail_Reason	Values{
		00_ NONE
		01 - SW CORRUPTED;
		02 - PERMANENT_INHIBIT;
		03 - USB_FAILURE;
		04 - WARNING;
		05 - PARTIAL
		}
					IF USB software activation failed then Use "USB_Failure"

26	LS_OTAM_SW_Update_Fail_Reason	{ErrorCode; }			USB Software update failed reason
27	LS_OTAM_Release_Notes_Info	Value {text}			Release Notes

28	LS_OTAM_Activation_TypeSW_AB_ER	Value{
		01- AB
		02- ER
		03 - AB and ER
		}
					OTA Manage sharing type of software update

29	LS_OTAM_Activation_Type	Value{
		01- NOIGNITIONCYCLE
		02- IGNITIONCYCLE
		03- INHIBIT
		}
					Activation Type

30	LS_OTAM_Vehicle_Inhibit_Type	Value{
		00 - NONE
		01 - ProgrammingSession
		02 - ActivatingNOW
		}
					Vehicle in Programming Mode or activating the software
					HMI Logic shall make decision
					if LS_OTAM_Activation_TypeSW_AB_ER = AB-ER then show LS_OTAM_Vehicle_Inhibit_Type = ProgrammingSession
					
					if LS_OTAM_Activation_TypeSW_AB_ER = AB then show LS_OTAM_Vehicle_Inhibit_Type = ActivatingNow
					
					if LS_OTAM_Activation_TypeSW_AB_ER = ER then show LS_OTAM_Vehicle_Inhibit_Type = ProgrammingSession

31	LS_OTAM_Activation_Time	Domain: 2 bytes (In seconds).			Activation for both E/R and/or AB Time range (2min to 30mins)

32	LS_OTAM_HMI_OTAUSB_Clear	Values{
		01 - Pending
		02 - ConfigtimeExpire
		03 - ClearHMIs
		}
					USB update is paused and OTA Manager shall clear cache after 7days

33	LS_OTAM_SW_Instalation_State	Values{
		01 - IN_PROGRESS
		02 - PENDING,
		03 - FAILED,
		04 - PAUSED,
		05 - SUCCESSFUL
		}
					HMI shows if Check for update was requested

34	LS_OTAM_SW_Download_State	Values{
		01 - IN_PROGRESS,
		02 - PENDING,
		03 - PAUSED,
		04 - FAILED,
		05 - SUCCESSFUL
		}
					HMI shows if Check for update was requested

35	LS_OTAM_SW_Update_Postpone	Values{True or false}			

36	LS_OTAM_SW_Update_State	Values{
		00 - Clear_HMI
		01 - IN_PROGRESS
		02 - PENDING
		03 - FAILED,
		04 - SUCCESSFUL;
		05 - UP_TO_DATE;
		}
					OTA Software update Status

37	LS_OTAM_No_ProgSession_Preconditions_Supported	Values{ 01 - Vehicle Speed Too High 02 - Voltage Out of Range 03 - Charging in Progress 04 - PRNDL Out of Range 05 - Hazards On 06 - After Run Active 07 - ESCL Lock Pending 08 - Alarm Actively Sounding 09 - Steering Pinsion Torque Out of 10 - Range 11 - Diagnostic Self-Test Active 12 - Engine RPM Too High (or 02 - Torque Available) 13 - Charging Fault 14 - Ignition Status Out of Range 15 - Liftgate Ajar 16 - Park Lamps On 17 - Limp Home Active 18 - Illuminated Exit Active 19 - Door Ajar 20 - Hot Reclamp Active 21 - Brake Pedal Pressed 22 - Park Brake Out of Range or Activation in Progress			If software activation is postponed then set a flag for HMI and next action

38	LS_OTAM_HMI_Master_Reset	Values{
		01 - Cancel
		02 - Pending,
		03 - Pause,
		}
				1. ASU = OFF Cancel the pre-download for only one-time
				2. ASU = ON Pending for consent, with additional consent
				3. ASU = ON Pause during master reset and resume after it's complete without additional consent	

39	LS_OTAM_Activation_Status	Values{
		01 - Expired
		02 - Pending,
		03 - Pause
		}
				Software Activation status	

40	OVTP_REQ				
41	OVTP_RES				

42	LS_OTAM_DisplayMsg_Type		VehStrtInhbt_D_Dsply: 200ms FP	Periodicity: 200ms FP Domain: 4 bits 0b0000 – NoMessage 0b0001 – DuringOtaActivate 0b0010 – PostOtaActivate Warning 0b0011 – PostOtaActivate PermFail 0x4 to 0xF – Reserved. Description: ECG sets value to display different messages in IPC ASIL: QM Cloud signed: NO	

43	LS_OTAM_DisplayMsgInfo_Time		VehStrtInhbt_T_Dsply: 100ms EP	Domain: 2 bytes Time16bit_ET in seconds. Description: ECG sets value to display time information Vehicle Inhibited display message. ASIL: QM Cloud signed: NO	
44	LS_OTAM_TO_VSC_VehInhbt_Req		CloudVehCtlData_Tp_Rq - Event Only TP	Domain: 269 bytes Byte1: 0x01 (Vehicle Inhibit) Byte 2-269 bytes: FESN, Cccounter, Signature Description: Authorize to Vehicle Inhibit. ASIL: B (meets E2E req) Cloud signed: YES	

45	LS_OTAM_TO_VSC_RunStartCtrl		CloudVehCtlData_Tp_Rq - Event Only TP	Domain: 269 bytes Byte1: 0x00 (Vehicle De-Inhibit) Byte 2-269 bytes: FESN, Cccounter, Signature Description: Authorize to Vehicle De-Inhibit. ASIL: B (meets E2E req) Cloud signed: YES	
46	LS_VIC_TO_VSC_ISPR_Fdbk		VehOn_D_RqCld	Periodicity: 200ms FP Domain: 2 bits 0b00 – Null, 0b01 – Off, 0b10 – On, 0b11 – Not used ASIL: B Cloud signed: NO	

47	LS_VIC_TO_VSC_ISPR_Fdbk		OtaActv_D_Stat	Periodicity: 200ms FP Domain: 4 bits (State encoded) 0x00 – NoInVehicleOta 0x01– Interruptible_AB 0x02 – NonInterruptible_AB 0x03 – NonInterruptible_ER 0x04 – NonInterruptible Config 0x05 – NonInterruptible_Pending 0x06 - NonInterruptible_KeyDist 0x07 to 0x0F – NotUsed Description: ECG sets appropriate state value based on OTA Manager state machine. ASIL: B Cloud signed: NO	

48	LS_VIC_TO_VSC_ISPR_Fdbk		VehOnRqstr_D_Stat	Periodicity: 200ms FP Domain: 4 bits (State encoded) 0x00 – NoRequestor 0x01– OverTheAir 0x02 – StolenVehInhbt 0x03 - FleetVehInhbt 0x04 to 0x0F – NotUsed Description: ECG sets appropriate state value based on Feature requesting for RunStart Bus Control ASIL: B Cloud signed: NO Dependability Signals: VehOnDRqCld_No_Crc VehOnDRqCld_No_Cnt	

49	LS_VSC_TO_OTAM_RunStartCtrl_Status		VehOnSrc2_D_Stat	Domain: 4 bits 0x00 – Off 0x01 – Manual 0x02 – RemoteStart 0x03 – RemoteParkAssist 0x04 – OverTheAir 0x05 – 0xF – NotUsed Description: BCM sets state based on feature for which RunStart Bus control was offered and being offered. ASIL: B Cloud signed: NO Dependability Signals: CrnkInhbt2_No_Cnt CrnkInhbt_No_Crc	

50	LS_VSC_TO_OTAM_RunStartCtrl_Status		VehOnCtl_D_Stat	Domain: 2 bits 0x00 – NULL 0x01 – 0ff 0x02 – On 0x03 – Not Used Description: BCM broadcasts commanded state of the Run/Start bus so ECG can verify it is being requested on. ASIL: B Cloud signed: NO Dependability Signals: CrnkInhbt2_No_Cnt CrnkInhbt_No_Crc	

51	LS_OTAM_TO_VSC_VehInhbt_Req		CloudVehCtlData_Tp_Rq -- Event Only TP	Domain: 270 bytes Byte1: 0x01 (Vehicle Inhibit) Byte 2-269 bytes: FESN, Cccounter, Signature Description: Authorize to Vehicle Inhibit. ASIL: B (meets E2E req) Cloud signed: YES	

52	LS_OTAM_TO_VSC_VehDeInhbt_Req		CloudVehCtlData_Tp_Rq -- Event Only TP	Domain: 270 bytes Byte1: 0x00 (Vehicle De-Inhibit) Byte 2-269 bytes: FESN, Cccounter, Signature Description: Authorize to Vehicle De-Inhibit. ASIL: B (meets E2E req) Cloud signed: YES	
53	LS_VSC_TO_OTAM_VehInhbt_Res		CloudVehCtlData_Tp_Res - Event Only TP	Postive response (3 bytes): 0x81,echo CP & Requestor Negative reponse (2 bytes): 0x7F, NRC ASIL: QM Cloud signed: NO	

54	LS_VSC_TO_OTAM_VehInhbt_Res		VehStrtInhbt_D_Stat – 200ms FP	Domain: 1 bit (0b0 – No Inhibit,0b1 – Inhibit) Description: Value set to 0b1- Inhibit – Vehicle Inhibited due to CAVC ASIL: B Cloud signed: NO Counter:VehOnDRqCld_No_Cnt CRC: VehOnDRqCld_No_Crc	
55	LS_VSC_TO_OTAM_VehDeInhbt_Res		CloudVehCtlData_Tp_Res - Event Only TP	Postive response (3 bytes): 0x80,echo CP & Requestor Negative reponse(2 bytes): 0x7F, NRC ASIL: QM Cloud signed: NO	

56	LS_VSC_TO_OTAM_VehDeInhbt_Res		VehStrtInhbt_D_Stat – 200ms FP	Domain: 1 bit (0b0 – No Inhibit,0b1 – Inhibit) Description: Value set to 0b0 - No Inhibit – “Vehicle NOT Inhibited due to CAVC” ASIL: B Cloud signed: NO Counter:VehOnDRqCld_No_Cnt CRC: VehOnDRqCld_No_Crc	

57	LS_VSC_TO_VIC_CrankInhibit		CrnkInhbt_B_Stat -- 60ms FP	Domain: 1 bit 0b0 – No Inhibit 0b1 – Inhibit Description: No Inhibit – No Crank Inhibit due to ESCL OR OTA. Inhibit – Crank Inhibit due to ESCL OR OTA. ASIL: B Cloud signed: NO	
58	LS_VIC_TO_VSC_ISPR_Fdbk		PtIgnSwtch_D_Stat -- 100ms EP	Domain: 2 bits Off, On, No data exists and Faulty ASIL: B Cloud signed: NO Counter: PtIgnSwtch_No_Cnt CRC/CS: PtIgnSwtch_No_Cs	

59	LS_OTAM_TO_VSC_GlblClk_Actl		1000ms EP GlblClkYr_No_Actl GlblClkHr_No_Actl GlblClkDay_No_Actl GlblClkMnte_No_Actl GlblClkScnd_No_Actl	Description: BCM broadcasts global clock	
60	LS_OTAM_TO_VSC_InhbtGlblClk_Req		1000ms EP InhbtGlblClkYr_No_Rq InhbtGlblClkHr_No_Rq InhbtGlblClkDay_No_Rq InhbtGlblClkMnte_No_Rq InhbtGlblClkScnd_No_Rq InhbtGlblClkScnd_B_Rq	Description: ECG requests to set global clock	
61	LS_ECG_TO_BCM_OnDemandRequest		BattULoChrg_D_RqOta	0x0 No request (don't initiate on account OTA request - default) 0x1 Request for charging	

62	LS_BCM_TO_ECG_Energy_Transfer_Request		BattULoChrgHyb_B_Rq	0x0 No_Request 0x1 Request_Energy_Transfer	
63	LS_HPCM_TO_ECG_Energy_Transfer_Status		ULoBattTransfer_D_Stat	0x0 No_Transfer 0x1 Transfer_in_Progress 0x2 Insufficient_Energy_To_Transfer 0x3 Transfer_Through_Grid_Energy 0x4 Transfer_Complete 0x5 Transfer_Error	
64	LS_OTAM_APP_Update_DOWN	Value{ 01 – App_Name}			APP name, such as Navigation
65	LS_OTAM_APP_UPDATE_Timer	Value{xx Minutes}			APP shut down time
66	LS_ASUHMI_APP_DOWN	Value {0- False 1-True}			APP Shut down Consent

67	LS_SVS_OTAM_Active	Value{ True or false}			Stolen vehicle service
68	LS_OTAM_Precondition_unknown_Error	Value { True False }			“Unknown reason” when it is not in $ D04F

69	LS_OTAM_HMI_Remote_Consent	Value{
		00 - ASU Setting On/OFF
		01 – One-time Consent
		02 – Additional Consent
		03 – PII Consent
		}
					OTAM shall set this flag when it receives remote consent changes and notify HMI.

70	LS_OTAM_HMI_Remote_Notification	Value{ 00 – Default 01 – Notification On/Off}			OTAM shall set this flag when it receives remote notification settings change and notify HMI.

71	LS_OTAM_HMI_Schedule_Changes	Value{
		00 – Clear the schedule
		01 – Update day/time
		}
					OTAM shall set this flag when it receives remote schedule and notify HMI.

72	LS_OTAM_HMI_ClearHMIsPrompts	Value { 00 – Service tool 01 – OTA cancel trigger 02 – Clear failures ICONs by service tool }			Update is cancelled
73	LS_OTAM_ConenctionType_WiFi	Value{ 00 - True 01 - False}			

74	LS_OTAM_SW_Download_Pause_Reason	Values{
		01 – USB Update,
		02 – Master Reset,
		03 - eCall,
		04 – Loadshed,
		05 – no connectivity
		06 – unknown,
		07 – none
		}
					If LS_OTAM_SW_Download_State = pause, send reason

75	LS_OTAM_download_progress_percentage	Value {percentage}			Software is downloading over Wi-Fi

Table 10: Signal / Parameter Mapping

5 Feature Implementation Modeling

All interaction/sequence diagrams in this section are for illustration purpose only. They are not requirements. Purpose of these diagrams are meant to be used as example.

5.1 Component Interaction Diagrams

Scenario: “ECG updating TCU via USB”

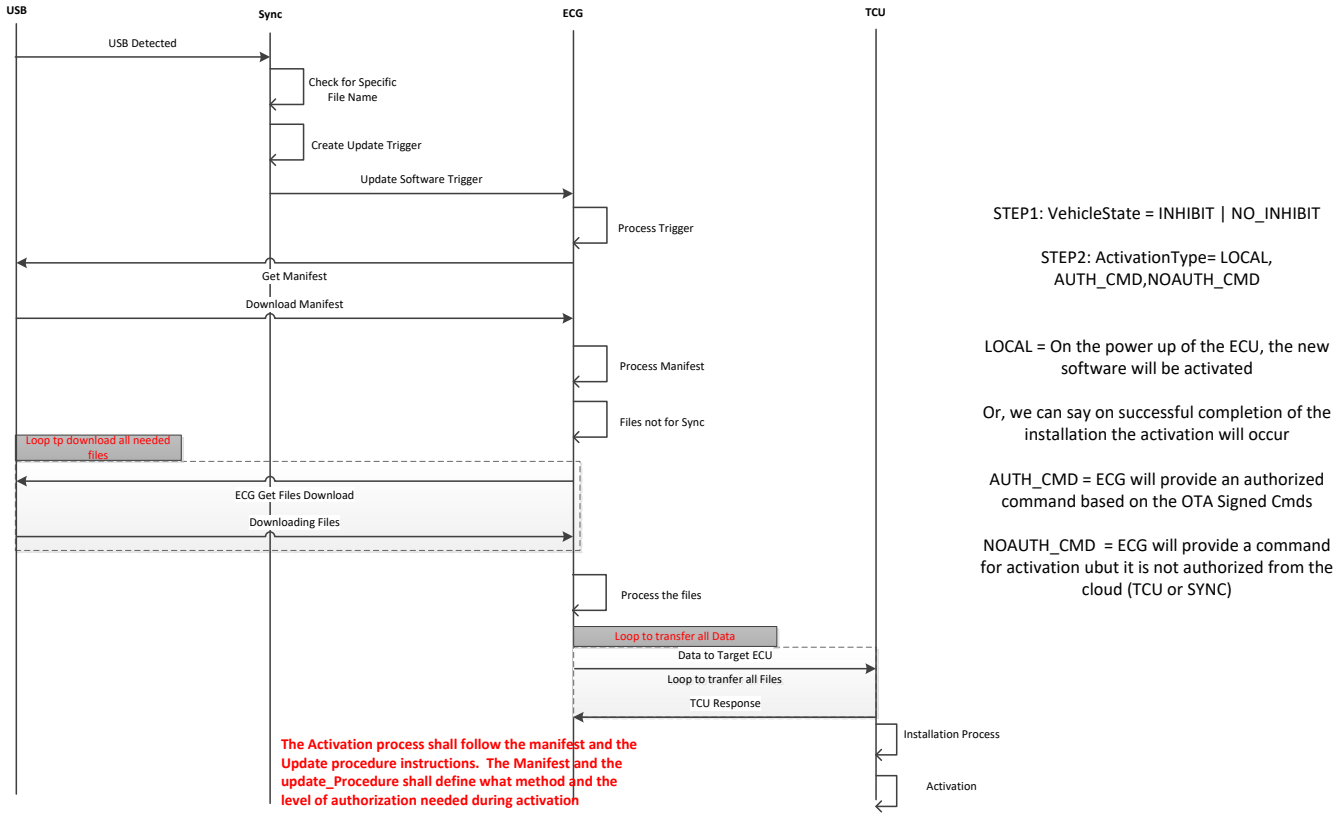
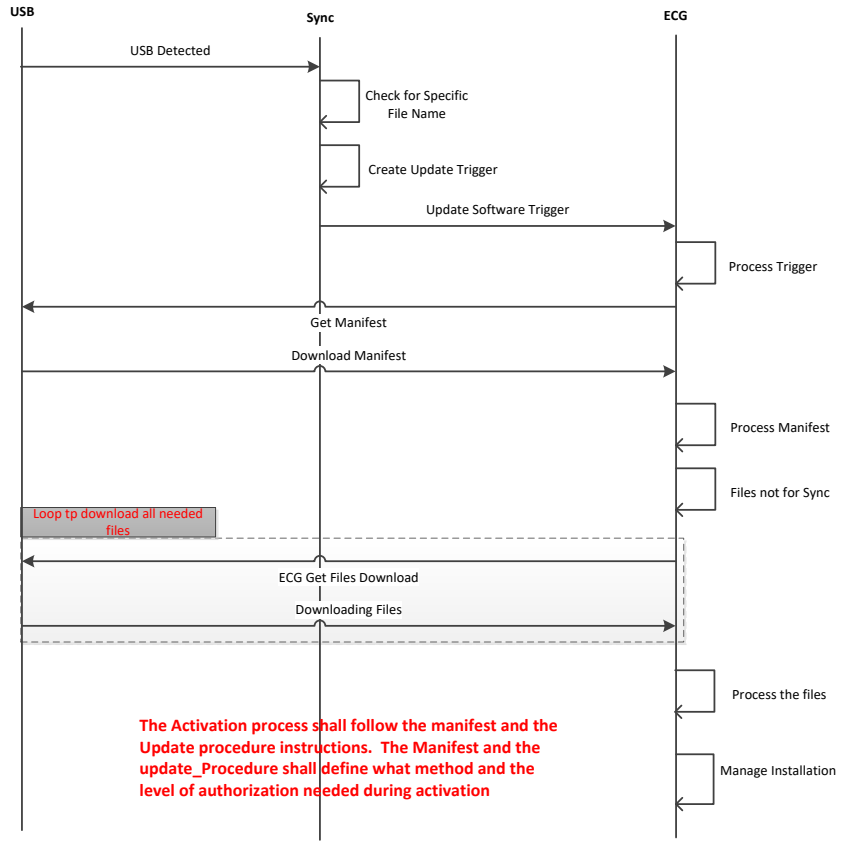


Figure 3: Flowchart of ECG Updating TCU via USB

5.1.1 Scenario: “ECG updating itself via USB”



STEP1: VehicleState = INHIBIT | NO_INHIBIT

STEP2: ActivationType= LOCAL, AUTH_CMD,NOAUTH_CMD

LOCAL = On the power up of the ECU, the new software will be activated

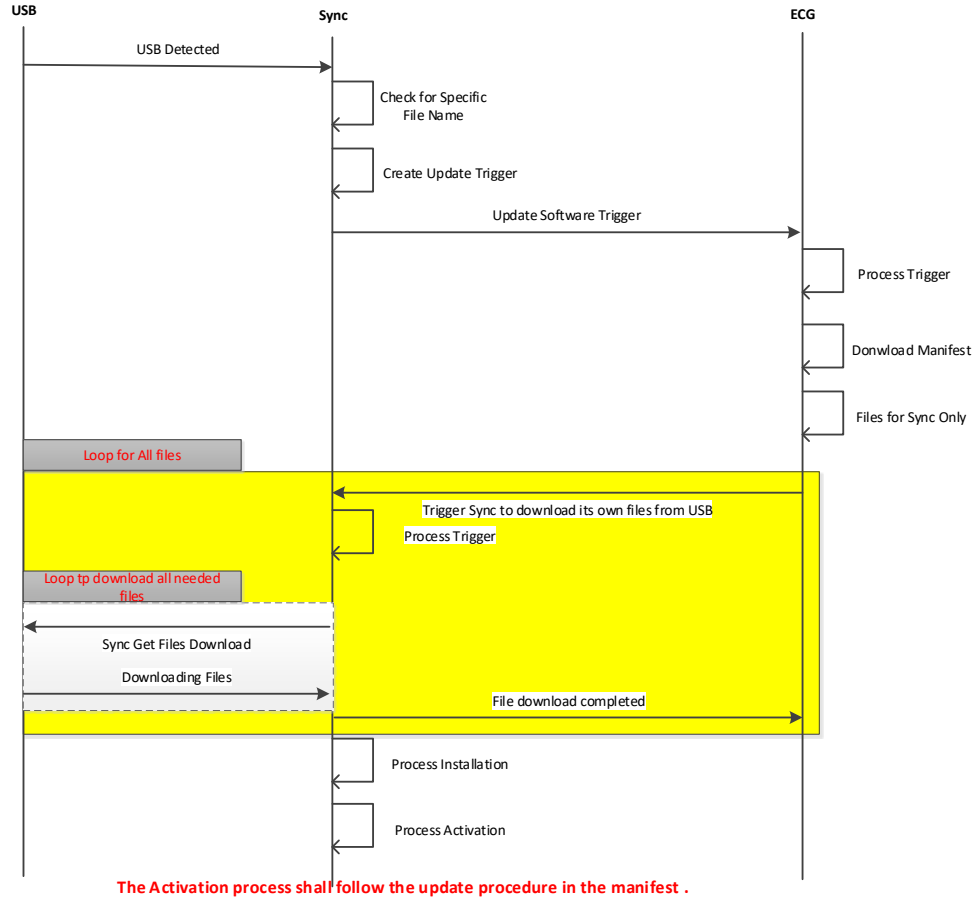
Or, we can say on successful completion of the installation the activation will occur

AUTH_CMD = ECG will provide an authorized command based on the OTA Signed Cmds

NOAUTH_CMD = ECG will provide a command for activation ubut it is not authorized from the cloud (TCU or SYNC)

Figure 4: Flowchart of ECG Updating itself via USB

5.1.2 Scenario: “Sync updating itself via USB”



STEP1: VehicleState = INHIBIT | NO_INHIBIT

STEP2: ActivationType= LOCAL, AUTH_CMD,NOAUTH_CMD

LOCAL = On the power up of the ECU, the new software will be activated

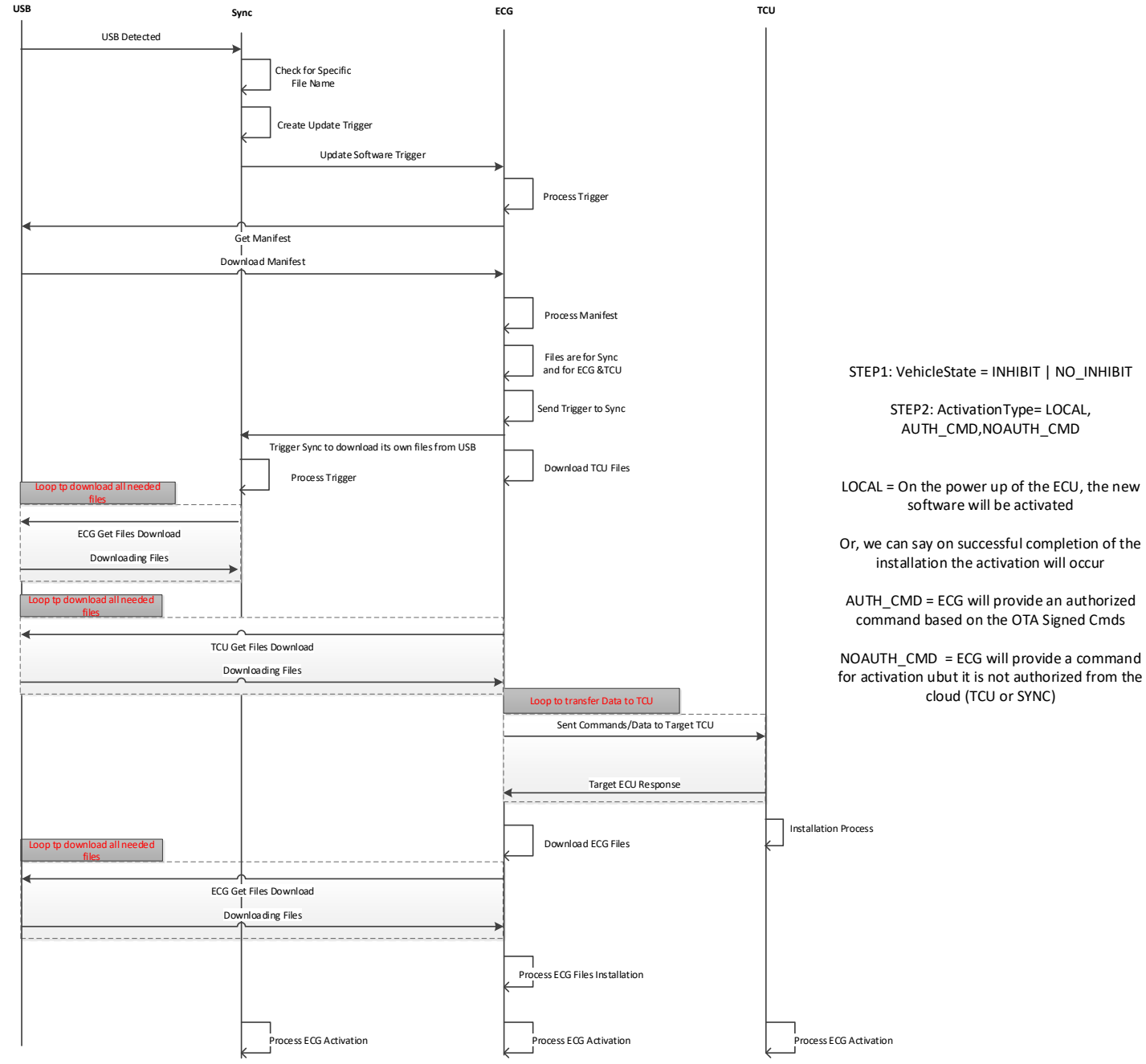
Or, we can say on successful completion of the installation the activation will occur

AUTH_CMD = ECG will provide an authorized command based on the OTA Signed Cmds

NOAUTH_CMD = ECG will provide a command for activation ubut it is not authorized from the cloud (TCU or SYNC)

Figure 5: Flowchart of Sync Updating itself via USB

5.1.3 Scenario: "Update of TCU, Sync and ECG via USB"



The Activation process shall follow the manifestns. The activation for all the three ECUs could happen during the next ignition cycle.

Figure 6: Flowchart of Sync, TCU and ECG update via USB

5.1.4 Scenario: “Update of Sync via OTA”

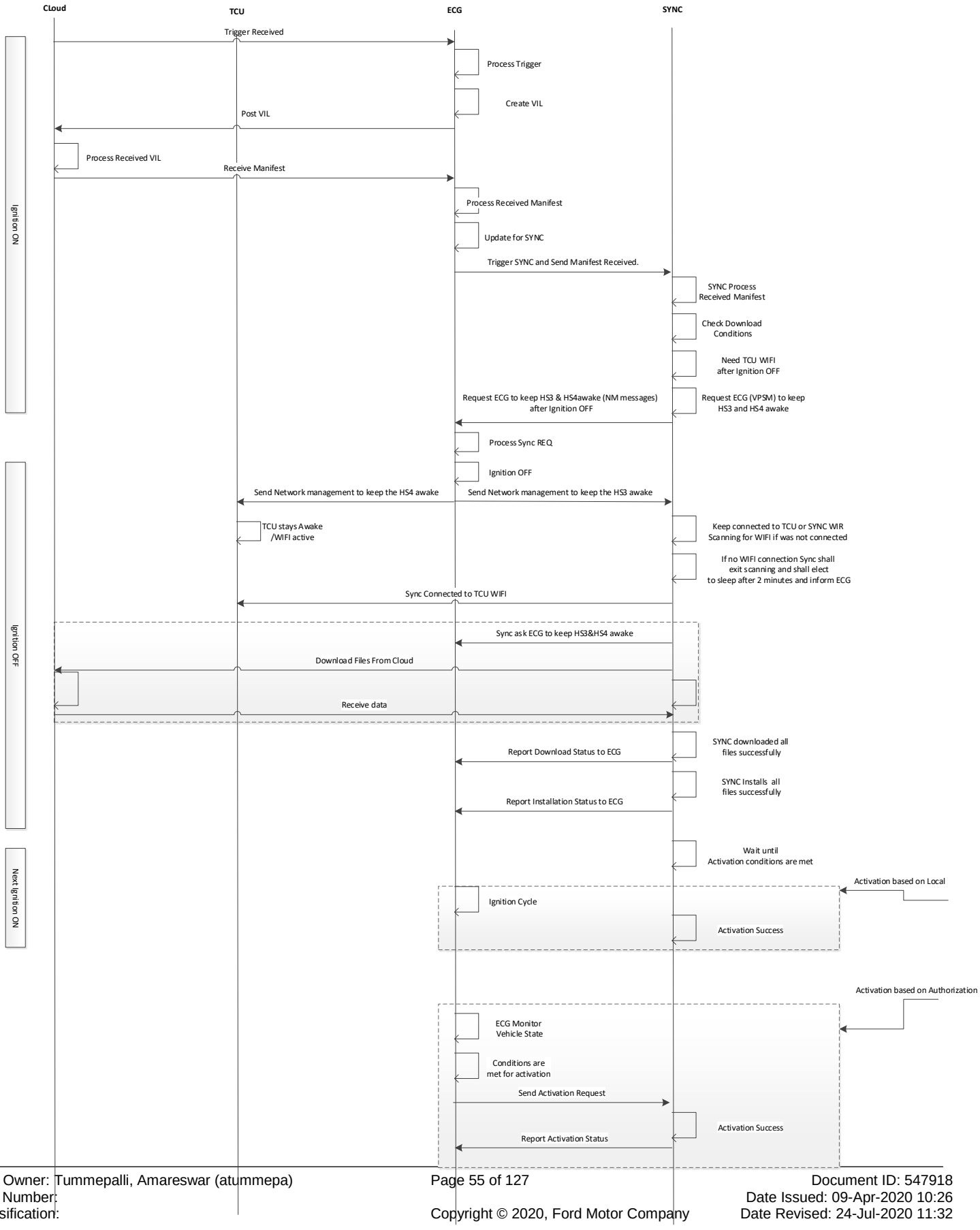


Figure 7: Flowchart of Sync Updating itself via OTA

5.1.5 Scenario: “Update Sync via TCU On Ignition On Engine Running”

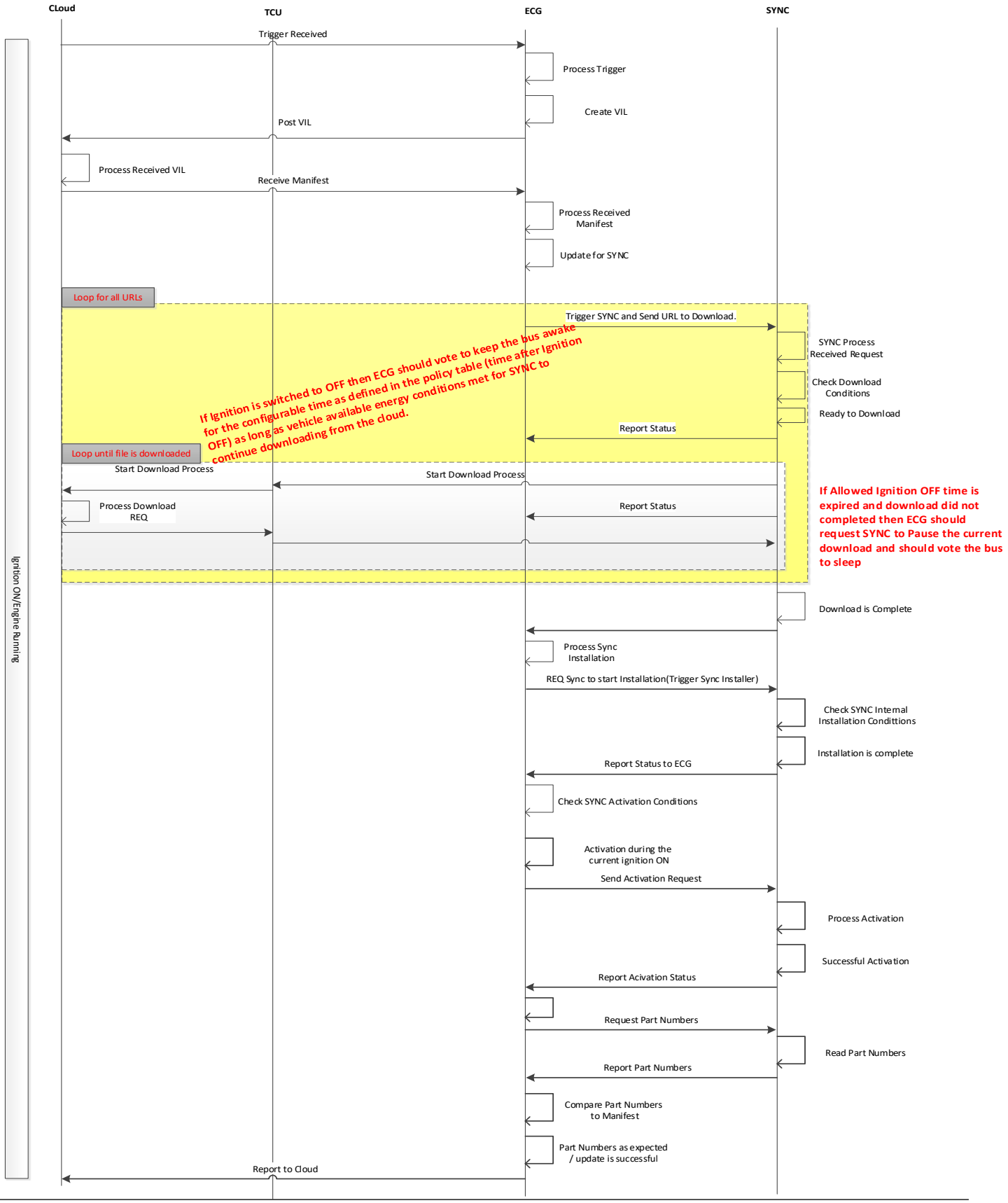


Figure 8: Update Sync Via TCU On Ignition On Engine Running

5.1.6 Scenario: “Update SYNC via External WIFI On Key Off”

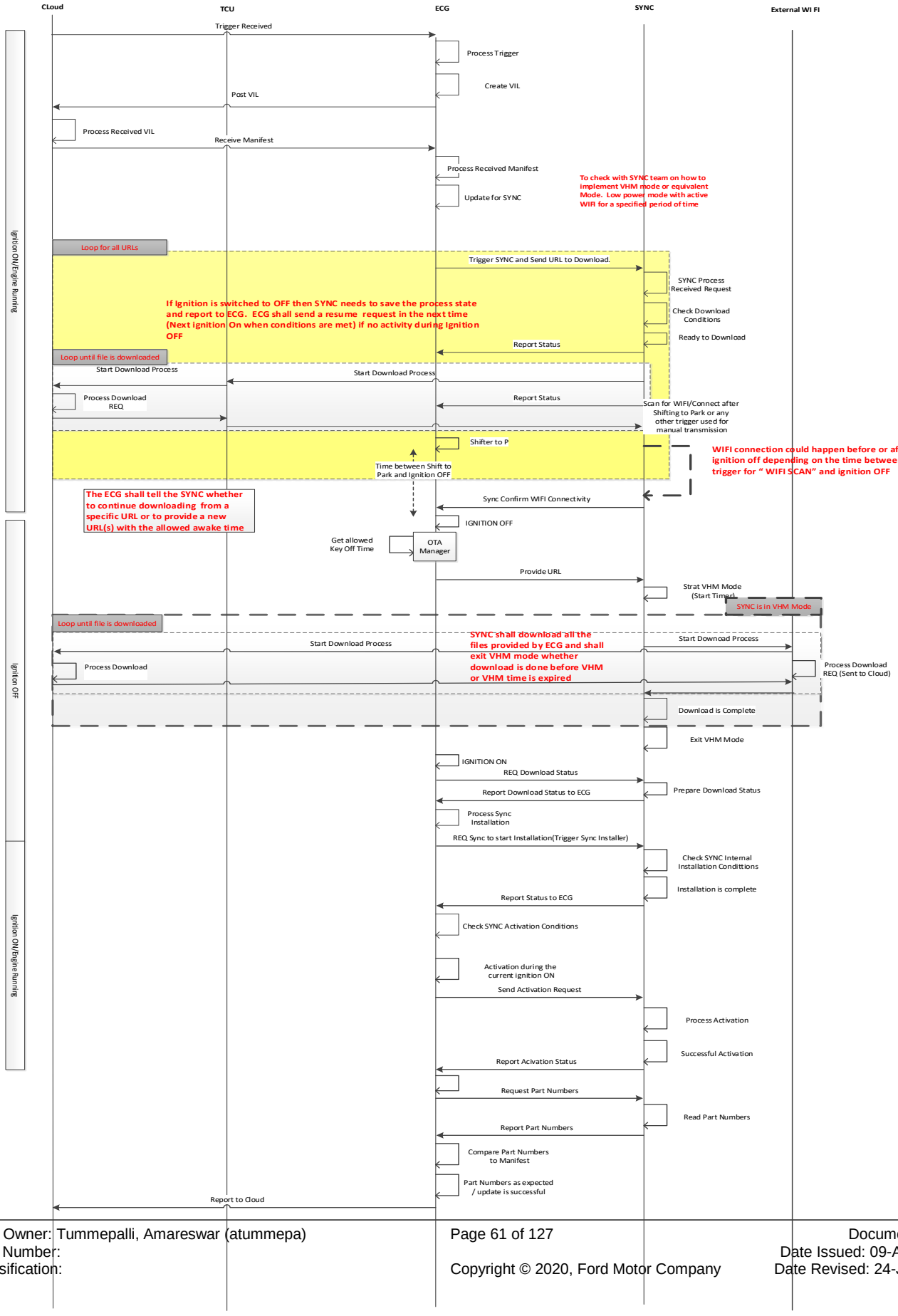


Figure 9: Update SYNC via External WIFI on Key Off

5.1.7 Scenario: “Update SYNC via TCU on Key Off”

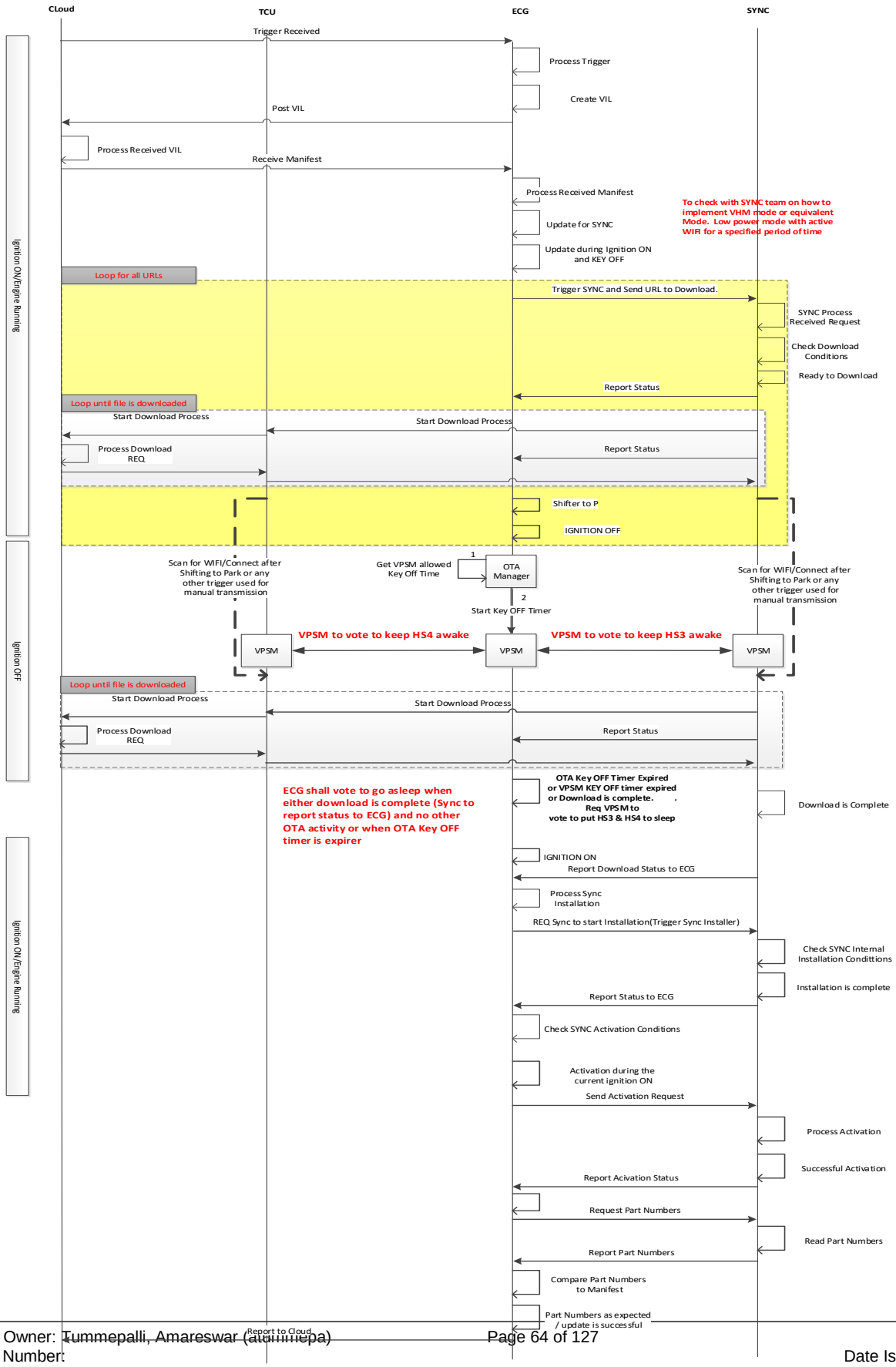


Figure 11: Update SYNC via TCU on Key Off

5.1.8 Scenario: “Update ECG via TCU on Key Off”

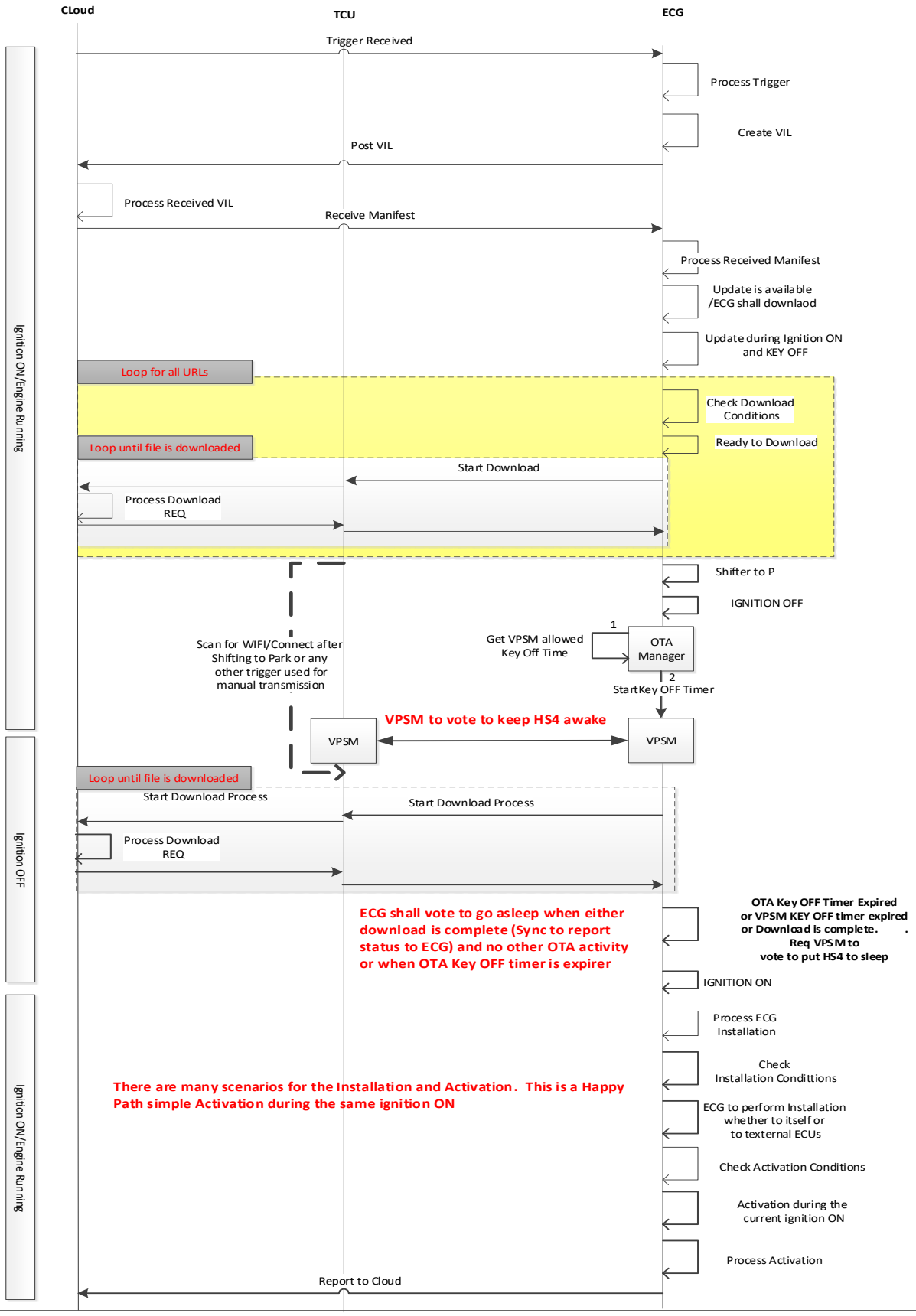


Figure 12: Update ECG via TCU on Key Off

5.1.9 Scenario: “Update ECG via SYNC on Key Off”

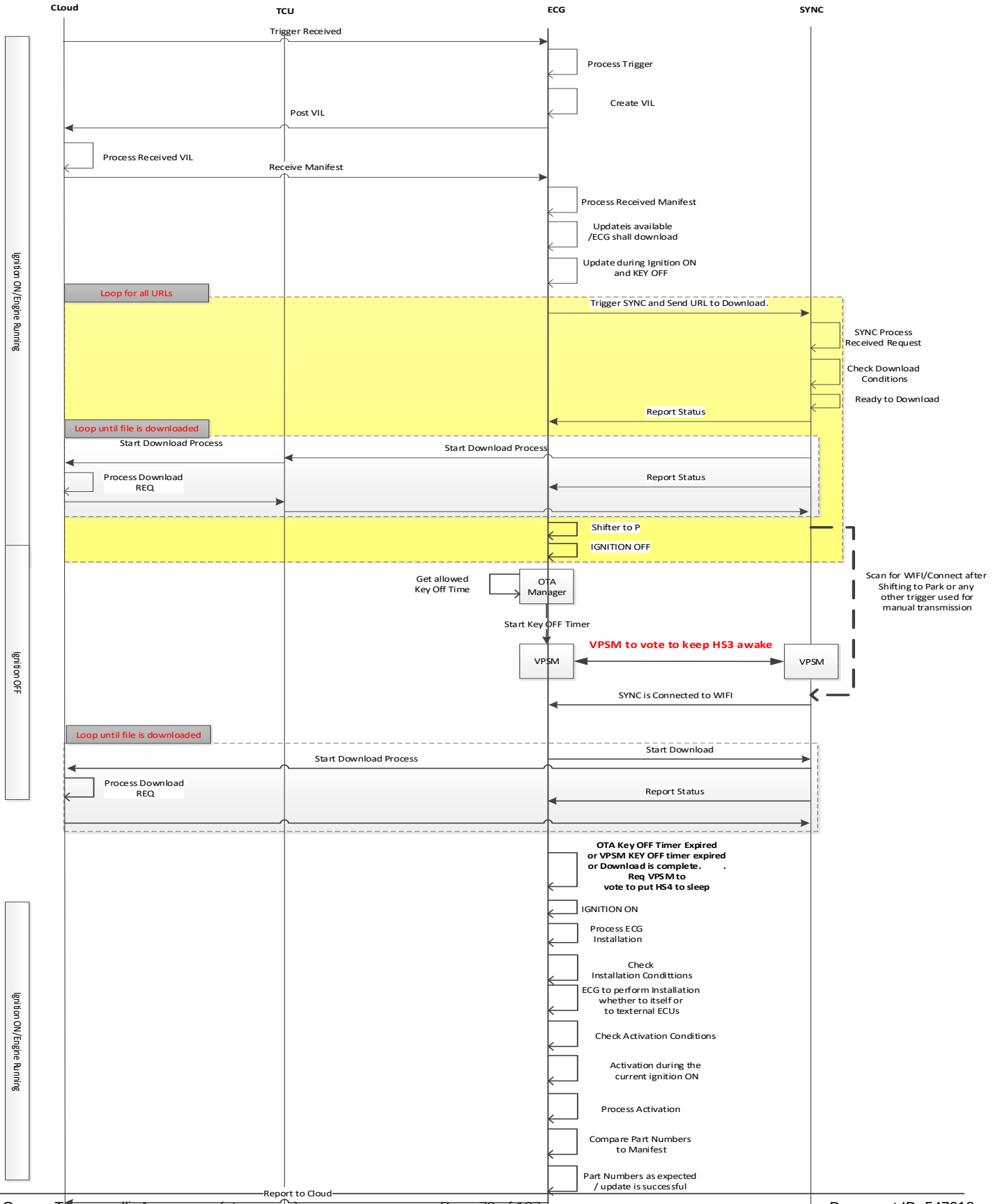


Figure 13: Update ECG via SYNC on Key Off

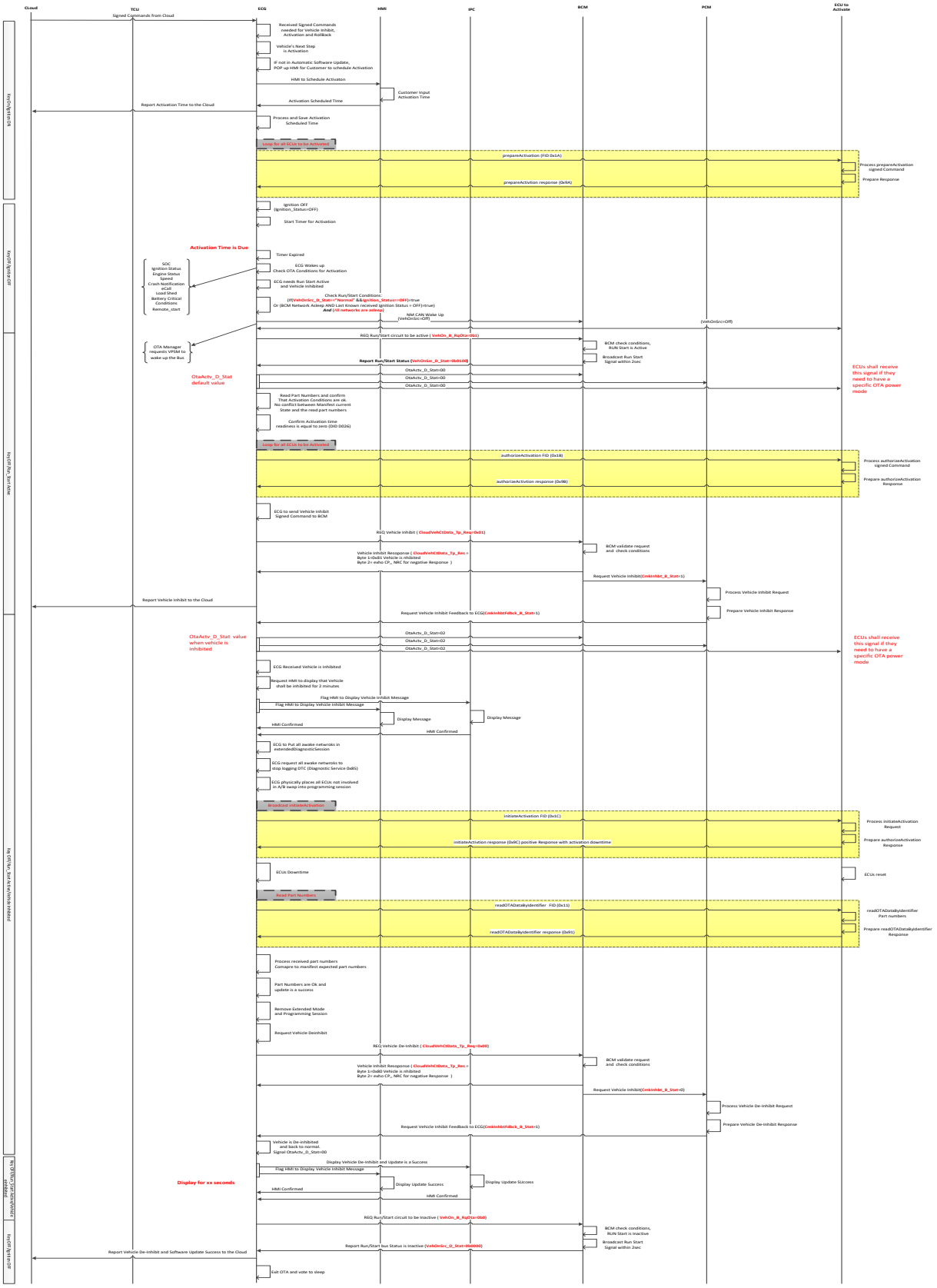


Figure 14: OVTP A/B Activation Flowchart (ECG is not part of the update)

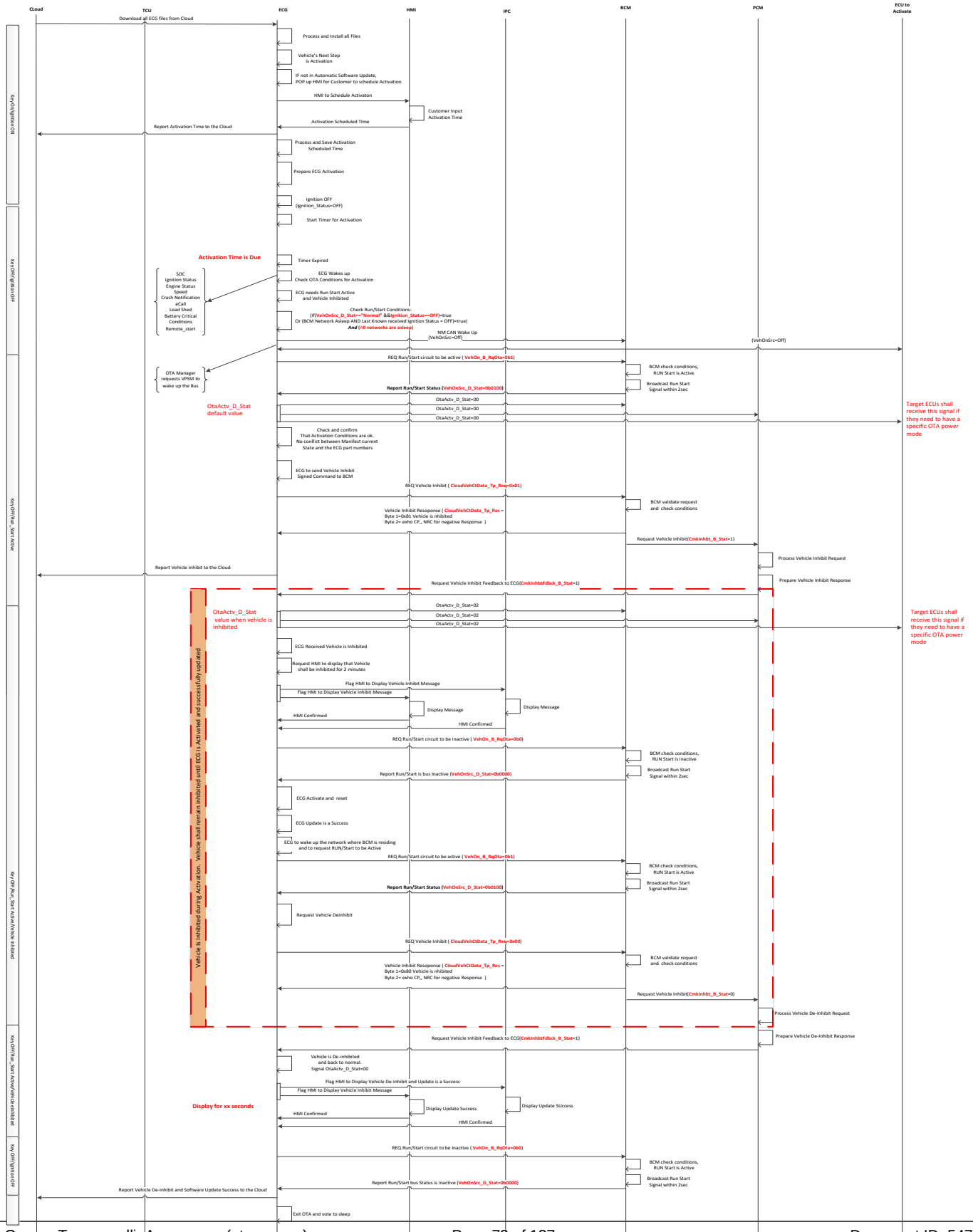


Figure 15: ECG Activation Flowchart

5.1.10 Scenario "On Demand Charging" Request

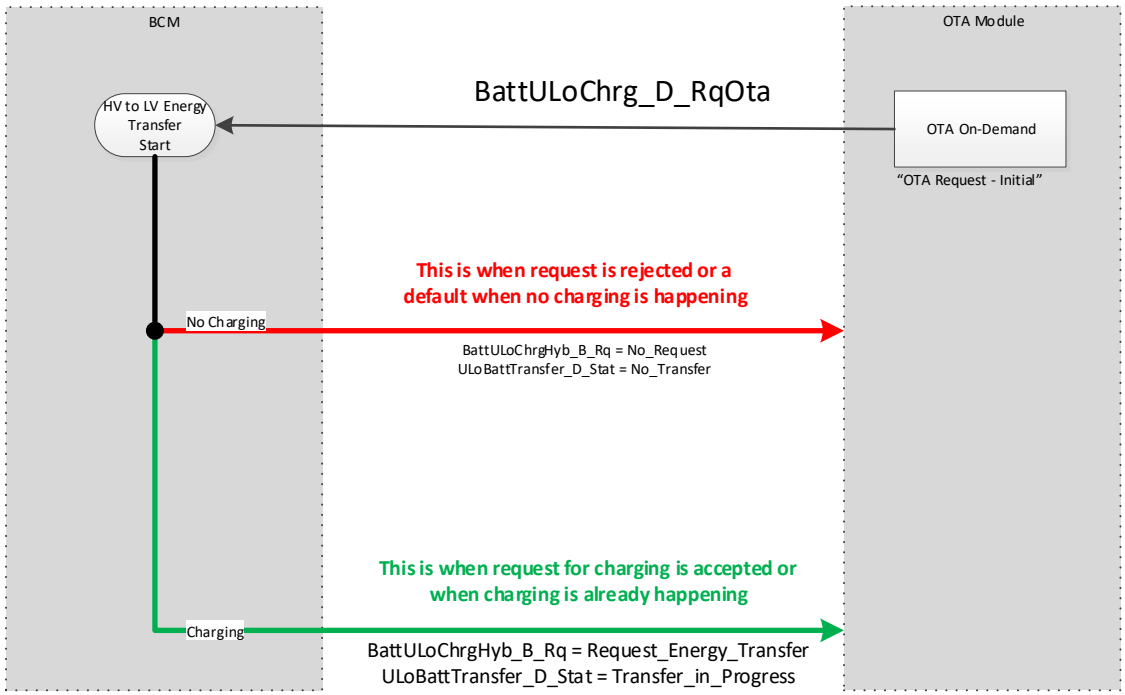


Figure 16: "On Demand Charging" Request

5.1.11 Scenario: "Update Target ECU with one Micro Via OVTP"

5.1.11.1 Read OTA Data by Identifier

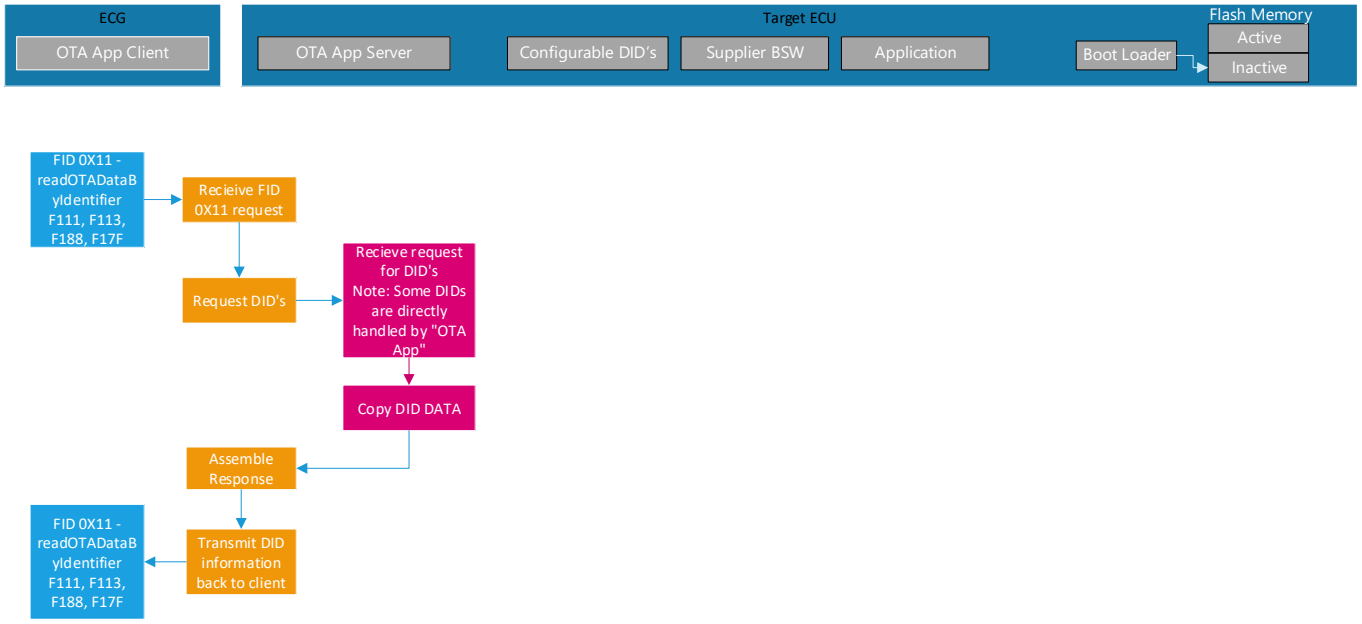


Figure 17: Read OTA Data by Identifier

5.1.11.2 Authorize Erase Memory

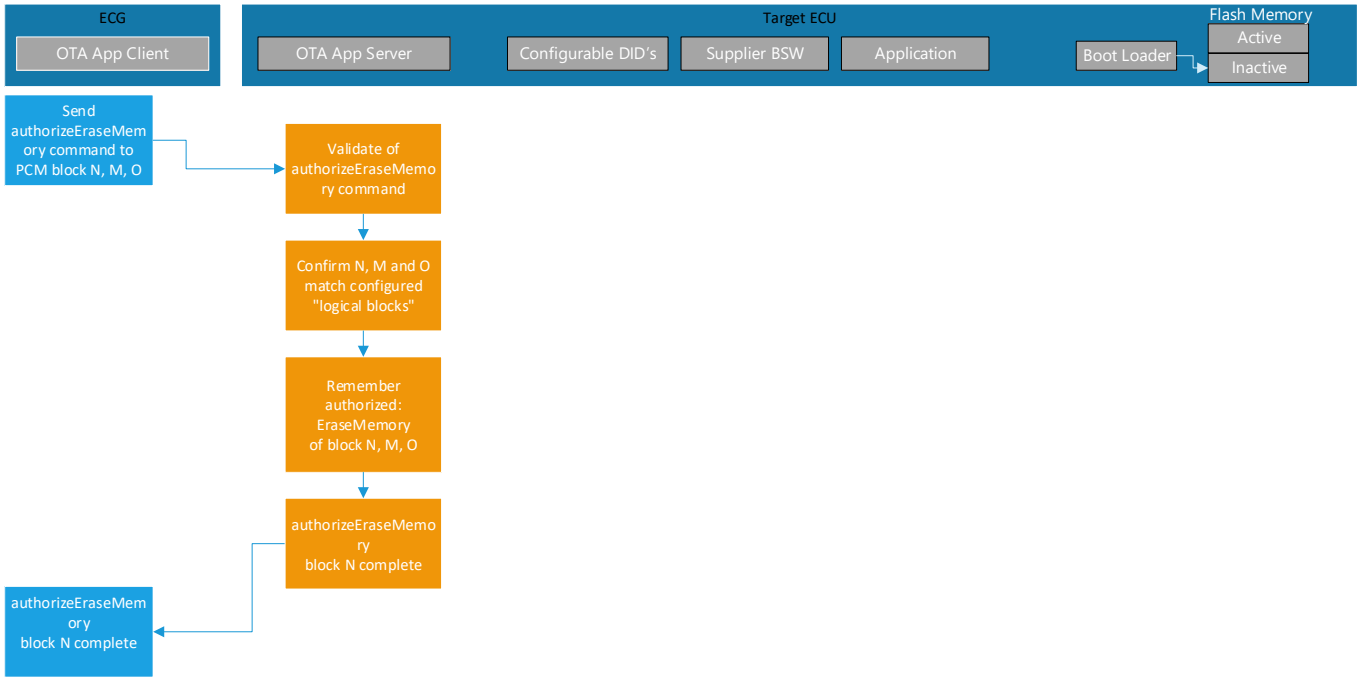


Figure 18: Authorize Erase Memory

5.1.11.3 Erase Memory

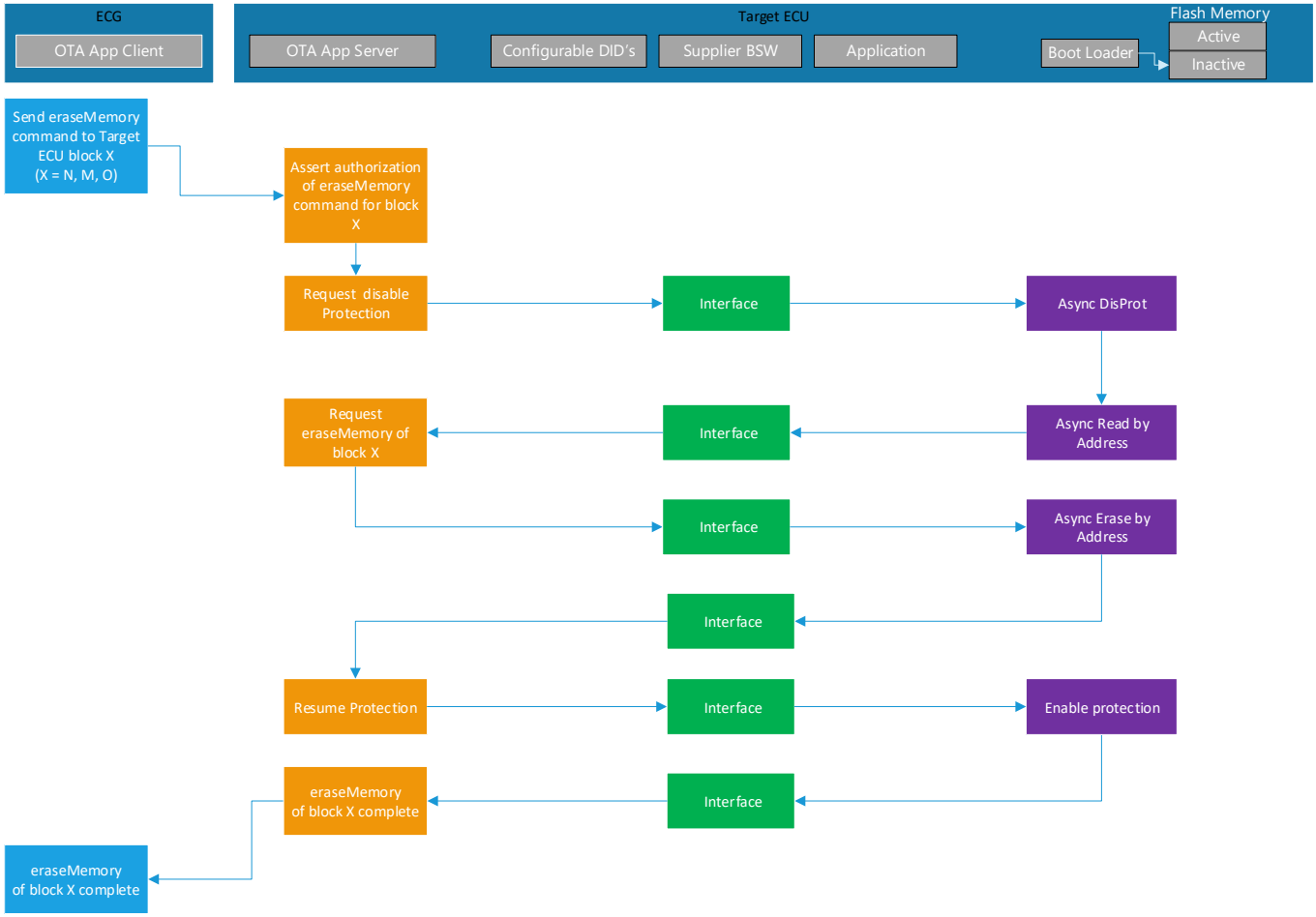


Figure 19: Erase Memory

5.1.11.4 Authorize Download

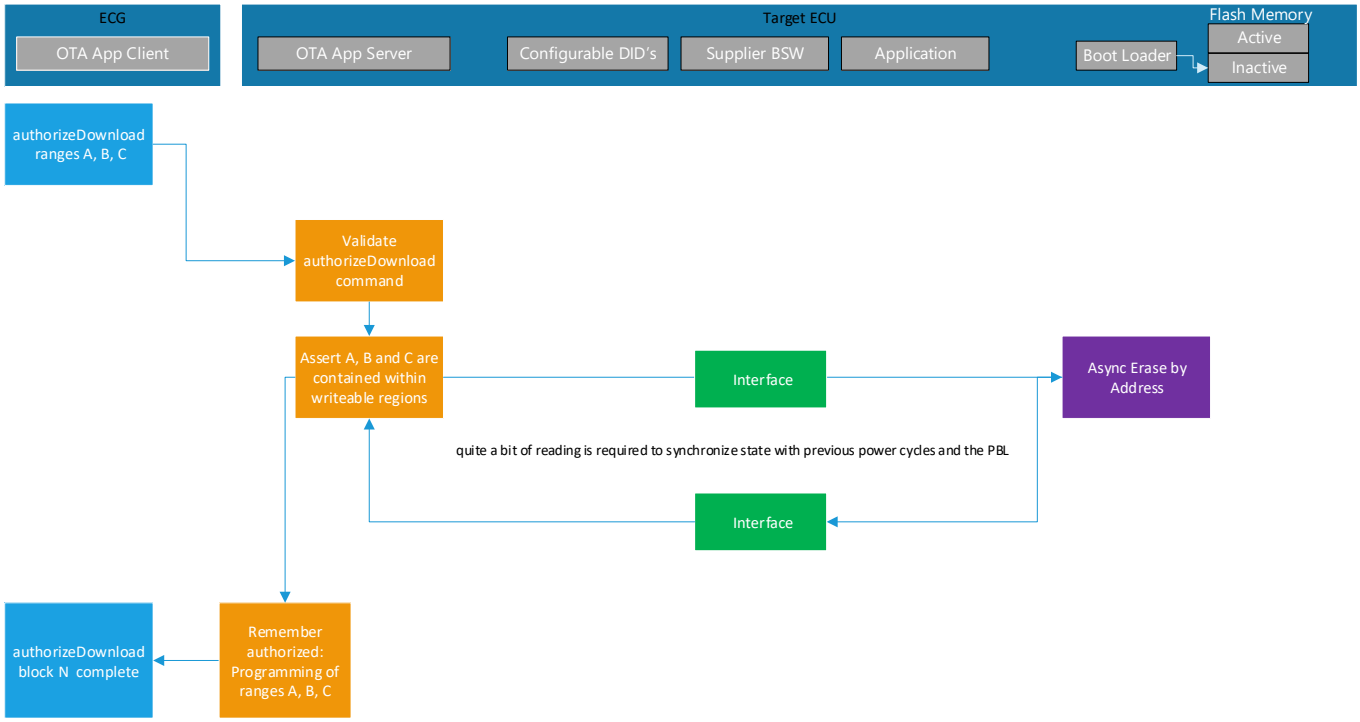


Figure 20: Authorize Download

5.1.11.5 Initiate Download

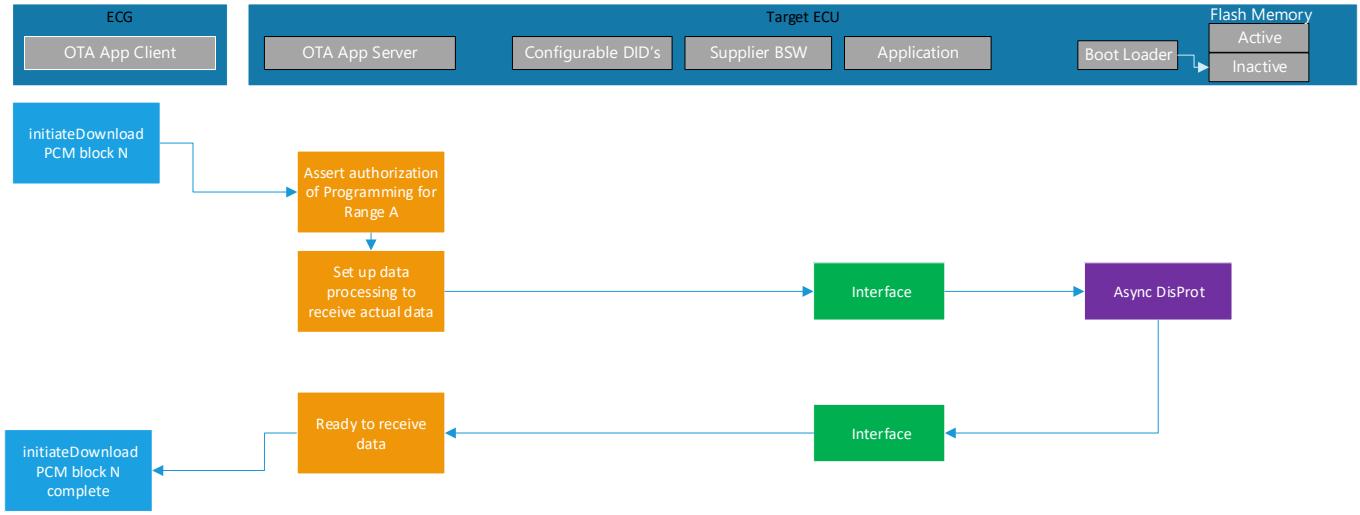


Figure 21: Initiate Download

5.1.11.6 Transfer Download Data

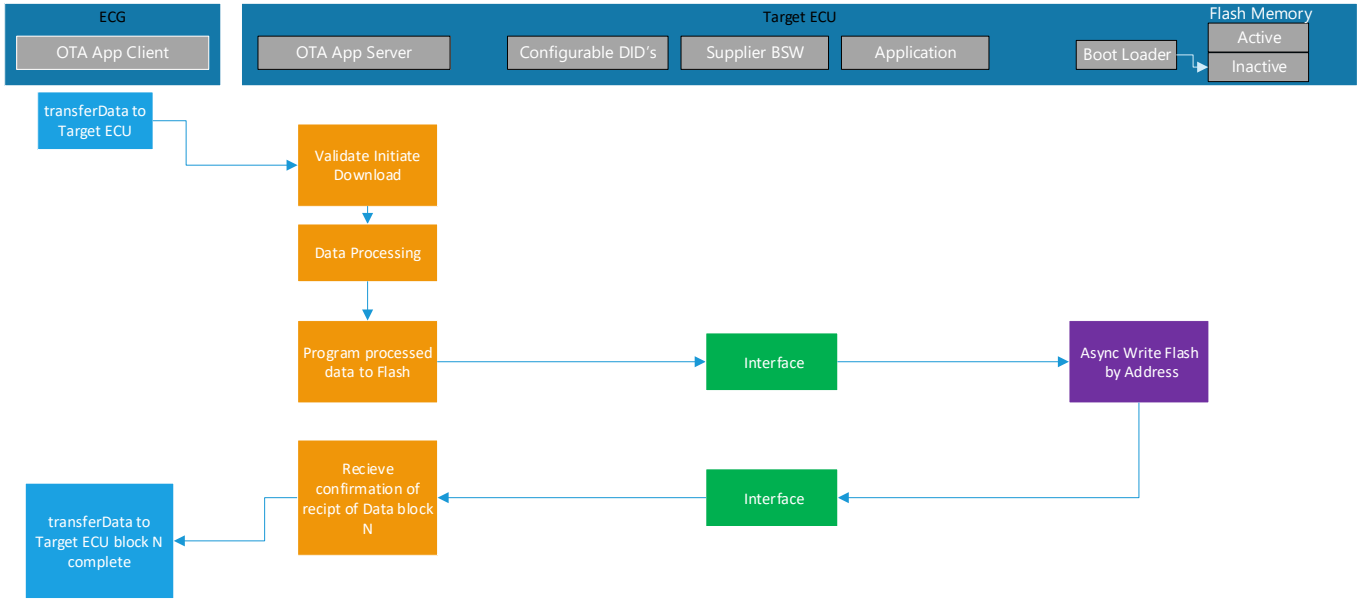


Figure 22: Transfer Download Data

5.1.11.7 Complete Download Data

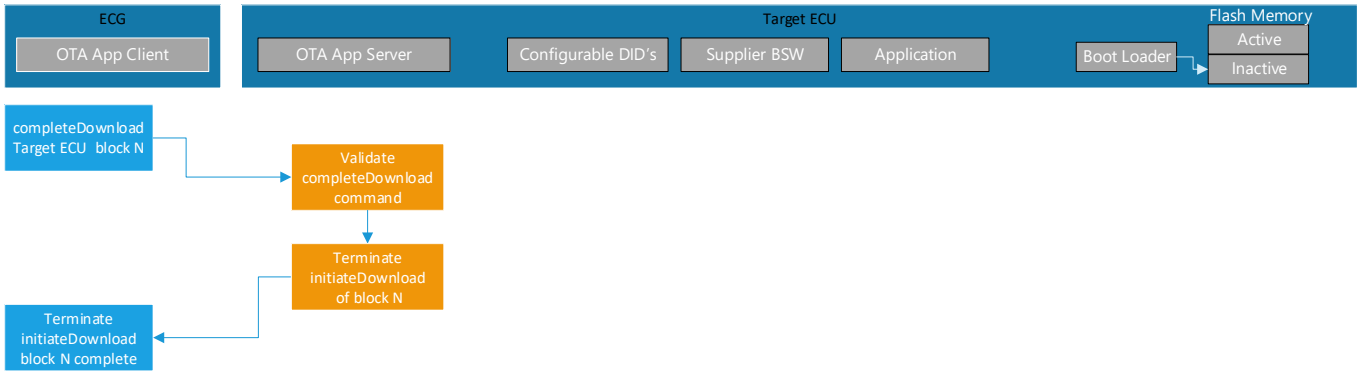


Figure 23: Complete Download Data

5.1.11.8 Validate Logical Block

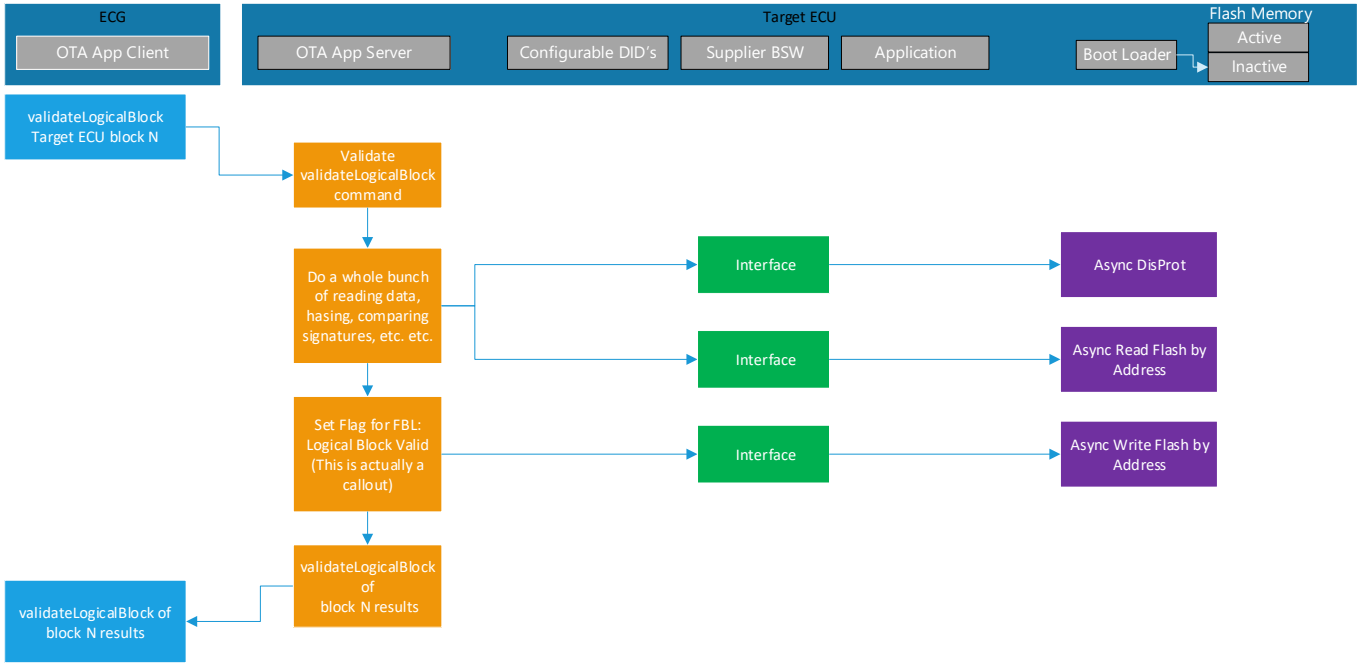


Figure 24: Validate Logical Block

5.1.11.9 Initiate Force Sync Counter

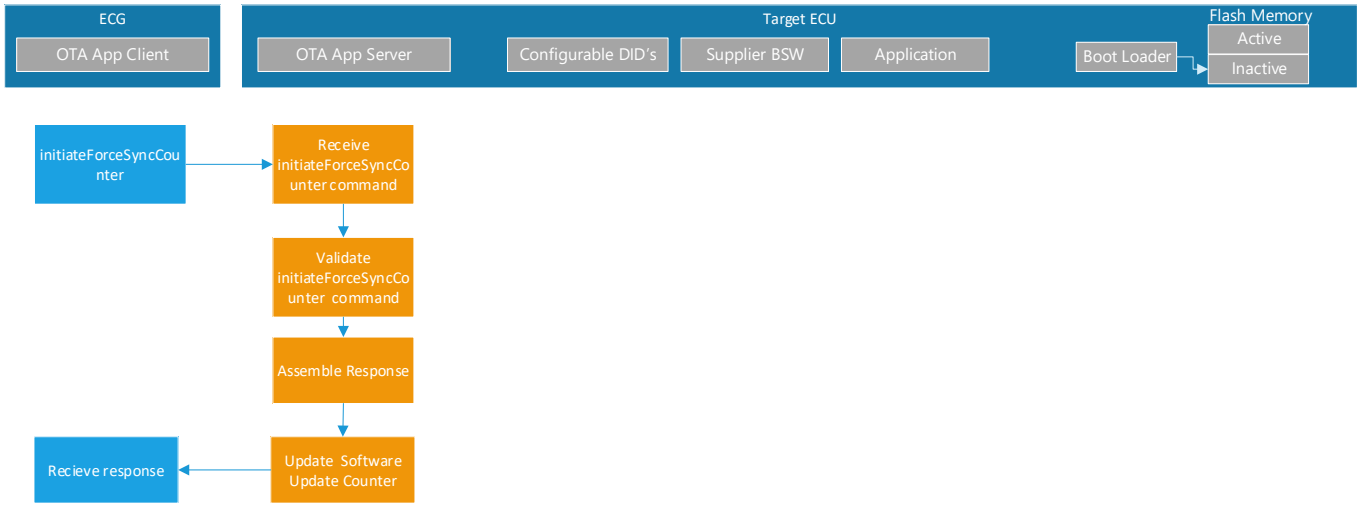


Figure 25: Initiate Force Sync Counter

5.1.11.10 Prepare for Activation

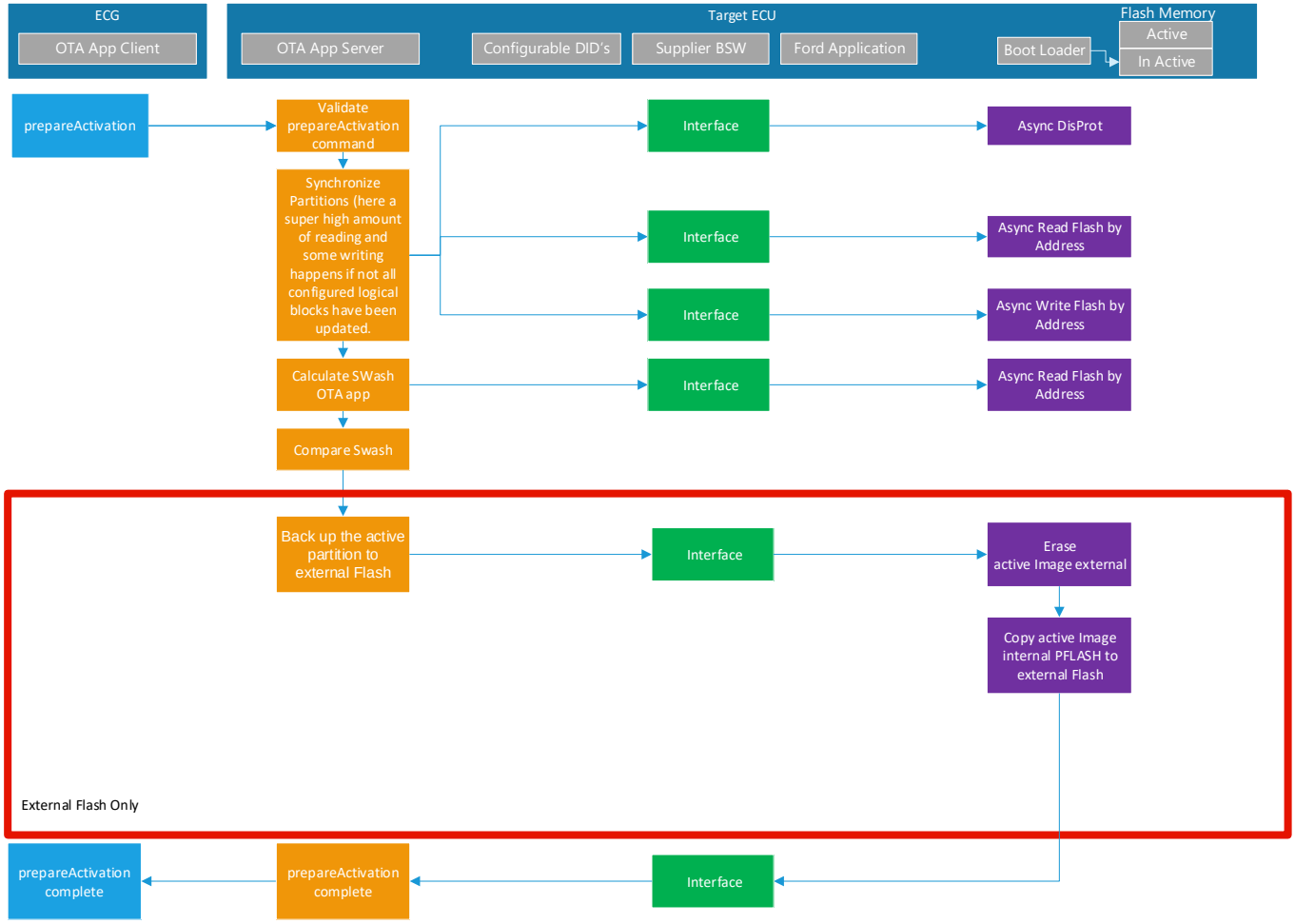


Figure 26: Prepare for Activation

5.1.11.11 Authorize Activation

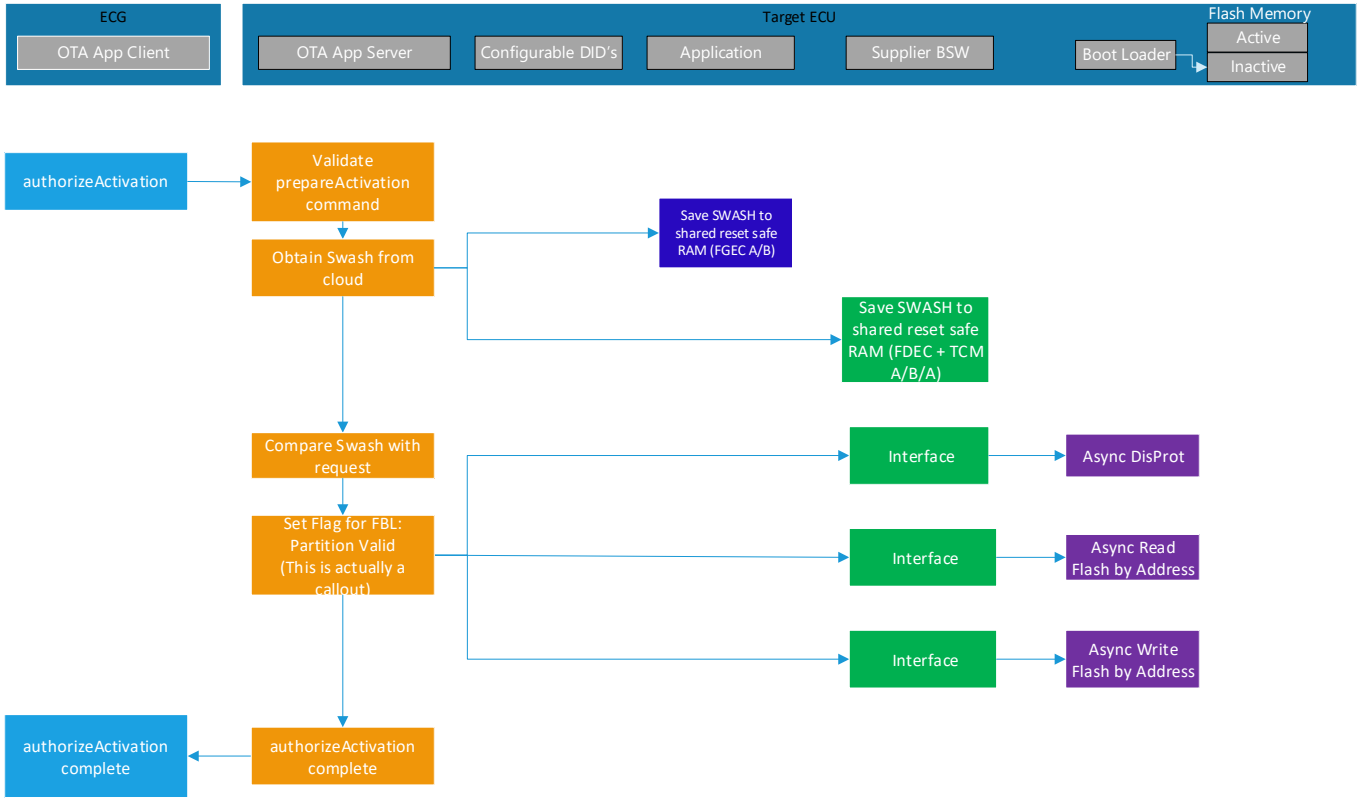


Figure 27: Authorize Activation

5.1.11.12 Initiate Activation

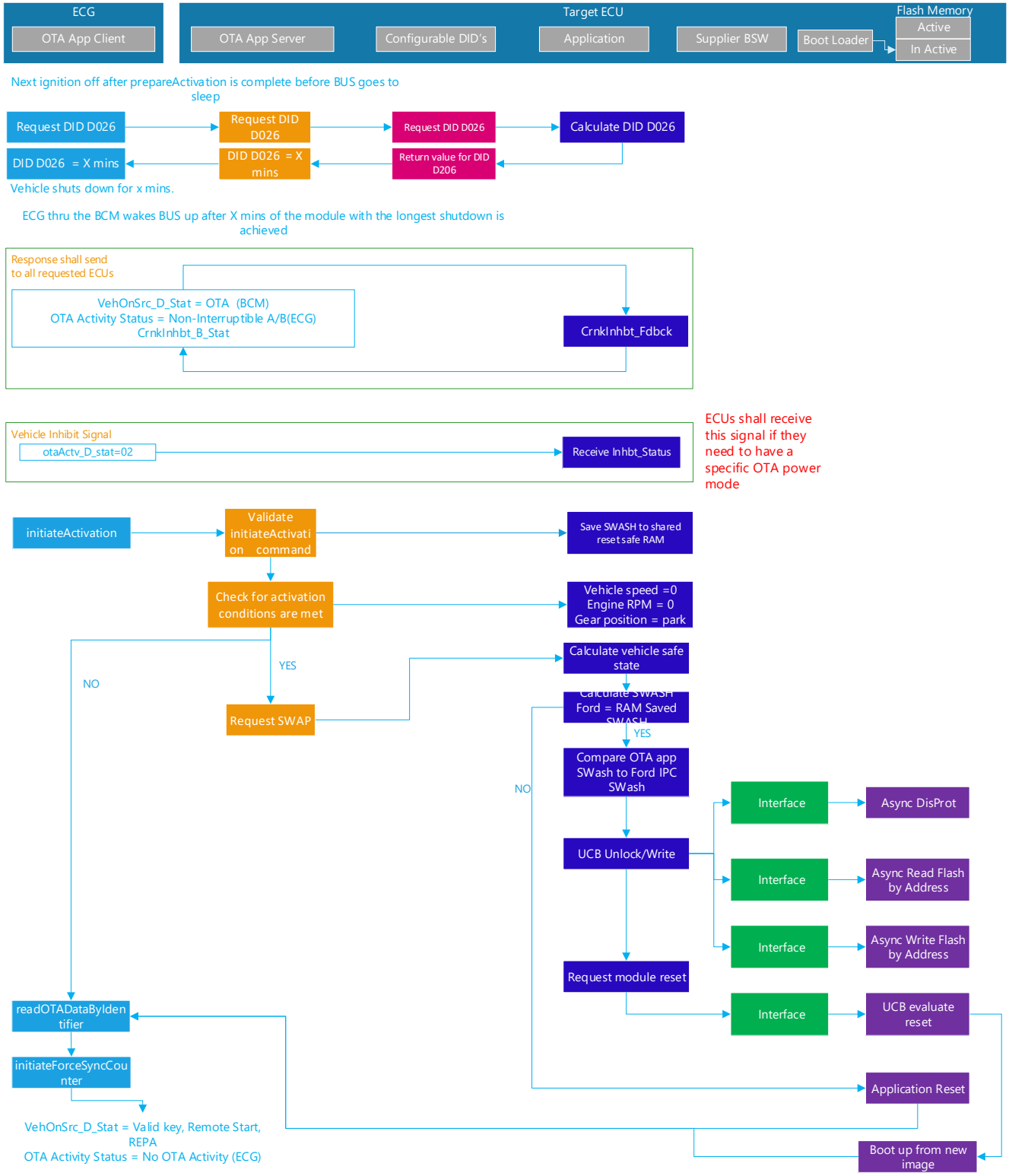


Figure 28: Initiate Activation

5.1.11.13 Initiate Rollback of in-active Flash Memory

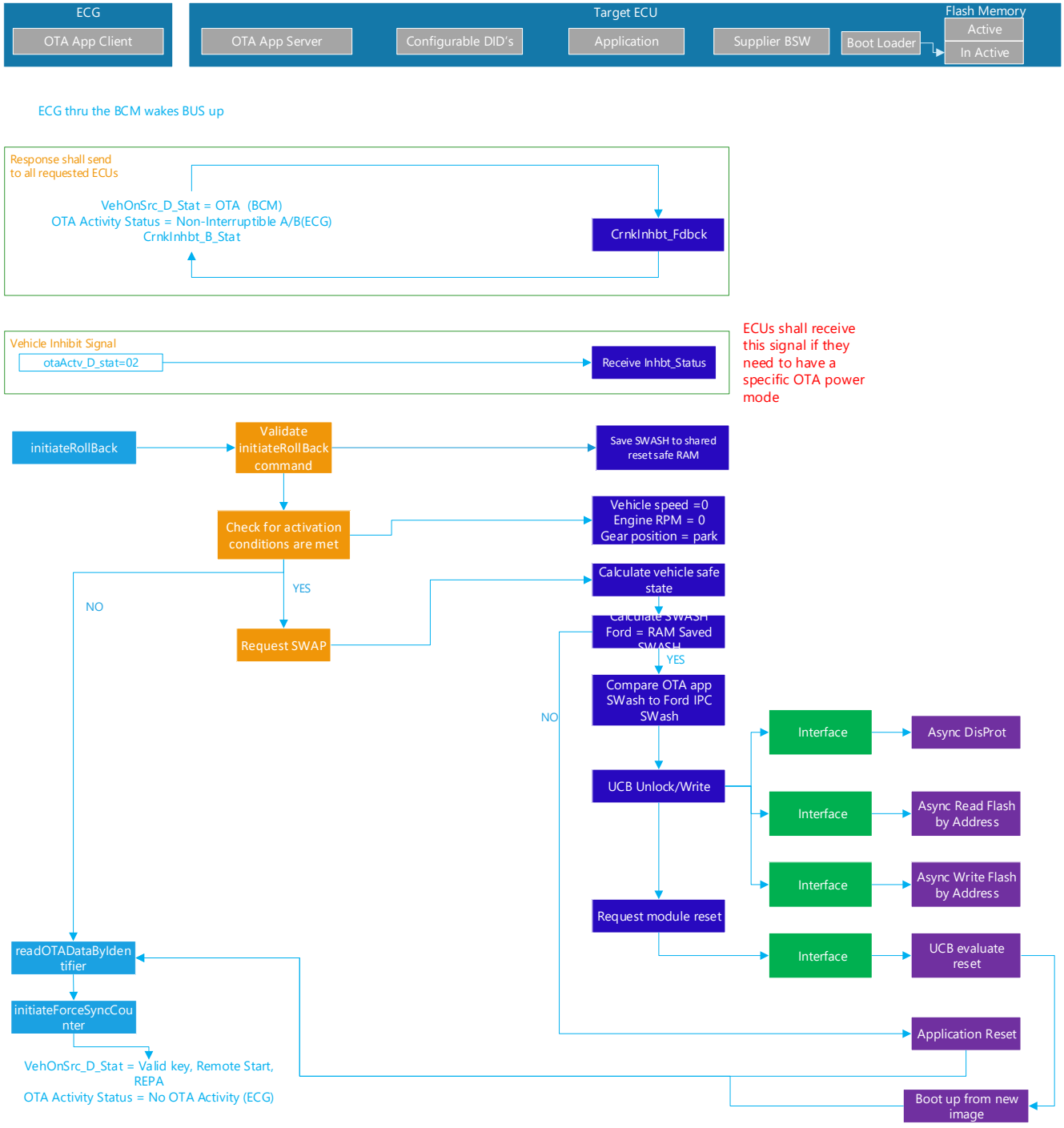


Figure 29: Initiate Rollback of in-active Flash Memory

5.1.12 Scenario: “Updating Target ECU which has two Micro Via OVTP”

5.1.12.1 Read OTA Data by Identifier for Two Micros

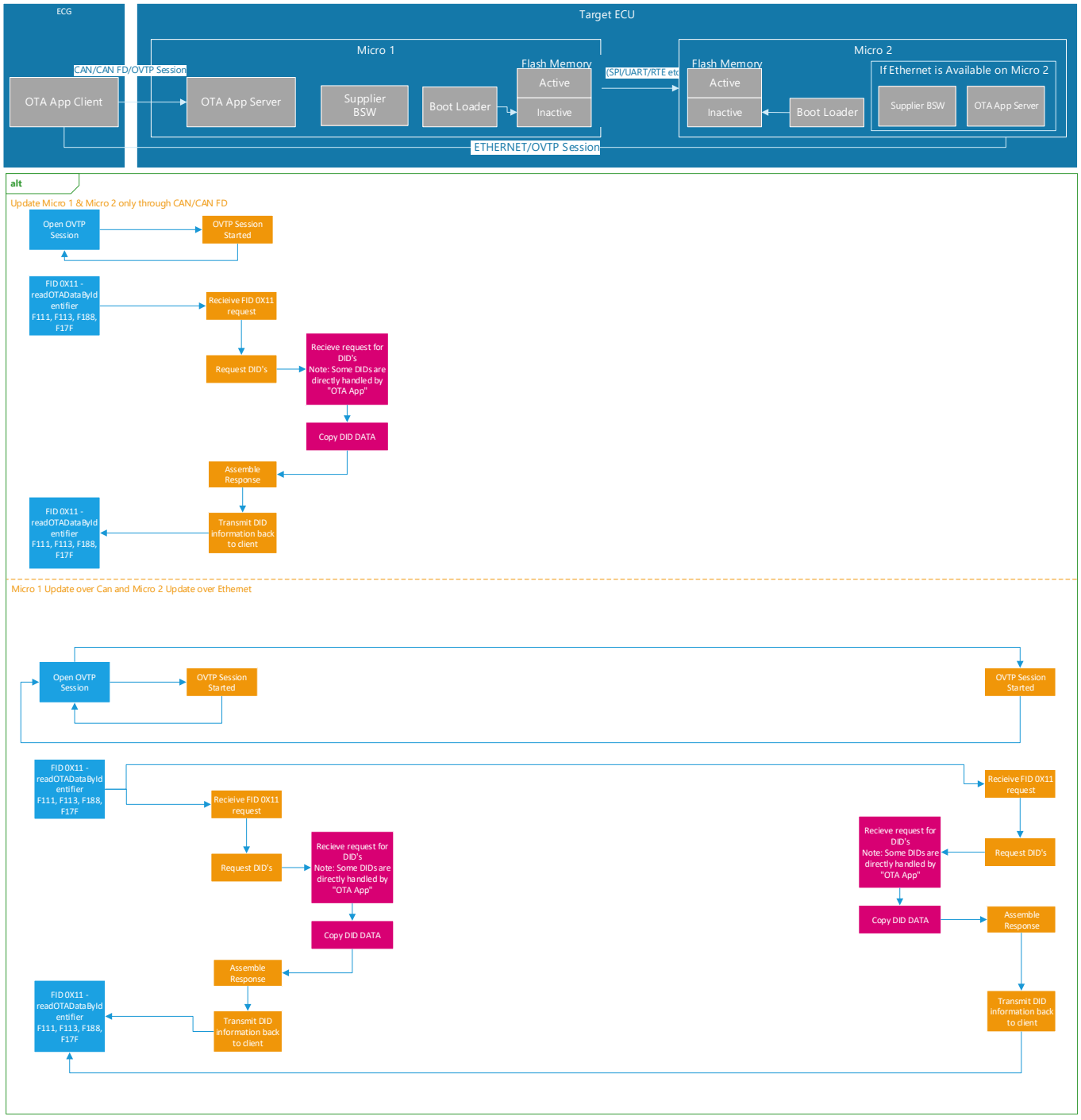
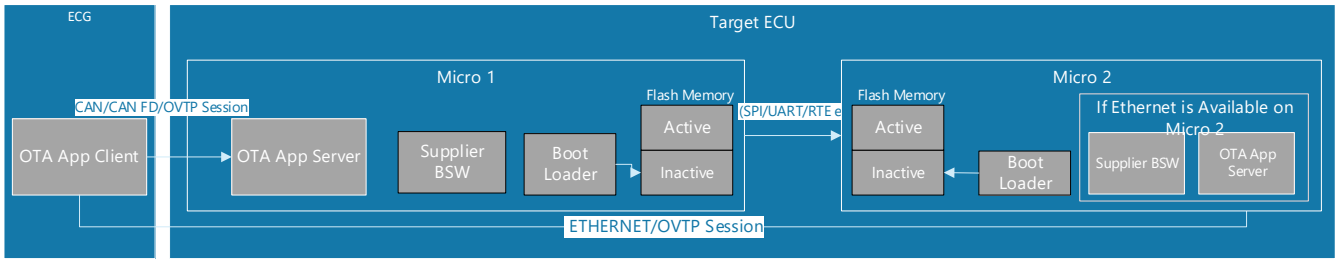


Figure 30: Read OTA Data by Identifier for Two Micros

5.1.12.2 Authorization for Erase Memory for Two Micros



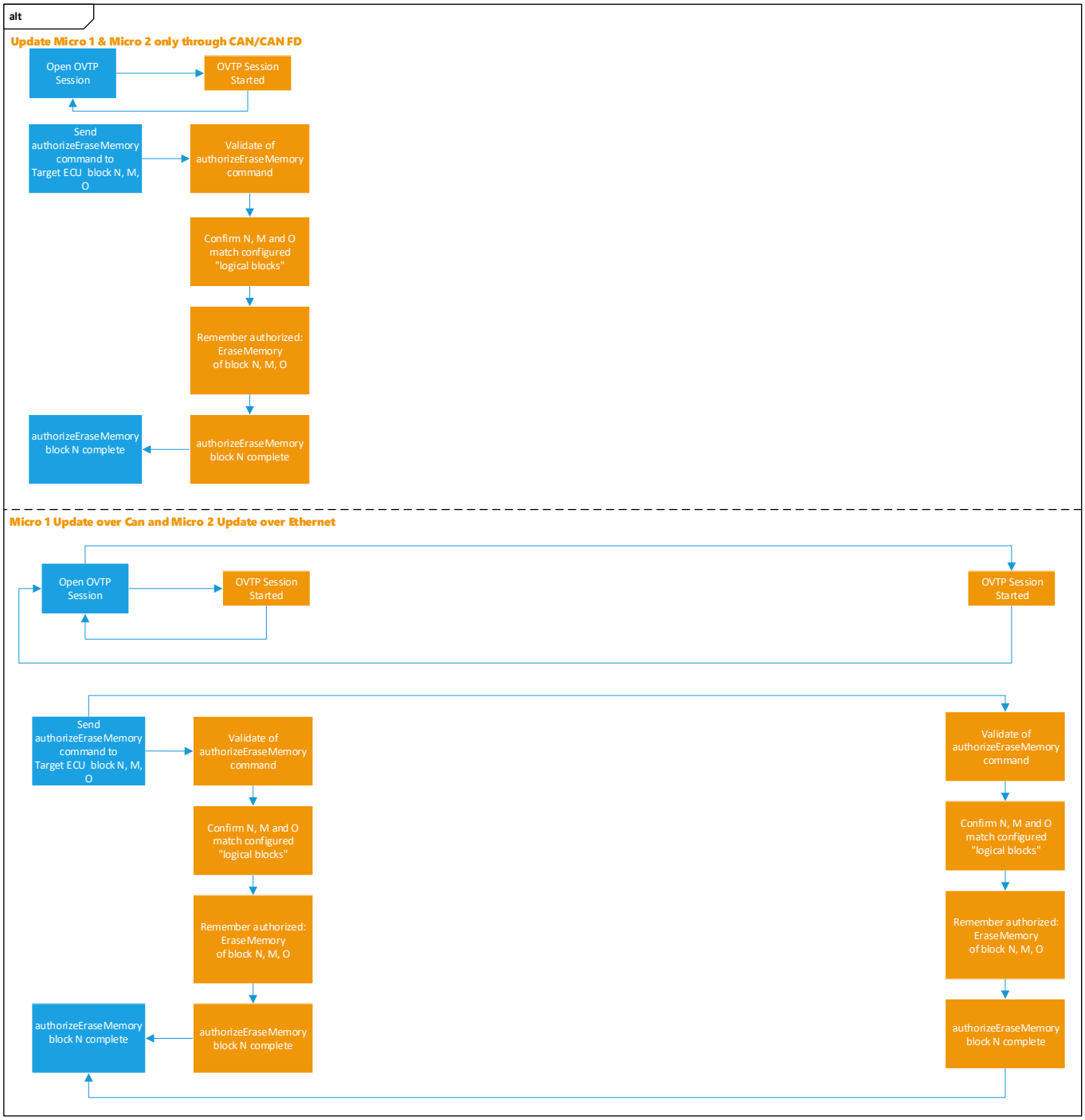
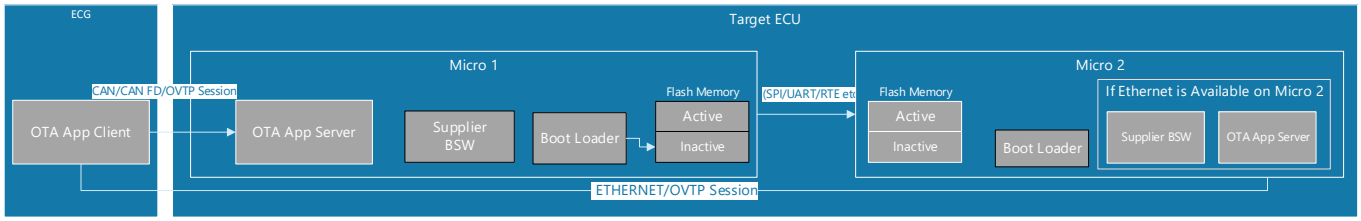


Figure 31: Authorization for Erase Memory for Two Micros

5.1.12.3 Erase Memory for both Micros of Target ECU Over Can/CanFD:



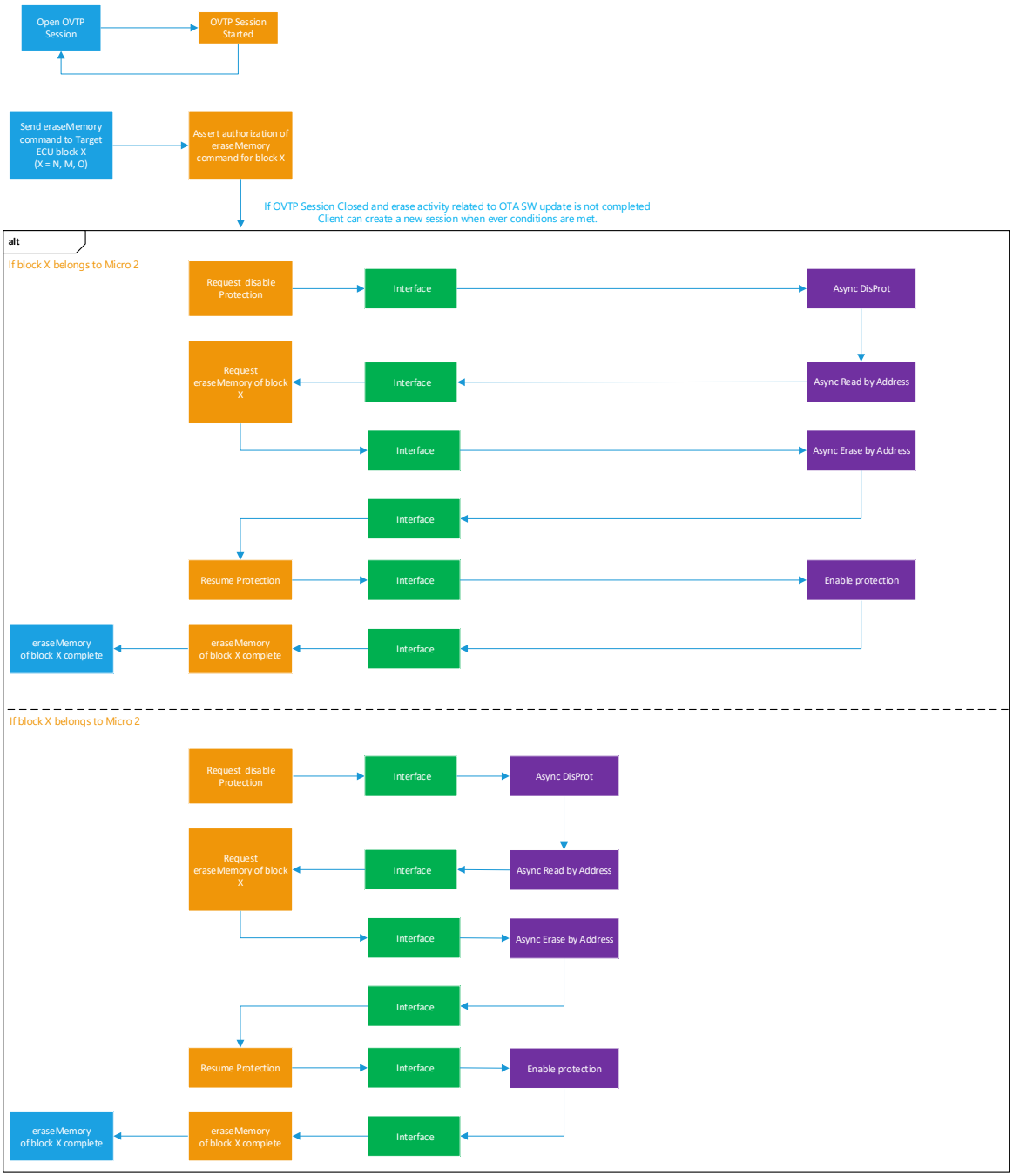


Figure 32: Erase Memory for both Micros of Target ECU Over Can/CanFD

5.1.12.4 Erase Memory Target ECU Micro 1 over Can/Can Fd and Micro 2 Over Ethernet:

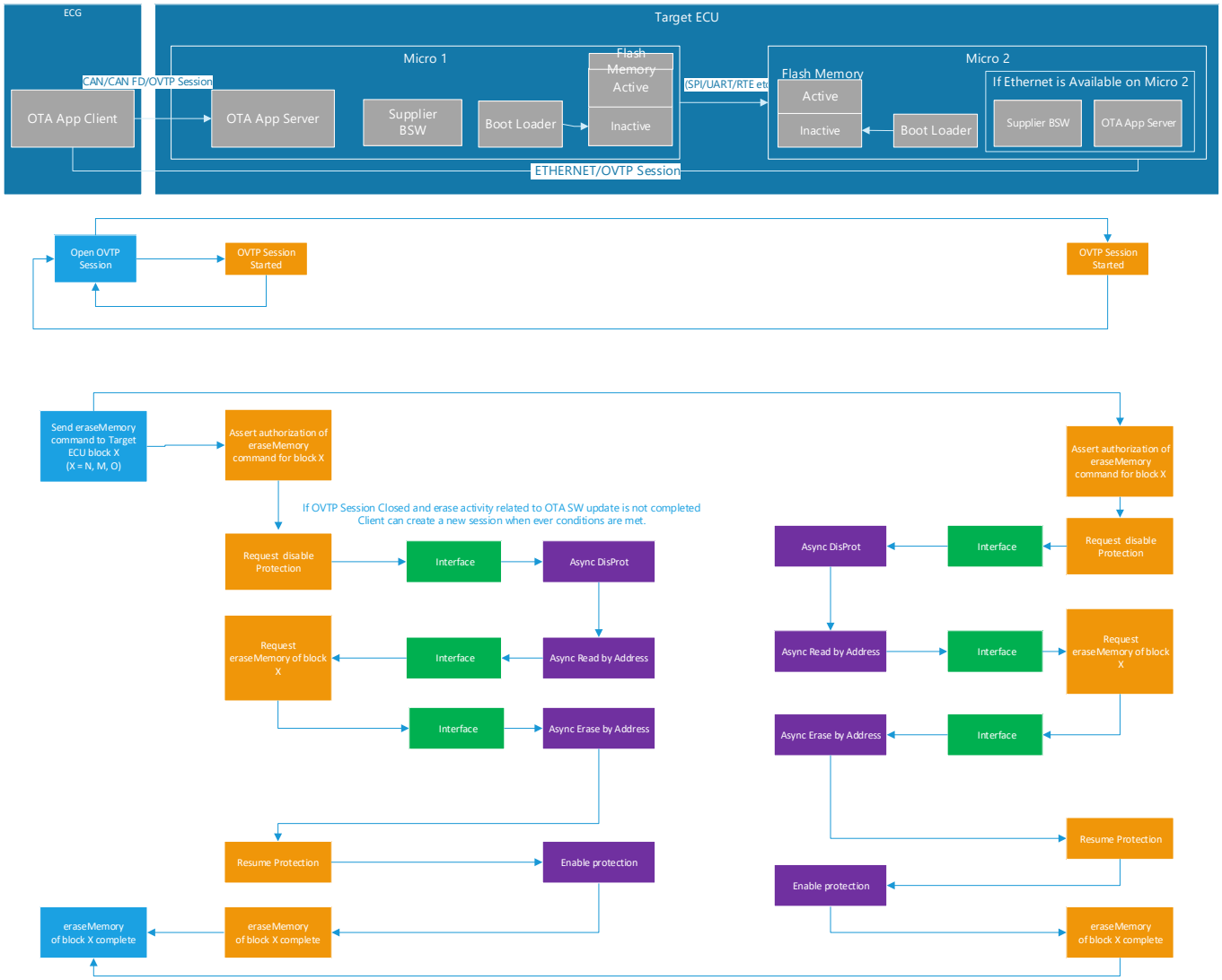
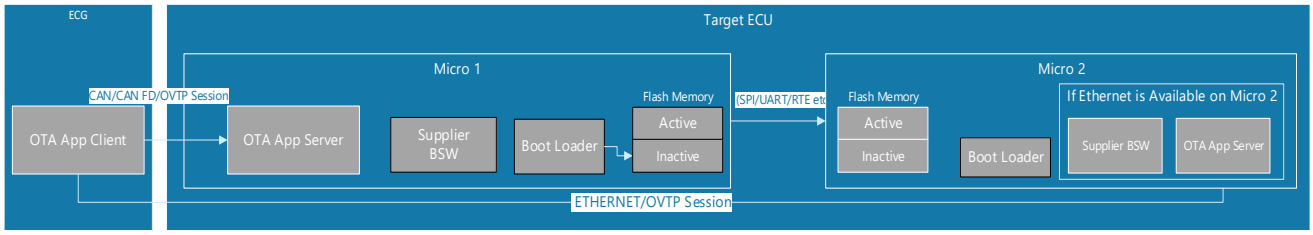


Figure 33: Erase Memory Target ECU Micro 1 over Can/Can Fd and Micro 2 Over Ethernet

5.1.12.5 Authorize Download for Both Micros of Target ECU:



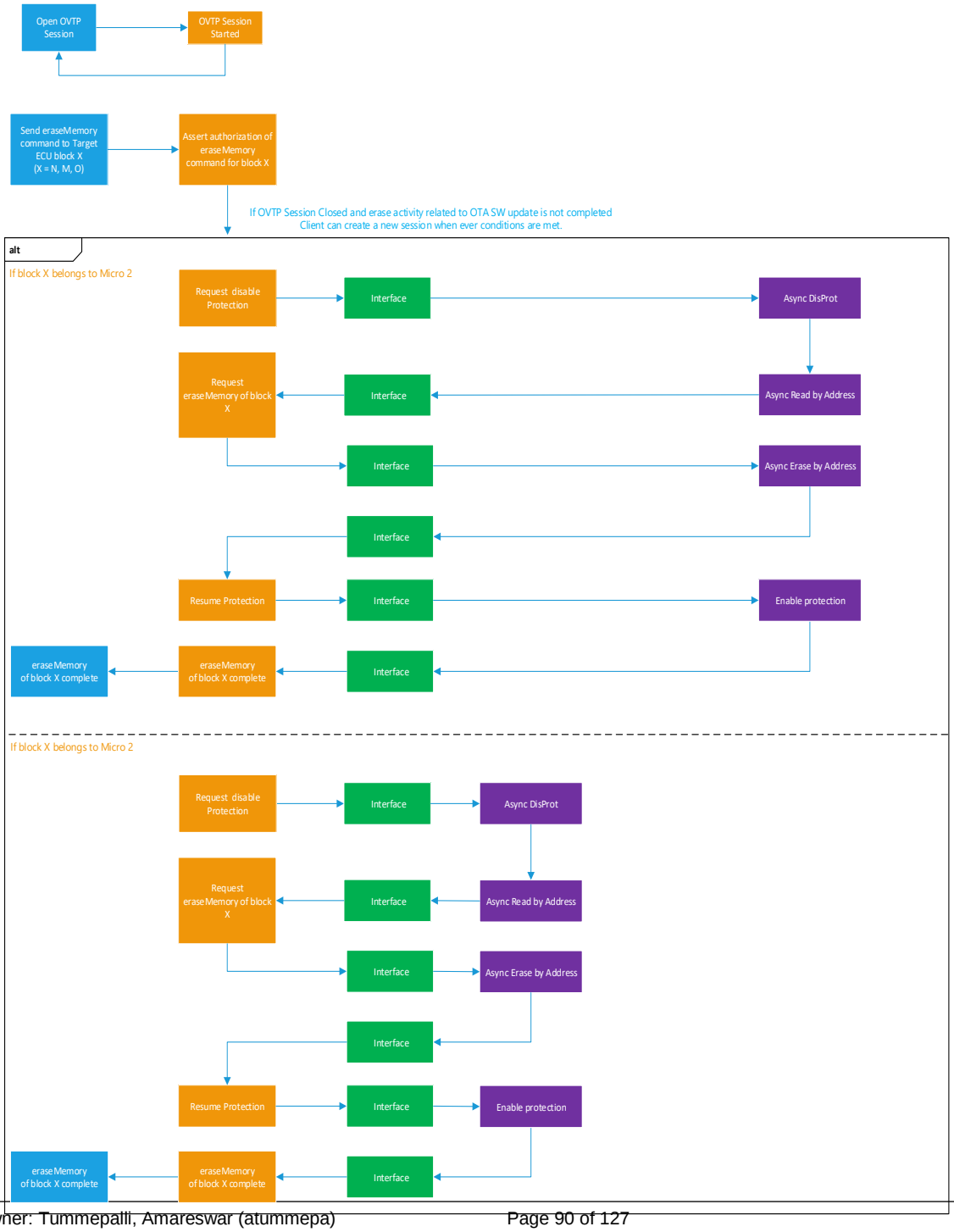


Figure 4: Authorize Download for Both Micros of Target ECU

5.1.12.6 Initiate Download for Both Micros of Target ECU:

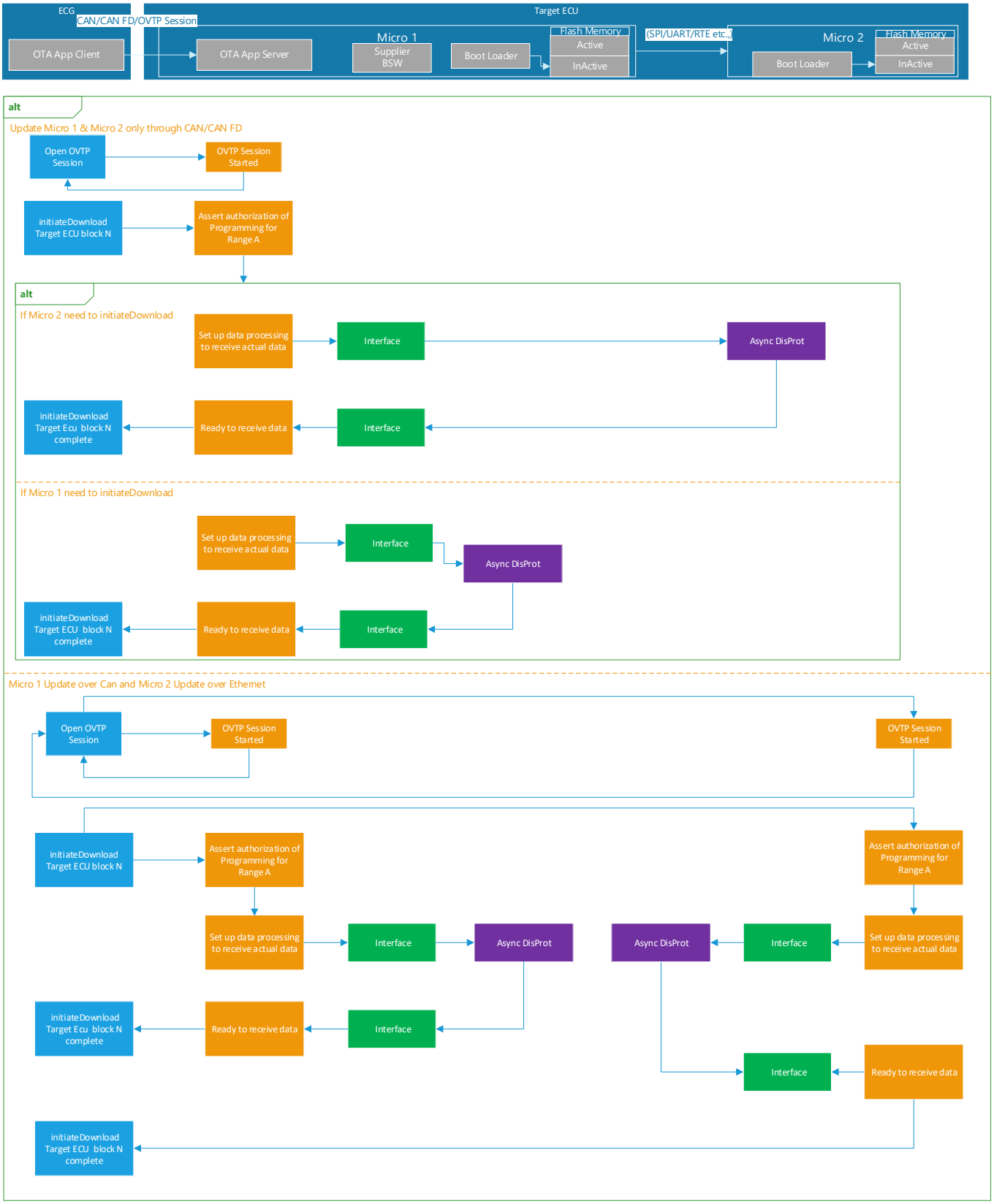


Figure 35: Initiate Download for Both Micros of Target ECU

5.1.12.7 Transfer OTA Update Download to Both Micros of Target ECU

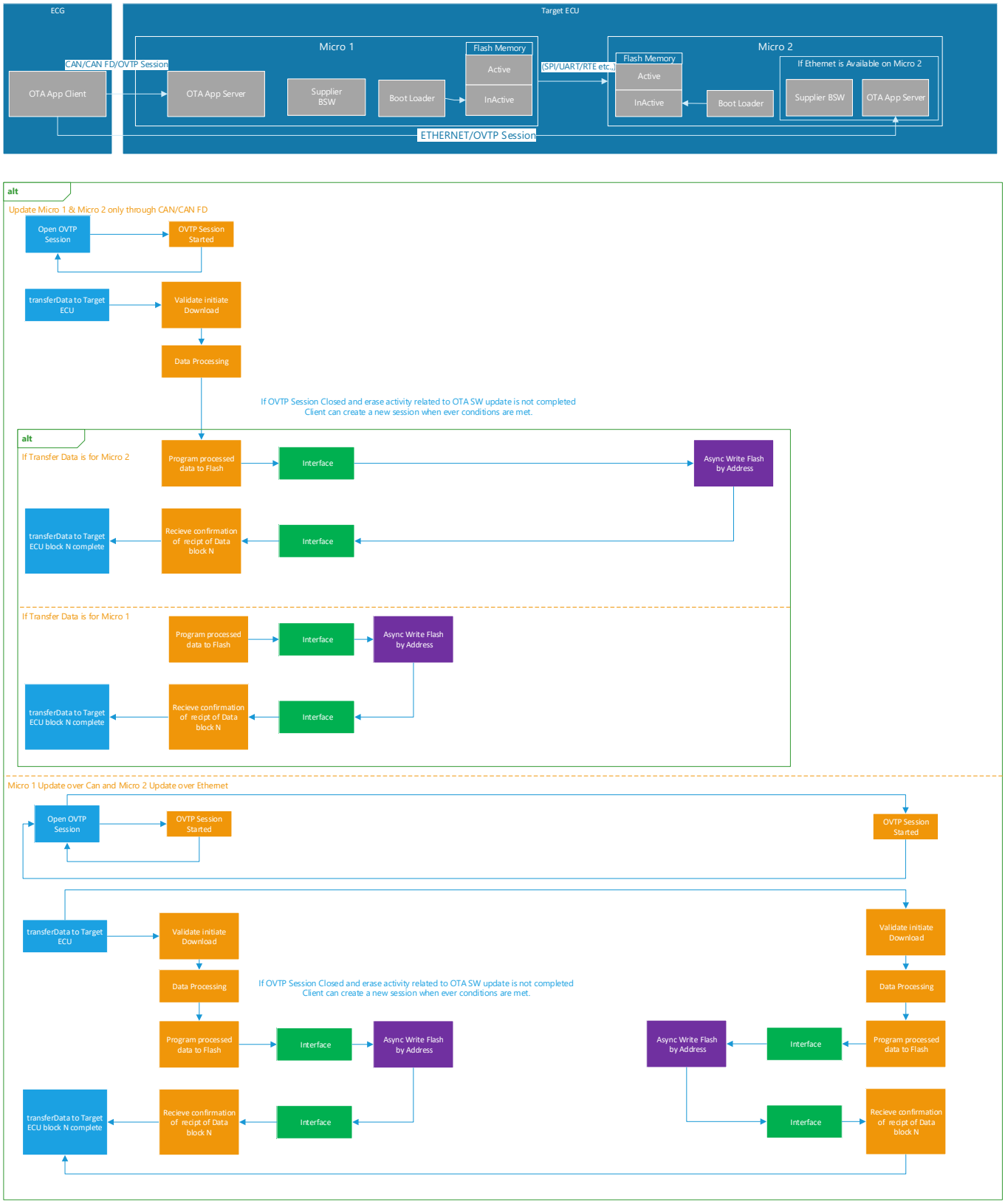


Figure 36: Transfer OTA Update Download to Both Micros of Target ECU

5.1.12.8 Complete Download for both Micros

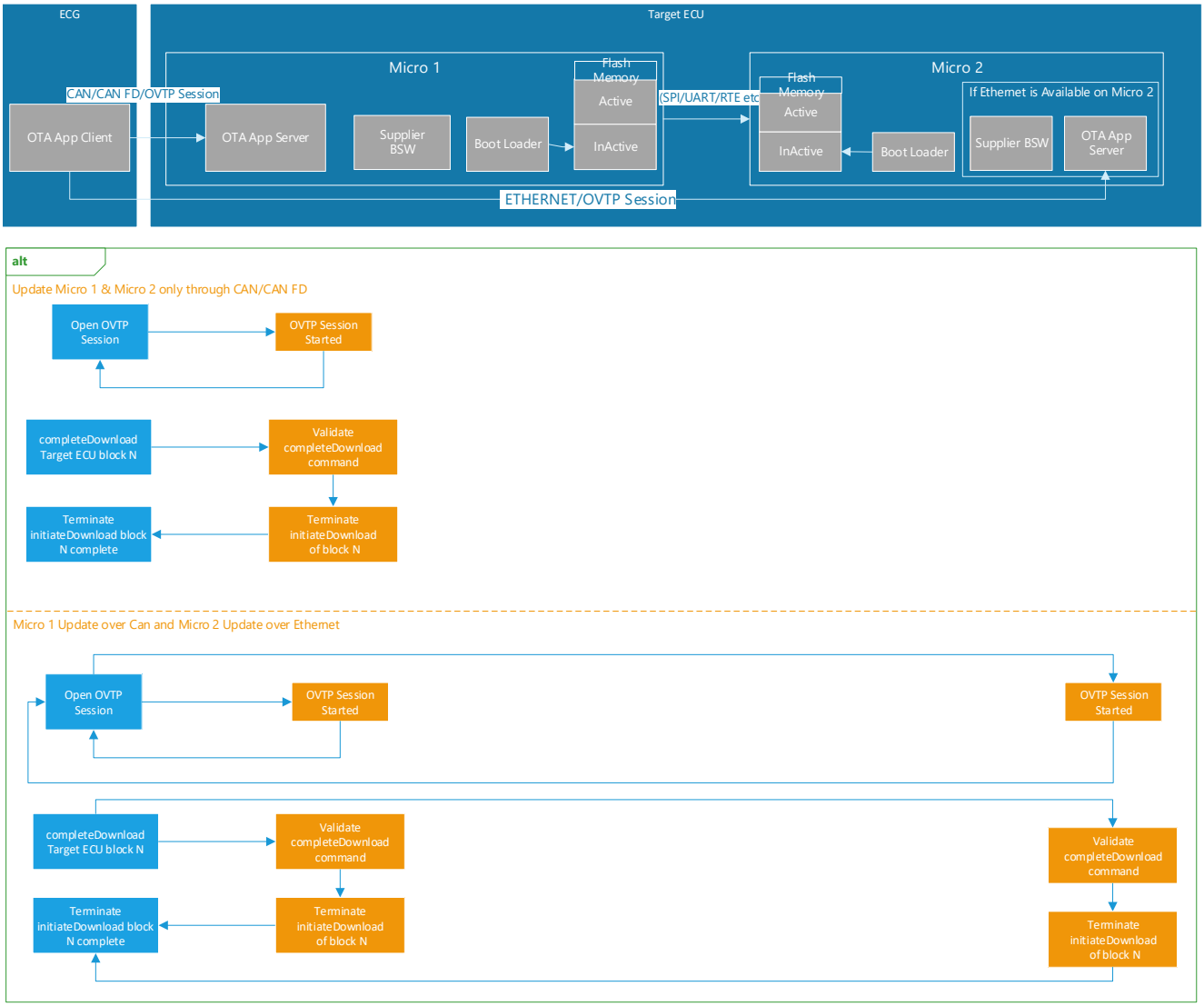


Figure 37: Complete Download for both Micros

5.1.12.9 Validate Logical Block for both Micros through CAN/CANFD

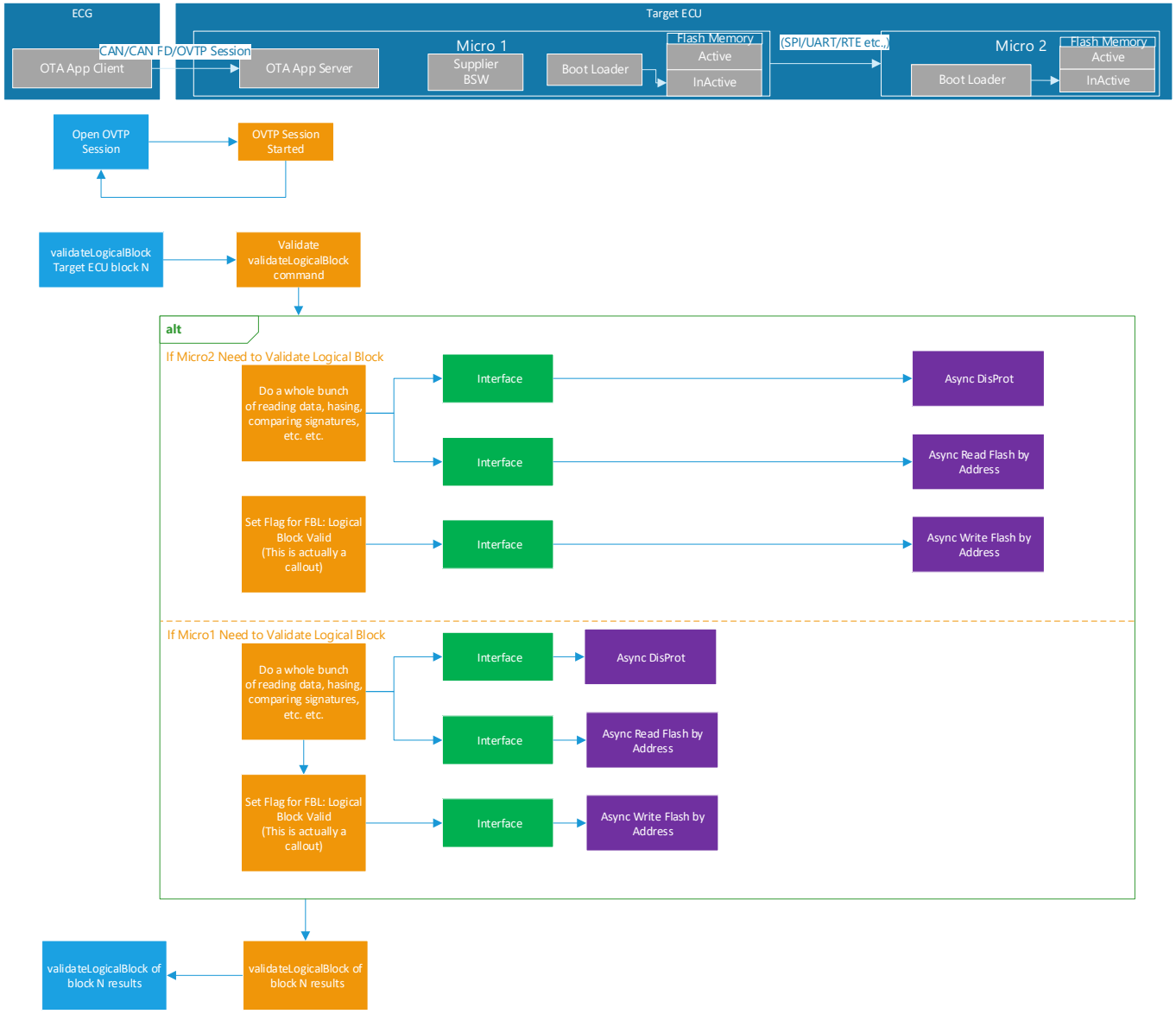


Figure 38: Validate Logical Block for both Micros through CAN/CANFD

5.1.12.10 Validation of Logical Block for Micro1 Via Can/CanFd and Micro2 Over Ethernet

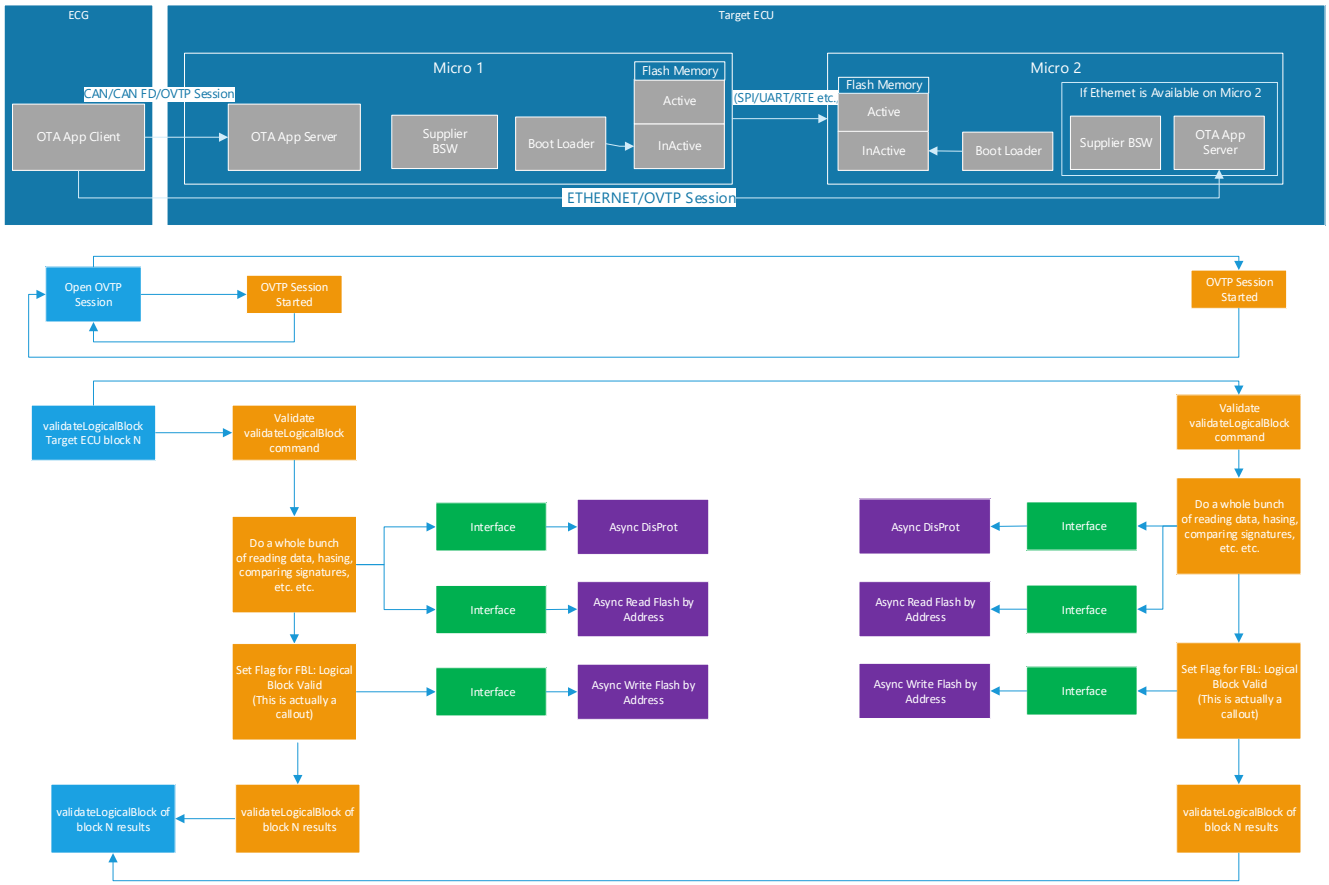


Figure 39: Validation of Logical Block for Micro1 Via Can/CanFd and Micro2 Over Ethernet

5.1.12.11 Initiate Force Sync Counter for Both Micros

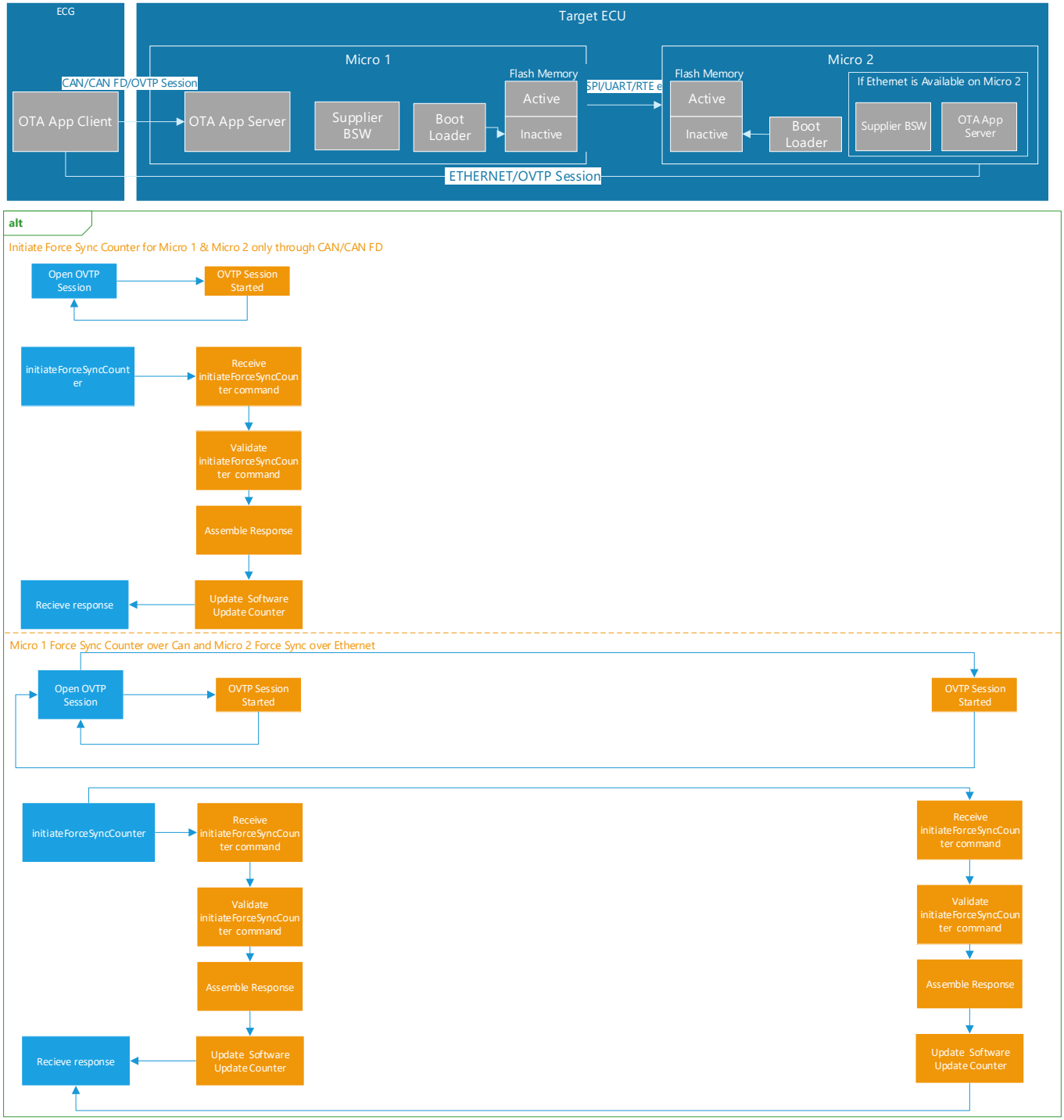


Figure 40: Initiate Force Sync Counter for Both Micros

5.1.12.12 Prepare for Activation for Both Micros

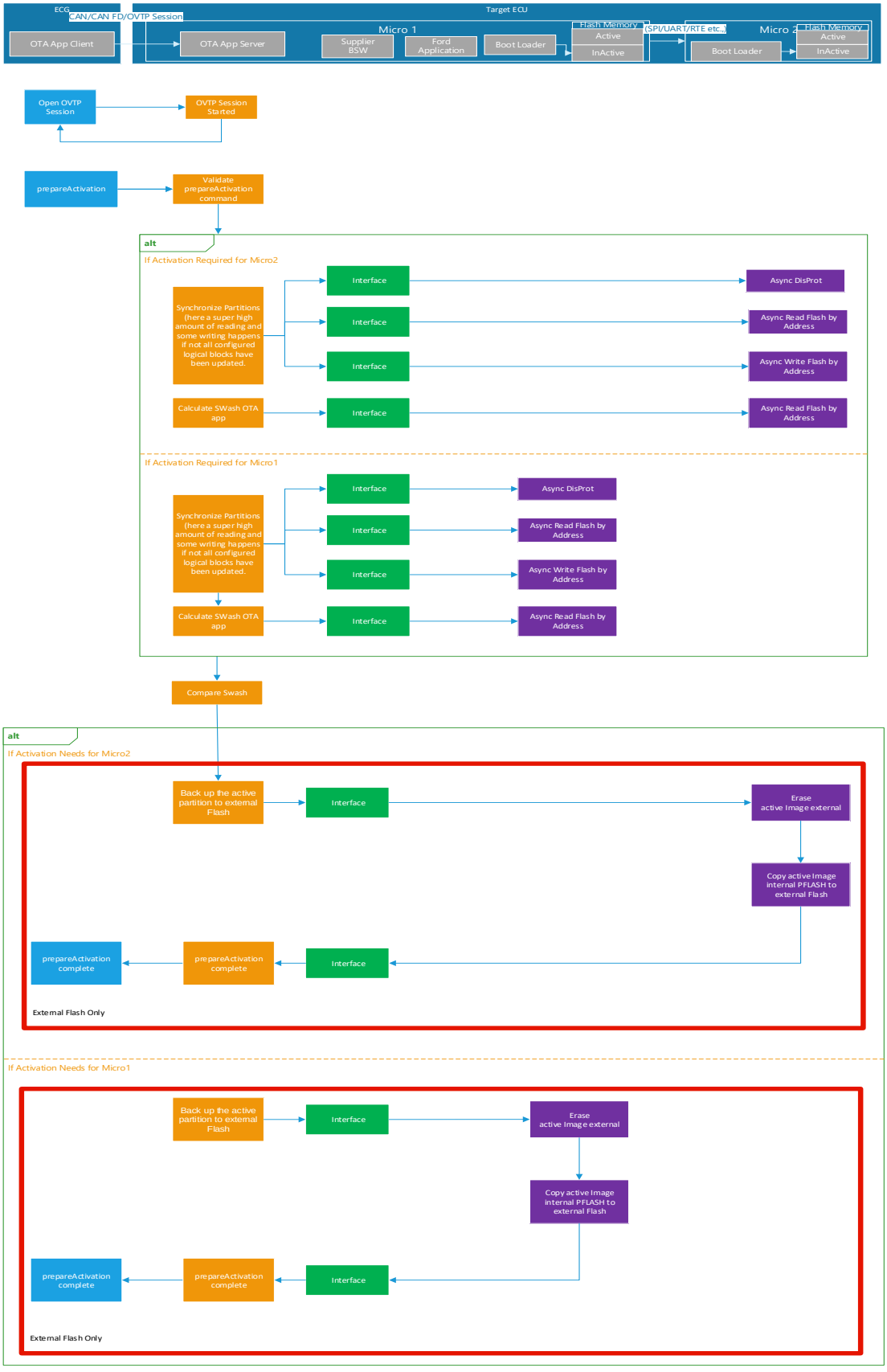


Figure 41: Prepare for Activation for both Micros

5.1.12.13 Authorize Activation for both Micros

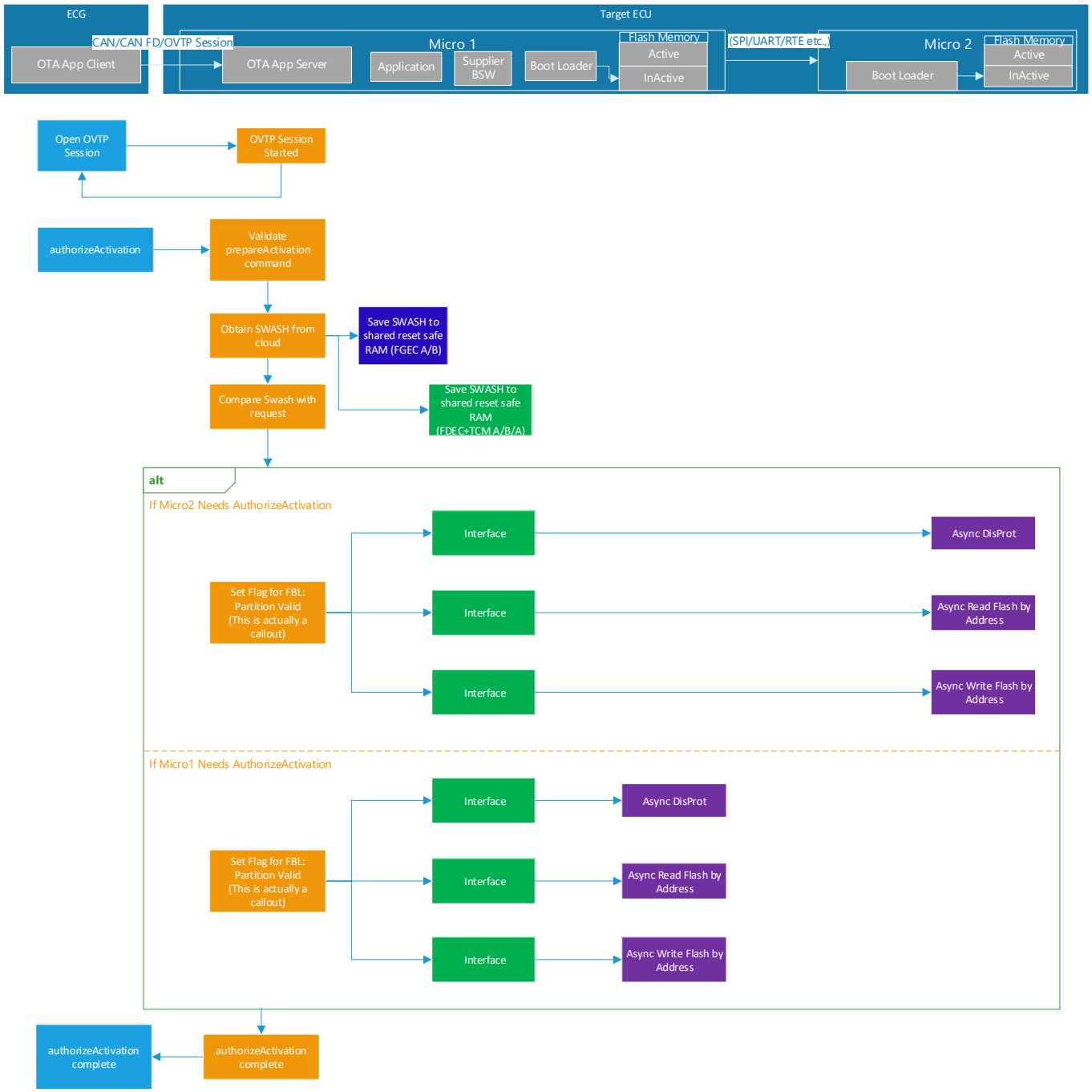


Figure 42: Authorize Activation for both Micros

5.1.12.14 Initiate Activation for both Micros

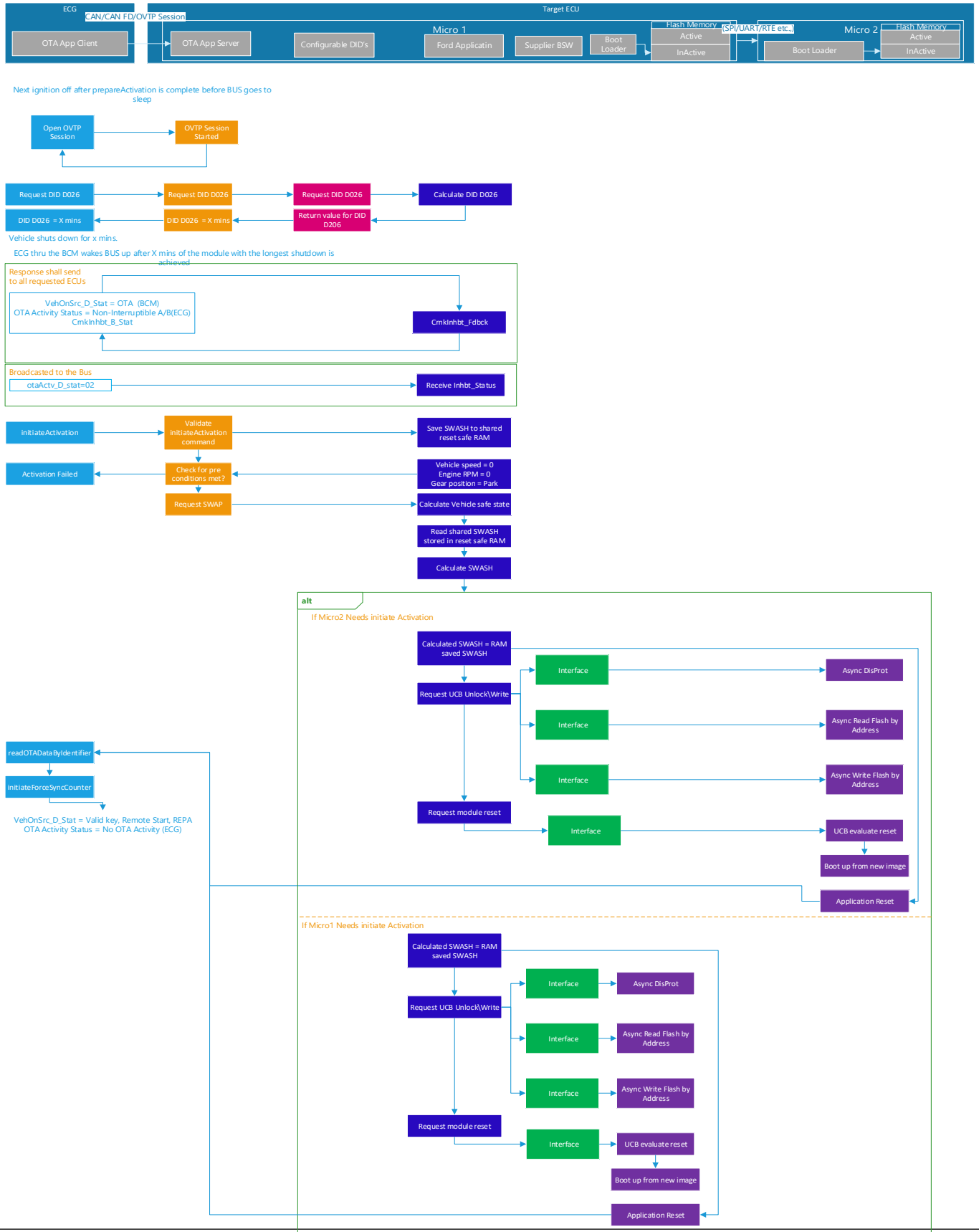


Figure 43: Initiate Activation for both Micros

5.1.12.15 Initiate RollBack for both Micros

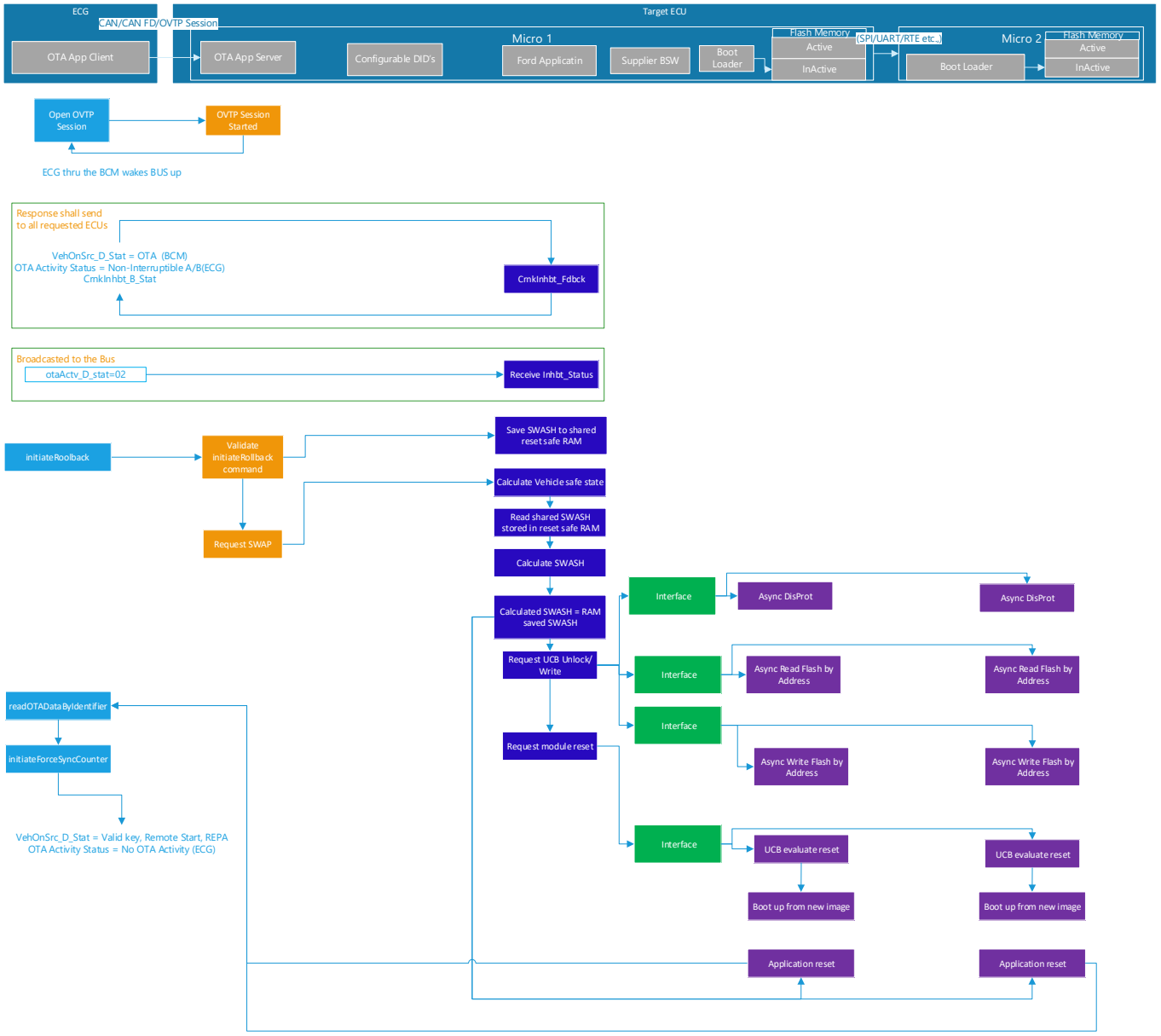


Figure 44: Initiate RollBack for both Micros

5.1.13 DC Configuration Scenario: “Change Parameter Over The Air”

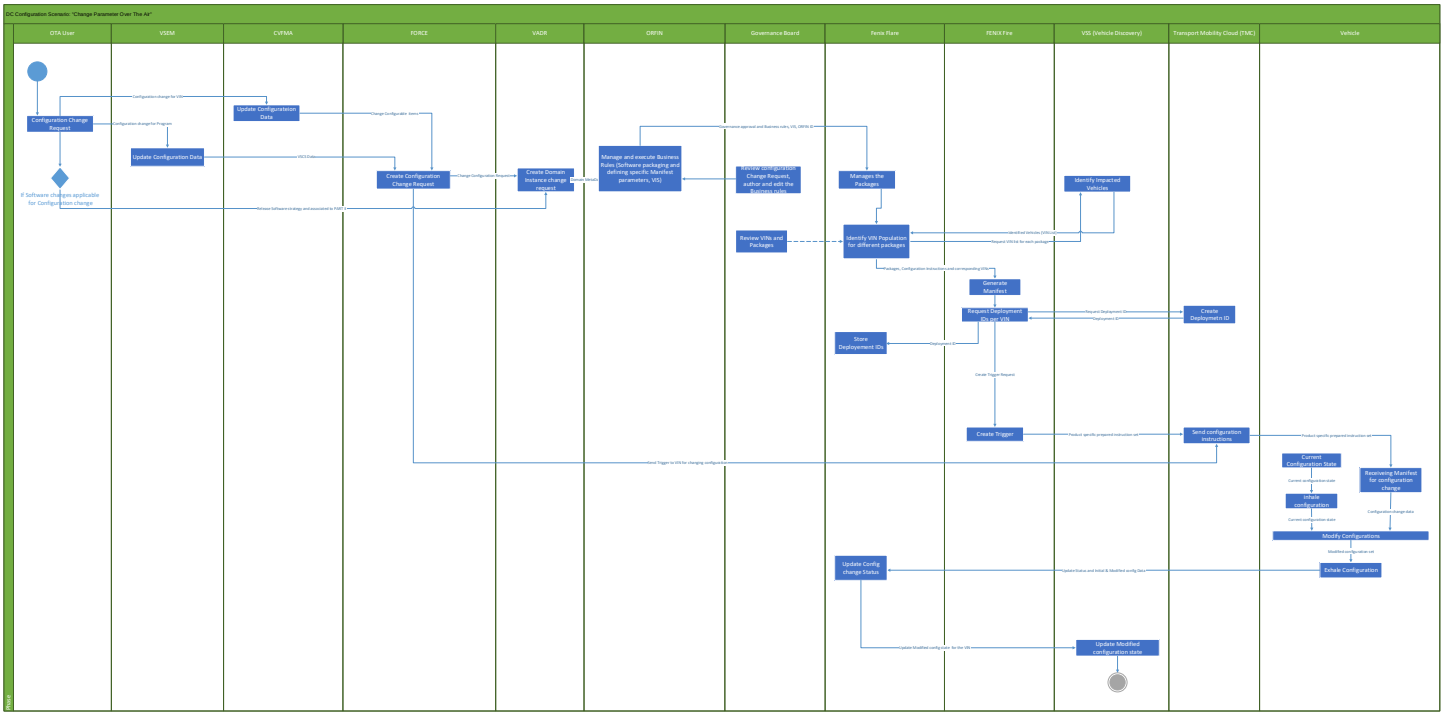


Figure 5: DC Configuration Scenario: “Change Parameter Over The Air”

5.1.14 REQ-365946/B-OTA Update during vehicle in transport mode

In Vehicle Software Update Vehicle FIS

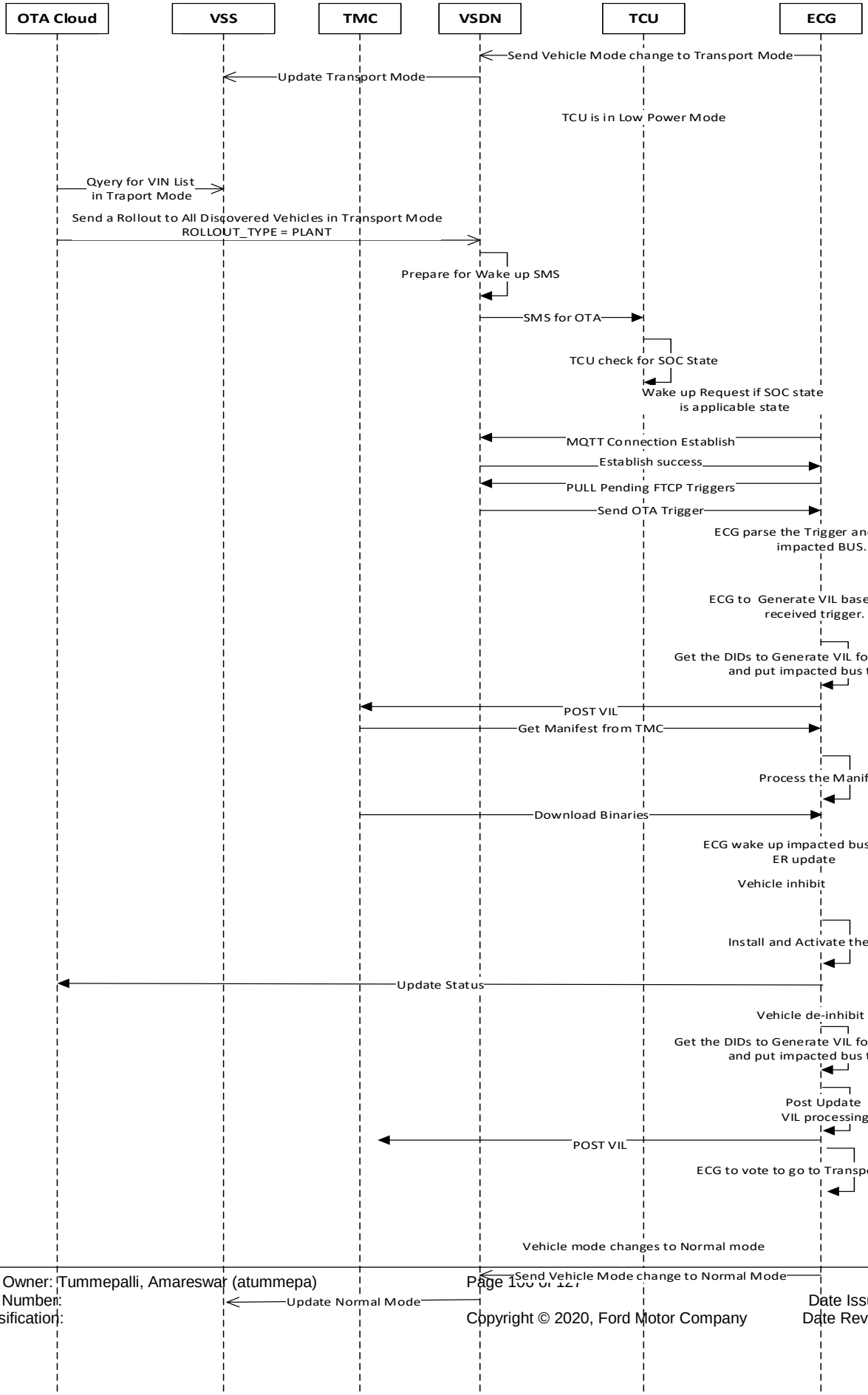


Figure 6: OTA Update during vehicle in transport mode

5.2 Component Interface Behavior Diagrams

6 Feature Implementation Requirements

6.1 Requirements Derivation Diagram

6.2 Requirements

6.2.1 Requirements on Electrical Components

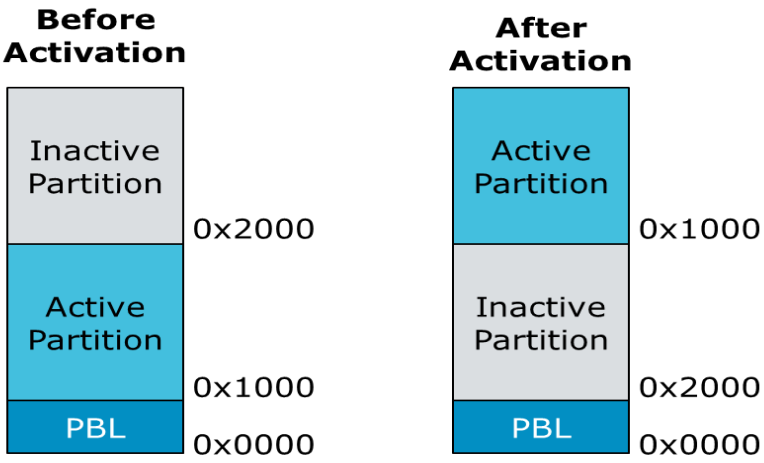
6.2.1.1 Hardware Variants

6.2.1.1.1 FRD-REQ-308073/A-###R_CMP_IVSU_V_00035### Hardware Variant Review

Each component can evaluate the hardware variants and choose the one that fits best in their overall system architecture. However, if a new variant is introduced than it shall be reviewed with CVS IVSU Team for approval and addition in the approved list.

6.2.1.1.2 OTA Architecture Type 1 – Hardware Facilitated Address Remapping

With this approach, activation of a partition involves remapping the active and inactive memory address spaces. This is normally achieved in hardware through the writing of a register or user configuration block.



High Level Requirements:

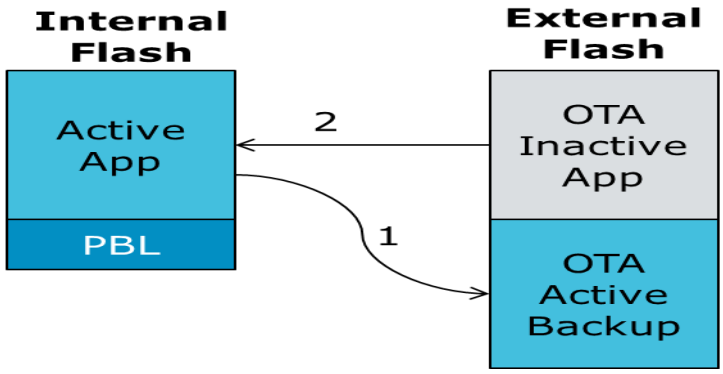
- Hardware assisted memory remapping
- 2x internal flash to support storage of both A & B memory
- Read-while-write capability to internal flash

General flow comments

In initiateActivation, the ECU shall execute the necessary actions (example: writes register/UCB) to perform the memory remapping and resets. This assumes the SWash calculation provided in the authorizeActivation request already verified is still valid .

6.2.1.1.3 OTA Architecture Type 2 –Memory Caching Option 1

With this approach, the new software is downloaded in the background into an allocated external memory area. Prior to activation of the new software, the currently active application is backed up into external memory and the new software is then copied into the active internal memory by the bootloader.



High Level Requirements:

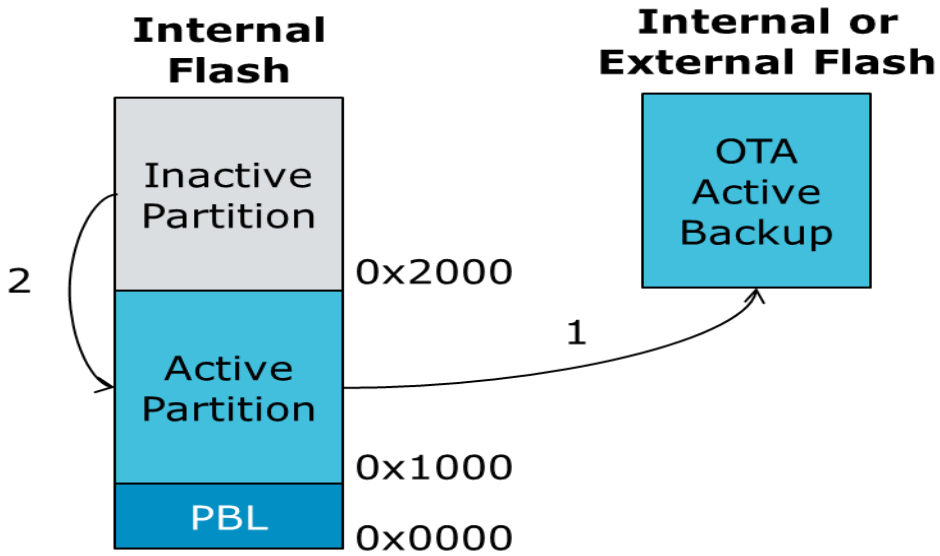
- 2x external flash to support storage of both A & B memory

General flow comments

1. Prepare for activation – The ECU will erase external flash and copy active application into external flash (in case it is needed for rollback). In case of failure during activation, the ECU shall be able to rollback using the OTA Active Backup copy automatically without the need for rollBack FID.
2. Perform activation and reset – The ECU will erase internal flash and copy the new software from external flash into internal flash. This assumes the SWash calculations match both in the OTA Inactive Map prior to beginning the erase and copy, and also the SWash calculations match in the Active App after copying prior to activation.

6.2.1.1.4 OTA Architecture Type 3 – Memory Caching Option 2

With this approach, the new software is downloaded in the background into an allocated internal memory area. Prior to activation of the new software, the currently active application is backed up into a dedicated backup location in either internal or external memory and the new software is then copied from the inactive internal partition to the active internal partition by the bootloader. The position independent code issue is addressed since the software is always running from the same memory address.



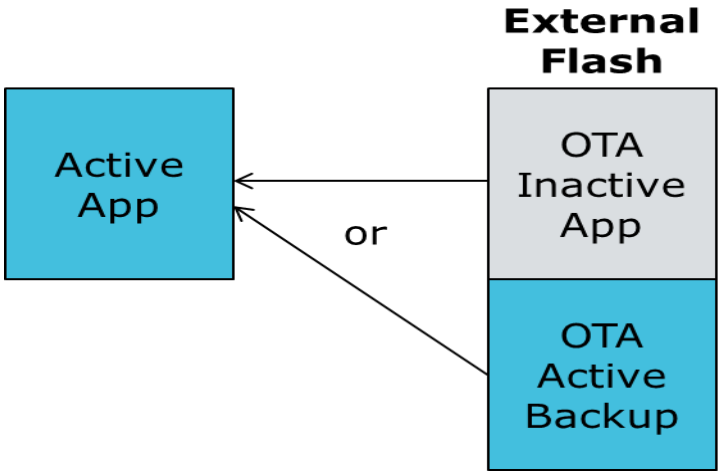
High Level Requirements

- 3x memory to support storage of both A & B memory along with backup
- Read-while-write capability to internal flash
- down time required to copy the internal memory to internal

General flow comments
1. Prepare for activation – The ECU will erase the backup memory area and copy active application into this area in case of rollback
2. Perform activation and reset – The ECU will erase internal flash and copy the new software from the inactive partition of internal flash to the active partition of internal flash. This assumes the SWash calculations match both in the Inactive Partition prior to beginning the erase and copy.

6.2.1.1.5 OTA Architecture Type 4 – Execute from RAM

With this approach, the software is compiled to run from a fixed location in RAM. On startup, a lookup table is used to determine which partition is copied into RAM. The position independent code issue is addressed since the software is always running from the same memory address (in RAM).



High Level Requirements

- 2x memory to support storage of both A & B memory along with backup
- Sufficient RAM to execute the application
- on microcontrollers with sufficient RAM, but often only a viable option for system on a chip configurations

General flow comments
1. Perform activation and reset – The ECU will update the lookup table and resets. This assumes the SWash calculation provided in the activation request matches prior to updating lookup table and resetting.

6.2.1.2 Component

6.2.2 Requirements on Electrical Distribution System (EDS)

6.2.2.1 FRD-REQ-308067/B-###R_CMP_IVSU_V_00055### Electrical Load Architecture

Current and future Vehicle architectures shall provide options to reduce current draw for various OTA activities
(For example: Background programming during Key Off, File download from cloud during Key off)
For example: In FNV2 architecture, If ECG downloads file from cloud via TCU, only HS4 shall kept awake for this purpose.

6.2.2.2 FRD-REQ-308070/B-###R_CMP_IVSU_V_00058### Programming NON A/B PAAT ECU on Key OFF State with Run/Start bus Active

To background download to a non-powered at all-time ECU during Key OFF, the OTA client shall request the Run/Start bus to be active (Conditions shall be met first for the Run/Start bus).

6.2.2.3 FRD-REQ-308072/B-###R_CMP_IVSU_V_00060### ECU Capable of Downloading from cloud shall be awake for certain time period as per ECG request

The ECU that can download from the cloud shall be able to stay awake local for a configurable time to perform the download activity. The OTA Manager shall request the ECU to download in that mode when needed.

6.2.2.4 FRD-REQ-328062/B-###R_CMP_IVSU_V_00062### ECU that requires learning algorithm for specific process or action after an update

Any ECU that requires a learning algorithm or a specific process or action after an OTA update, shall be able to do so without any customer intervention.

6.2.2.5 FRD-REQ-362586/B-TCU shall track SOC in vehicle transport mode

Power Manager to keep track of the Battery SOC when in transitioned and during Transport mode and log timestamp (to be used for wake up time cycle), TCU wake up timer will stop once vehicle mode exit from transport mode.

6.2.2.6 FRD-REQ-362587/B-TCU to handle OTA SMS in vehicle transport mode

TCU shall be able to listen to SMS and to handle SMS for OTA update in Transport Mode if the battery SoC allows

6.2.2.7 FRD-REQ-362589/B-TCU to wake up ECG when there is a wake SMS from cloud in vehicle transport mode

TCU shall wake up ECG (HS4) if there is a wake up SMS form cloud.

6.2.2.8 FRD-REQ-365739/B-TCU/ECG shall send vehicle mode change information to cloud regardless of ignition state

ECG shall send vehicle mode change information (enter Factory Mode/Transport Mode/Normal Mode etc.,) to cloud regardless of ignition state.

6.2.2.9 REQ-365775/B-TCU shall sync SOC from BCM periodically to get accurate SOC

Power Manager shall sync SOC from BCM periodically and the period to sync shall be a configurable parameter.

6.2.2.10 FRD-REQ-394957/A-Routine to Cancel an Active OTA Update

The purpose of this Routine is to cancel an active OTA update (triggered) that is not yet in progress (has not yet been activated on any ECU or has begun a SWDL update of any ECU).

Request Parameters:

Name	Type	Size	Values	Repeatable
OTA Event Update ID	ASCII	36 bytes	Identifier of an OTA Event Update IDs to cancel	Yes: 0..N

This a list of OTA Event Update IDs to cancel. If no OTA Event Update IDs are given, then ***all active*** OTA Event Update IDs will be cancelled.

Response Parameters:

Name	Packeted	Type	Size	Values	Repeatable
OTA Cancel Result	Cancel Result	State Encoded	1 byte	00 = Success 01 = Invalid OTA Event Update ID 02 = Unable to Cancel 03 = No OTA Event IDs Active or In Progress	Yes: 1..N
	OTA Event Update ID	ASCII	36 bytes	OTA Event Update ID	

This is a list of all OTA Event Update IDs that were requested to be cancelled and the result for each of them.

- If the request was to cancel all OTA Event Update IDs (no OTA Event Update IDs given in the request) then all Active or in-progress OTA Event Update IDs are listed with their corresponding cancel result.
- If the request was to cancel all OTA Event Update IDs and there were none active or are in progress, then there shall be only one OTA Cancel Result. The OTA Event Update ID shall be zero and the Cancel Result shall be: 03 = OTA Event Update ID Active or in progress.
- If a particular OTA Event Update ID is repeated in the Routine Request (normally should not be done) then there will only be one corresponding OTA Event Update ID in the OTA Cancel Result.

Examples

Scenario: OTA Event Update IDs 1, 2, 3, and 4. 1, 2, 3 and are active and 4 is in progress

Example 1: Request to cancel all, with above scenario

- Request Parameters: <empty>
- Response parameters -- OTA Cancel Result List
  - OTA Event Update ID: 1
  - Cancel Result: 00
  - OTA Event Update ID: 2
  - Cancel Result: 00
  - OTA Event Update ID: 3
  - Cancel Result: 00
  - OTA Event Update ID: 4
  - Cancel Result: 02

Example 2: Request to cancel a specified list with an invalid OTA Event Update, with above scenario

- Request Parameters: OTA Event Update IDs 1, 3, 4, 5
- Response parameters -- OTA Cancel Result List
  - OTA Event Update ID: 1
  - Cancel Result: 00
  - OTA Event Update ID: 3
  - Cancel Result: 00
  - OTA Event Update ID: 4
  - Cancel Result: 02

- OTA Event Update ID: 5
- Cancel Result: 01

Example 3: Request to cancel all, with no OTA Event Update IDs active or in progress

- Request Parameters: <empty
- Response parameters -- OTA Cancel Result List
  - OTA Event Update ID: 0
  - Cancel Result: 03

6.2.2.11 FRD-REQ-394958/A-Reading DIDs in Application Default and Diagnostic sessions

All the DIDs specified in this Spec shall able to read in Application Default and extended diagnostic sessions for MMOTA needs.

6.2.2.12 FRD-REQ-394960/A-Update\Flash All ECUs regardless of Vehcile Mode

All ECU shall be updateable\flash able regardless of the vehicle mode (Factory, Transport, In plant, Normal, etc.,).

6.2.3 Requirements on DTC and DIDs

Note: Following list provides consolidated list. For details, refer to OTA server and Client documents

Name	ID	Description	Number of Bytes	Sour
In Progress OTA Download Address	D022	Address of next byte to program	5 Bytes	OVTP C Server
OTA Activation Preconditions	D026	Precondition Status/Ignition OFF Time	2 Bytes	OVTP C Server
OTA Over OVTP Support Level	D029	Allows OTA Client to verify OTA spec version	2 Bytes	OVTP C Server
OTA Software Update Counter	D02B	OTA client to verify ECU's value	4 Bytes	OVTP C Server
OTA Debug Information	D03B	OTA Update debug information	24 Bytes	OVTP C Server
OTA Partition Status	D039	The intent of this DID is to provide information on the ECU partition and status. A/B ECUs would not support bits in Byte 3 (as they don't have a 3rd partition). Therefore, a non-zero value in Byte 3 then always indicates this is an "A/B/A architecture", whereas a 0x00 value in byte 3 always indicates it is an "A/B architecture"	3 Bytes	OVTP C Server Applicab AB or AI memorie
In-Use Cloud Authorized Vehicle Control Public Key Hash	D034	Used to report out the public key hash used as part of CAVC	32 Bytes	BCM

Troubleshooting OTA update	DID 0xD042 Most Recent OTA Event Status DID 0xD043 2nd Most Recent OTA Event Status DID 0xD044 3rd Most Recent OTA Event Status DID 0xD045 4th Most Recent OTA Event Status DID 0xD046 5th Most Recent OTA Event Status	This is the most recent OTA Event Status. This DID provide statuses for the different OTA update steps and ECUs. The OTA update steps include trigger status, download status, Data Configuration status, file transfer status, activation status, Rollback status and the overall update status. This DID shall be used for trouble shooting and diagnostic purposes		OVTP C Client
Vehicle Interrogator Log Status	0xD047	This DID provide statuses for the different VIL steps statuses and information. The VIL steps include VIL creation, VIL pause and VIL post. The VIL information includes VIL reason and time VIL is created. This DID shall be used for trouble shooting and diagnostic purposes	18 Bytes	OVTP C Client
Reason for Most Recent OTA Event Non-Activation	0xD032	DID D032 provides reason on why the" most recent OTA event is stuck or waiting for activation and never activated. The DID specifies the reason why a scheduled event for activation did not happen. The possible reason would be if the activation was cancelled by the user, vehicle was in use during the scheduled time and/or certain preconditions were not met. These reasons and preconditions can be vehicle and/or ECU specific. This DID shall be used for trouble shooting and diagnostic purposes.	86 Bytes	OVTP C Client
DTC	U102D (0XD02D)	Incompatible Vehicle Software DTC: 0xD02D57 when vehicle is not inhibited but in Reduce functionality due to Invalid/Incompatible Software Component. DTC:0xD02D53 when Vehicle is Inhibited as not in drivable conditions due to Invalid/Incompatible Software Component.	SAE_J2012-DA_DTCFormat_00	OVTP C Client

7 Open Concerns

ID	Concern Description	e-Tracker Reference	Status	Solution

Table 15: Open Concerns

8 Verification Review

Completed appropriately		Yes / No
Input from System Design, Item Definition / Feature Document, and Functional Safety Concept *(GPDS: UNV0/UPV0, GTDS: <AR>)*	External Interfaces	
	Constraints	
	Technical Block Diagram	
	Functional Overview of Components/Subsystems	
	Implementation Details of Internal Interfaces	
	System Level architecture (including redundancy)	
Technical Safety Requirements Specification Technical Safety Requirements Derivation	Derivation of Technical Safety Requirements (without V&V acceptance criteria) *(GPDS: UNV0/UPV0, GTDS: <AR>)*	
	Definitionof Technical Safety Requirements V&V acceptance criteria *(GPDS: UNV1/UPV1)*	
	Derivation of Fault Tolerant Time *(GPDS: UNV0/UPV0, GTDS: <AR>)*	
	Derivation of Reduced Functionality (interval) *(GPDS: UNV0/UPV0, GTDS: <AR>)*	
	Each Technical Safety Requirement • contains all required attributes (except "V&V acceptance criteria") *(GPDS: UNV0/UPV0, GTDS: <AR>)*	
	Each Technical Safety Requirement • is simple, atomic, verifiable, necessary, achievable, and traceable *(GPDS: UNV0/UPV0, GTDS: <AR>)*	
	Each Technical Safety Requirement • is accepted by the component/subsystem provider *(GPDS: UNV0/UPV0, GTDS: <AR>)*	
	Constraints are transformed into requirements *(GPDS: UNV0/UPV0, GTDS: <AR>)*	
	HW Metric Requirements - Derivation and Rationale • the metric values assigned to the components fulfil the Safety Goal metric requirements. *(GPDS: UNV0/UPV0, GTDS: <AR>)*	
	ASIL Decomposition (Optional) *(GPDS: UNV0/UPV0, GTDS: <AR>)*	
	Safety Related Parameters *(GPDS: UNV0/UPV0, GTDS: <AR>)*	
	Requirements concerning the ability to configure a system by calibration data are defined *(GPDS: UNV0/UPV0, GTDS: <AR>)*	

	Each Technical Safety Requirement can be verified *(GPDS: UNV0/UPV0, GTDS: <AR>)*	
	The Technical Safety Requirements are consistent and complete regarding the System Design, including "Response to Stimuli". *(GPDS: UNV0/UPV0, GTDS: <AR>)*	
	For all categories (Safety Related Function, Internal Fault Handling, External Fault Handling, Latent Fault Handling, Reduced Functionality, User Information, Maintain Safe State / Recovery, General Requirement, Decomposition Requirement) Technical Safety Requirements are derived if relevant. *(GPDS: UNV0/UPV0, GTDS: <AR>)*	
	Technical Safety Requirements necessary for the achievement of the Functional Safety Requirement are generated and documented. *(GPDS: UNV0/UPV0, GTDS: <AR>)*	
Description of other functions of the system *(GPDS: UNV0/UPV0, GTDS: <AR>)*		
System Design *(GPDS: UNV0/UPV0, GTDS: <AR>)*	Technical Safety Requirements included in the system design specification(s). Aligned with Technical Safety Requirements System Design developed in accordance with requirements related to: • System architectural design constraints • Avoidance of systematic faults • Usage of well-trusted design principles • Measures for control of random hardware failures during operation • Allocation to hardware and software • Hardware-Software Interface Specification *(see guideline for "FFSD 04 Safety Requirements Specification")*	
Requirements for Operation, Service and Decommissioning *(GPDS: UNV0/UPV0, GTDS: <AR>)*	Requirements for Operation and Service completed	
Technical Safety Requirements on Components/Subsystems *(GPDS: UNV0/UPV0, GTDS: <AR>)*	V&V acceptance criteria	

9 Revision History

Rev. (revision)	Vers.	Description	Approved by	Responsible
9/15/17	1.0	Requirements. This is a draft version of the specification.		
11/15/17	1.1	1- Updated the numbering and few clarifcations in the Input requirement section 2- updated the logical function diagram to show the functions for starting and inhibitnig the start of the vehicle and the display		
12/14/2017	1.2	1- Flowchart for update over USB and OTA 2- Added fucntions tht are common between ECG and SYNC related to OTA Manager		
04/13/2018	2.0	1- Modified Requirement R_CMP_IVSU_V_00002 DIDs for OTA Command Signing Keys and Application Signing Keys 2- Modified R_CMP_IVSU_V_00003 to support Differential Updater for A/B or ABA methods. 3- Added below input requirements for ECG to perform OTA Activity and OTA Run/Start request. 1. R_CMP_IVSU_V_00015 2. R_CMP_IVSU_V_00016 3. R_CMP_IVSU_V_00017 4. R_CMP_IVSU_V_00018 5. R_CMP_IVSU_V_00019 6. R_CMP_IVSU_V_00020 7. R_CMP_IVSU_V_00021 4- Added Below Electrical Distribution System Requirements for Target ECU's to perform OTA Activity. 1. R_CMP_IVSU_V_00056 2. R_CMP_IVSU_V_00057 3. R_CMP_IVSU_V_00058 4. R_CMP_IVSU_V_00060 5- Added R_CMP_IVSU_V_00059 for Network Availability for OTA Activity 6- Updated Figure 1 Functional Architecture to match with latest OTA Architecture. 7- Updated Figure 2: E/E Architecture, to match with latest OTA Architecture 8- Updated the Function List Section 3.1 as per new Functional Architecture. 9- Updated the Section 4.1.1.3 Function Allocation to respective Modules 10- Added Section 4.1.1.4 Signal / Parameter Mapping table. 11- Added Section 5.1.5 to 5.1.11 various Scenarios for OTA update procedure for Sync, TCU, ECG and all Target ECU's with single micro or two micros. 12- Added Section 11.1 ECG DID's		

Rev. (revision)	Vers.	Description	Approved by	Responsible
		13- Added 11.2.5 Implementation Guide Report. 14- updated Section 1.5 References 15- Updated Section 1.3.1 with Stack Holder list. 16- Updated Section 7 Open Concerns: deleted the earlier concerns which are not valid anymore.		
07/31/2018	*2.1.0*	Updated 1.6.2 Abbrivations: Updated FESN description and added DID. Added R_CMP_IVSU_V_00022 DID for Entering in to OTA ProgrammingSession Added R_CMP_IVSU_V_00061 User start Vehicle during OTA Vehicle Inhibit Added R_CMP_IVSU_V_00062 ECU that requires learning algorithm for specific process or action after an update Modified R_CMP_IVSU_V_00025 for Capacitance Requirement Availability in case of Power OFF While OTA Update		
8/30/2018	*2.2.0*	1. Updated the sequences 5.1.11 5.1.12 all the scenarios as per latest reference to latest PCM sequences. 2. Added DC configuration sequences 5.1.13 to 5.1.16 3. Updated section 4.1.1.4 Signal/Parameter Mapping table as per new CAVC signals.		
8/31/2018	*2.2.1*	1. Updated section 4.1.1.4 Signal/Parameter Mapping table with ECG <--> BCM signals. 2. Added 5.1.11 OTA OTA On Demand Request 3. Updated 3. Functional Architecture 4. Updated 4.1 E/E Architecture Variant 1		
9/4/2018	*2.2.2*	1. Updated section 4.1.1.4 Signal/Parameter with HMI and USB signals. 2. Updated Reference Documents		
9/11/2018	2.2.3	1. Updated 4.1.3 Function Allocation		
1/7/2019	2.2.4	1 Updated section 4.1.1.4 Signal/Parameter with HMI signals.		
1/8/2019	3.0	Released v1.7 in the VSEM plus removed the all reference to RE template and unused sections		
4/8/2019	4.0	Deleted R_CMP_IVSU_V_00019 Deleted R_CMP_IVSU_V_0001, R_CMP_IVSU_V_00014, R_CMP_IVSU_V_00015, R_CMP_IVSU_V_00024, R_CMP_IVSU_V_00056, R_CMP_IVSU_V_00057, R_CMP_IVSU_V_00059, R_CMP_IVSU_V_00061		

Rev. (revision)	Vers.	Description	Approved by	Responsible
		Deleted: DC Configuration Scenarios “Add New Feature Content Over The Air”, “Perform Initial Configuration Over The Air”, “Restore And Replace Electronic Module”. Updated DC Configuration Scenario: “Change Parameter Over The Air” Updated: R_CMP_IVSU_V_00002 to R_CMP_IVSU_V_00013, FRD-REQ-308060, FRD-REQ-308061, FRD-REQ-308062, FRD-REQ-308065, FRD-REQ-324142, FRD-REQ-348263, FRD-REQ-308756, FRD-REQ-308073, FRD-REQ-308067, FRD-REQ-308070, FRD-REQ-308072, FRD-REQ-328062 Updated All the Scenarios from 5.1.1 to 5.1.10. Updated section 4.1.1.4 Signal/Parameter with HMI signals.		
8/28/2019	4.1	Added TCU Requirement: FRD-REQ-362586/A, FRD-REQ-362587/A, FRD-REQ-362588/A, FRD-REQ-362589/A, FRD-REQ-362590/A, FRD-REQ-362591/A		
3/9/2020	4.2	Updated the DO39 and D034 DIDs in the Req 533276 Requirements on DTC and DIDs		
7/21/2020	5.0	Added the requirements FRD-REQ-394957/A, FRD-REQ-394958/A, FRD-REQ-394960/A Updated 6.2.3 with DIDs D042 to D046, D047, D032 and DTC U102D. Added new signals from OTAM to HMI LS_OTAM_Precondition_unknown_Error LS_OTAM_HMI_Remote_Consent LS_OTAM_HMI_Remote_Notification LS_OTAM_HMI_Schedule_Changes LS_OTAM_HMI_ClearHMIsPrompts LS_OTAM_ConenctionType_WiFi		

10 Appendix

10.1 ECG DID's

DID Number (Hex)	Parameter Number	Parameter Name	Size (Bits)	State (Hex)	State Name
D03B	11	4th Most Recent OTA FID Response Type	8	000000	positiveResponse
D03B	11	4th Most Recent OTA FID Response Type	8	000010	generalReject
D03B	11	4th Most Recent OTA FID Response Type	8	000011	functionNotSupported
D03B	11	4th Most Recent OTA FID Response Type	8	000013	incorrectMessageLengthOrInvalidFormat
D03B	11	4th Most Recent OTA FID Response Type	8	000014	responseTooLong
D03B	11	4th Most Recent OTA FID Response Type	8	000015	endToEndSignatureInvalid
D03B	11	4th Most Recent OTA FID Response Type	8	000016	ESNInvalid
D03B	11	4th Most Recent OTA FID Response Type	8	000017	softwareUpdateCounterInvalid
D03B	11	4th Most Recent OTA FID Response Type	8	000021	busyRepeatRequest
D03B	11	4th Most Recent OTA FID Response Type	8	000022	conditionsNotCorrect
D03B	11	4th Most Recent OTA FID Response Type	8	000024	requestSequenceError
D03B	11	4th Most Recent OTA FID Response Type	8	000031	requestOutOfRange
D03B	11	4th Most Recent OTA FID Response Type	8	000033	securityRequired
D03B	11	4th Most Recent OTA FID Response Type	8	000070	downloadNotAccepted
D03B	11	4th Most Recent OTA FID Response Type	8	000071	transferDataSuspended
D03B	11	4th Most Recent OTA FID Response Type	8	000072	generalProgrammingFailure
D03B	11	4th Most Recent OTA FID Response Type	8	000073	wrongSequenceCounter
D03B	11	4th Most Recent OTA FID Response Type	8	000078	requestCorrectlyReceived-ResponsePending

D03B	11	4th Most Recent OTA FID Response Type	8	000079	validationFailed
D03B	11	4th Most Recent OTA FID Response Type	8	00007D	sessionMismatch
D03B	11	4th Most Recent OTA FID Response Type	8	00007F	noActiveSession
D03B	2	Most Recent OTA FID Response Type	8	000000	positiveResponse
D03B	2	Most Recent OTA FID Response Type	8	000010	generalReject
D03B	2	Most Recent OTA FID Response Type	8	000011	functionNotSupported
D03B	2	Most Recent OTA FID Response Type	8	000013	incorrectMessageLengthOrInvalidFormat
D03B	2	Most Recent OTA FID Response Type	8	000014	responseTooLong
D03B	2	Most Recent OTA FID Response Type	8	000015	endToEndSignatureInvalid
D03B	2	Most Recent OTA FID Response Type	8	000016	ESNInvalid
D03B	2	Most Recent OTA FID Response Type	8	000017	softwareUpdateCounterInvalid
D03B	2	Most Recent OTA FID Response Type	8	000021	busyRepeatRequest
D03B	2	Most Recent OTA FID Response Type	8	000022	conditionsNotCorrect
D03B	2	Most Recent OTA FID Response Type	8	000024	requestSequenceError
D03B	2	Most Recent OTA FID Response Type	8	000031	requestOutOfRange
D03B	2	Most Recent OTA FID Response Type	8	000033	securityRequired
D03B	2	Most Recent OTA FID Response Type	8	000070	downloadNotAccepted
D03B	2	Most Recent OTA FID Response Type	8	000071	transferDataSuspended
D03B	2	Most Recent OTA FID Response Type	8	000072	generalProgrammingFailure
D03B	2	Most Recent OTA FID Response Type	8	000073	wrongSequenceCounter
D03B	2	Most Recent OTA FID Response Type	8	000078	requestCorrectlyReceived-ResponsePending
D03B	2	Most Recent OTA FID Response Type	8	000079	validationFailed
D03B	2	Most Recent OTA FID Response Type	8	00007D	sessionMismatch
D03B	2	Most Recent OTA FID Response Type	8	00007F	noActiveSession
D03B	5	2nd Most Recent OTA FID Response Type	8	000000	positiveResponse
D03B	5	2nd Most Recent OTA FID Response Type	8	000010	generalReject
D03B	5	2nd Most Recent OTA FID Response Type	8	000011	functionNotSupported

D03B	5	2nd Most Recent OTA FID Response Type	8	000013	incorrectMessageLengthOrInvalidFormat
D03B	5	2nd Most Recent OTA FID Response Type	8	000014	responseTooLong
D03B	5	2nd Most Recent OTA FID Response Type	8	000015	endToEndSignatureInvalid
D03B	5	2nd Most Recent OTA FID Response Type	8	000016	ESNInvalid
D03B	5	2nd Most Recent OTA FID Response Type	8	000017	softwareUpdateCounterInvalid
D03B	5	2nd Most Recent OTA FID Response Type	8	000021	busyRepeatRequest
D03B	5	2nd Most Recent OTA FID Response Type	8	000022	conditionsNotCorrect
D03B	5	2nd Most Recent OTA FID Response Type	8	000024	requestSequenceError
D03B	5	2nd Most Recent OTA FID Response Type	8	000031	requestOutOfRange
D03B	5	2nd Most Recent OTA FID Response Type	8	000033	securityRequired
D03B	5	2nd Most Recent OTA FID Response Type	8	000070	downloadNotAccepted
D03B	5	2nd Most Recent OTA FID Response Type	8	000071	transferDataSuspended
D03B	5	2nd Most Recent OTA FID Response Type	8	000072	generalProgrammingFailure
D03B	5	2nd Most Recent OTA FID Response Type	8	000073	wrongSequenceCounter
D03B	5	2nd Most Recent OTA FID Response Type	8	000078	requestCorrectlyReceived-ResponsePending
D03B	5	2nd Most Recent OTA FID Response Type	8	000079	validationFailed
D03B	5	2nd Most Recent OTA FID Response Type	8	00007D	sessionMismatch
D03B	5	2nd Most Recent OTA FID Response Type	8	00007F	noActiveSession
D03B	8	3rd Most Recent OTA FID Response Type	8	000000	positiveResponse
D03B	8	3rd Most Recent OTA FID Response Type	8	000010	generalReject
D03B	8	3rd Most Recent OTA FID Response Type	8	000011	functionNotSupported

D03B	8	3rd Most Recent OTA FID Response Type	8	000013	incorrectMessageLengthOrInvalidFormat
D03B	8	3rd Most Recent OTA FID Response Type	8	000014	responseTooLong
D03B	8	3rd Most Recent OTA FID Response Type	8	000015	endToEndSignatureInvalid
D03B	8	3rd Most Recent OTA FID Response Type	8	000016	ESNInvalid
D03B	8	3rd Most Recent OTA FID Response Type	8	000017	softwareUpdateCounterInvalid
D03B	8	3rd Most Recent OTA FID Response Type	8	000021	busyRepeatRequest
D03B	8	3rd Most Recent OTA FID Response Type	8	000022	conditionsNotCorrect
D03B	8	3rd Most Recent OTA FID Response Type	8	000024	requestSequenceError
D03B	8	3rd Most Recent OTA FID Response Type	8	000031	requestOutOfRange
D03B	8	3rd Most Recent OTA FID Response Type	8	000033	securityRequired
D03B	8	3rd Most Recent OTA FID Response Type	8	000070	downloadNotAccepted
D03B	8	3rd Most Recent OTA FID Response Type	8	000071	transferDataSuspended
D03B	8	3rd Most Recent OTA FID Response Type	8	000072	generalProgrammingFailure
D03B	8	3rd Most Recent OTA FID Response Type	8	000073	wrongSequenceCounter
D03B	8	3rd Most Recent OTA FID Response Type	8	000078	requestCorrectlyReceived-ResponsePending
D03B	8	3rd Most Recent OTA FID Response Type	8	000079	validationFailed
D03B	8	3rd Most Recent OTA FID Response Type	8	00007D	sessionMismatch
D03B	8	3rd Most Recent OTA FID Response Type	8	00007F	noActiveSession
D03C	1	Campaign #1 Source	8	0000000000000000	OTA
D03C	1	Campaign #1 Source	8	0000000000000001	USB
D03C	1	Campaign #1 Source	8	0000000000000002	Vehicle
D03C	4	Campaign #2 Source	8	0000000000000000	OTA
D03C	4	Campaign #2 Source	8	0000000000000001	USB
D03C	4	Campaign #2 Source	8	0000000000000002	Vehicle

D03C	7	Campaign #3 Source	8	0000000000000000	OTA
D03C	7	Campaign #3 Source	8	0000000000000001	USB
D03C	7	Campaign #3 Source	8	0000000000000002	Vehicle
D03C	10	Campaign #4 Source	8	0000000000000000	OTA
D03C	10	Campaign #4 Source	8	0000000000000001	USB
D03C	10	Campaign #4 Source	8	0000000000000002	Vehicle

10.2 Data Dictionary

10.2.1 Logical Signals

#Macro: Add Ins -> Add Requirement macro (select “Logical Signal” as type)

10.2.2 Logical Parameters

#Macro: Add Ins -> Add Requirement macro (select “Logical Parameter” as type)

10.2.3 Technical Signals

#Macro: Add Ins -> Add Requirement macro (select “Technical Signal” as type)
#Hint:This section lists all GSDB + GDT + SW signals relevant for the feature deployment. Additionally to the he basic attributes, it shall capture the detailed requirements of a signal, such as:

10.2.4 Technical Parameters

#Macro: Add Ins -> Add Requirement macro (select “Technical Parameter” as type)
#Hint: This section lists all Method 2, Method 3 and calibration parameters relevant for the feature deployment.

10.2.5 Data Types

Implementation Guide Report										
Requirement ID	**Feature/Function/Requirment/Use Case**	**Comments**	**OVTP ECUs**	**ECG**	**SYNC**	**TCU**	**Erase & Replace Ecus**	**BCM**	**PCM**	**Cluster**
FRD-REQ-308047	DIDs for OTA Command Signing Keys and Application Signing Keys		x							
FRD-REQ-308048	Differential Updater		x	x	x	x				
FRD-REQ-308049	Number of Software Updates		x	x	x	x	x			

FRD-REQ-308050	Temporary Vehicle Storage for Software Files			x	x					
FRD-REQ-308052	Maximum ECU Activation Time		x	x	x	x				
FRD-REQ-308053	Component Hardware Review		x	x	x	x				
FRD-REQ-308054	Downloading in background		x	x	x	x				
FRD-REQ-308055	Software Signing		x	x	x	x	x			
FRD-REQ-308056	Vehicle Inhibit			x				x	x	x
FRD-REQ-308057	Preserve Data		x	x	x	x	x	x	x	x
FRD-REQ-308058	Configuration Data									
FRD-REQ-308060	ECUs that can download files from Cloud/USB shall be capable to have local wake up/stay awake			x	x					
FRD-REQ-308061	OTA Client shall not request the OTA Run/Start active if ignition_status <> Off			x						
FRD-REQ-308062	OTA Client shall NOT start any OTA Activity if it receives a load shedding signal.			x						
FRD-REQ-308065	OTA Client shall NOT initiate or process any OTA activity when Battery is in critical condition			x						
FRD-REQ-324142	DID for Entering in to OTA ProgrammingSession		x	x	x	x	x	x	x	x
FRD-REQ-348263	Self Install ECU during Load shed			x	x	x				
FRD-REQ-308756	Capacitance Requirement Availability in case of Power Off While OTA Update		x	x	x	x				
FRD-REQ-308073	Hardware Variant Review									
FRD-REQ-308067	Electrical Load Architecture		x	x	x	x				x
FRD-REQ-308070	Programming NON A/B PAAT ECU on Key OFF State with Run/Start Bus Active		x							
FRD-REQ-308072	ECU Capable of Downloading from cloud shall be awake for certain timer period as per ECG request			x	x					
FRD-REQ-328062	ECU that requires learning algorithm for specific process or action after an update		x	x	x	x	x	x	x	x

Table 16: Implementation Guide Report