Function Specification (FncS)

()

Document Type	Function Specification (FncS)	
Document ID	547914	
Document Location	VSEM Rich Client, VSEM Active Workspace	
Document Owner	Munir, Omar (O.) (omunir)	
Document Version	F	
Document Status	Released	
Date Issued	28-Jun-2022 15:19	
Date Revised	02-Aug-2022 18:34	
Document Classification	GIS1 Item Number:	
	GIS2 Classification:	

Document Approval			
Person	Role	Email Confirmation	Date

CONTENTS

List of Figures

List of Tables

1 REVISION HISTORY

Revision	Date [YYYY-MM-DD]	Description	Editor	Approver
1.0.0	2018-03-15	Initial version	Gill, Balwinder (B.K.)	
1.9.2	2018-08-31	Erase and Replace File download Multiple USB Device Found with Software Update USB Progress bar Invalid Software found on USB, such as old software USB Parser shall not send any errors (files are valid-start update or invalid-do nothing) Added flows to clarify	Gill, Balwinder (B.K.)	
3.5	2021-09-28	Remove the USB VIL upload requirements. Service tool is able to upload the VIL so no need for USB device to update VIL.	Gill, Balwinder (B.K.)	
E	2022-05-24	As a part of the specification clean-up exercise, removed the following 30 Not-Implemented/Obsolete/Invalid requirements: 1. REQ-304794 16. REQ-304767 2. REQ-322904 17. REQ-324909 3. REQ-304763 18. REQ-304776 4. REQ-324899 19. REQ-324910 5. REQ-304765 20. REQ-324911 6. REQ-324902 21. REQ-304780 7. REQ-324903 22. REQ-323043 8. REQ-322744 23. REQ-324914 9. REQ-322912 24. REQ-326797 10. REQ-304768 25. REQ-324915 11. REQ-304769 26. REQ-328365 12. REQ-304770 27. REQ-304786 13. REQ-323044 28. REQ-304796 14. REQ-304771 29. REQ-304784 15. REQ-304766 30. REQ-322905 Updated and re-formatted the list of figures and the list of tables. Modified the following 12 partially-implemented requirements to reflect the current implementation: REQ-304793; Removed references to VIS in the USB package as it is not present and reworded to clarify the requirement REQ-329634; Removed reference to Vehicle Inhibit as USB does not (currently) support updates involving vehicle inhibit REQ-304761; Removed references to VIS, cloud signed commands, and updated the file and folder names REQ-324897; Removed the example and updated the HMI notification comment REQ-324905; Corrected mention of updated being paused to update being cancelled, reworded to clarify the requirement REQ-324906; Corrected mention of updated being paused to update being cancelled, reworded to clarify the requirement REQ-304773; Removed outdated parameter reference	Munir, Omar (O.)	Hofsess, William (W.)

		REQ-304777; Updated HMI reference to REQ-488712 and corrected description based on current implementation **REQ-324917; Removed parameter reference and corrected description** **REQ-304788; Corrected description to reflect current implementation** **REQ-304760; Removed mentioned of VIS and cloud-signed files, reworded to clarify the requirement** **REQ-304764; Removed references to switching between different ports, reworded to clarify the requirement** **Reworded the following 12 requirements for clarification:** **REQ-304795 (also moved to non-functional requirements)** **REQ-322903 (also moved to non-functional requirements)** **REQ-322154** **REQ-304798** **REQ-304783** **REQ-324900** **REQ-322743** **REQ-323046** **REQ-324907** **REQ-326796** **REQ-304772** **REQ-304787** **Updated diagram in the 'Functional Architecture' section by removing the arrow 'Post VIL if connection is available' as this functionality is currently not supported.** **Removed the following outdated sections:** **Stakeholder List** **HMI Functions** **Open Concerns** **Updated 'Figure 2: Function USB Update', 'Table 6: List of Logical Functions', and 'Table 7: Logical Inputs' by removing signals associated with failure reasons, release notes, and cloud posting** **Renamed the 'Audience' section to 'Audience and Reference Documentation' and added VSEM IDs to the Ford document references.**		
F	**2022-08-02**	**Figure 1; Updated to remove "upload VIL" signal between <F1> and <F6>, Removed "Post VIL after Update" signal** **Figure 3; Removed "ODL" reference** **Table 6; Removed <F9> VIL upload** **Table 9; Corrected descriptions for the "battery voltage low", "crash", and "Master Reset" parameters** **REQ-304761, REQ-304760, REQ-324905; Removed references associated with "zipped" files as they are not supported** **Removed REQ-323046, REQ-304772, and REQ-304777 as they are duplicates of requirements REQ-328203, REQ-350940, and REQ-305473 which are present in the OTA Manager specification document (FNS-547911)** **Removed REQ-324906 as it is not implemented and not supported**	**Munir, Omar (O.)**	**Hofsess, William (W.)**

		Removed REQ-304788 as update progress is no longer displayed **Removed REQ-322903 as the Service team is reporting issues when connectivity is not present in the vehicle**		

2 INTRODUCTION

2.1 Purpose

The purpose of the USB Function Specification is to provide the requirements for in vehicle software updates through a USB drive. USB software updates follow a similar process to OTA update except that the binaries are delivered via a USB drive instead of being downloaded from the IVSU cloud. A USB software update can be performed offline as the software package which contains the binaries and manifest files are obtained from the cloud.

2.2 Scope

The scope involved the following functions that are used to update the ECU(s) software through the USB port.

Table 1: Functions described in this specification

Function Name	Owner	Reference
USB Parser	Balwinder Gill	FNS-547914
SFTP	Mirela Fonoage	VDOC082344
Download Manager	Vijay, Jayaraman	FNS-547923
OTA Manager	Mohamad Nasser	FNS-547911

2.3 Audience and Reference Documentation

This Functional Specification describes the interfaces related to the Ford backend, the OTA Software Release, the development website, and the in-vehicle ECU interfaces. Therefore, the target audience are the software developers, testing teams, service teams, and the product managers.

2.3.1 Ford Documents

The following table provides references to the affiliated Ford internal documentation.

Table 2: Ford Documents

#	Title	Doc. ID	Revision
1	OTA Manager	FNS-547911	H
2	Status Manager	FNS-548480	C
3	Trigger Manager	FNS-547921	C
4	VIL Client	FNS-547912	F
5	Download Manager	FNS-547923	B
6	Self-Install Manager	FNS-547922	B
7	Vehicle HMI	FNS-1048650	A
8	Cloud Function Signed Commands	FNS-547916	B
9	OTA Cloud Interface Specification	FNS-1018638	B

2.3.2 External Documents and Publications

Abbreviation	Expansion	Description
FS	Functional Specification	The document describing, collecting and developing the requirements of a function or a group of functions.
MSD	Mass Storage Device	USB Mass Storage Device such as a pen-drive. This includes other external devices such as SD cards.

Table 3: External Documents and Publications

Reference	Document/Publication
TFAT	http://msdn.microsoft.com/en-us/library/aa915463.aspx

2.4 Terminology

2.4.1 Definitions

Table 4: Definitions used in this Documents

Definition	Description
VIL	Vehicle Interrogator Log
ECU	Electronic Control Unit
OTA	Over The Air
GIVIS	Global In-Vehicle Software Information
VIN	Vehicle Identification Number
DTC	Diagnostic Trouble Code
ESN	Electronic Serial Number
HMI	Human Machine Interface/Interaction

2.4.2 Abbreviations

TFAT	Transaction-Safe FAT File System	Transaction-Safe FAT File System allows file modification operations to be interrupted before completion, provides greater protection against power removal and corruption. This is accomplished by storing two copies of the FAT table otherwise it is very similar to FAT32.
exFAT/TexFAT	Extended File system	Extended File system, with extended size and security controls. When the two are combined, it is commonly referred to as TexFAT.
MSA	Memory Stick Audio	Don't index (Memory Stick Audio Format).
FAT32	File Allocation Table 32	Legacy file system for storing and retrieving files from storage devices.
NTFS	New Technology File System	Windows NT Files system for storing and retrieving files from storage devices.
SFTP	Software File Transfer Protocol	Protocol to transfer files between QNX OS.

Table 5: Abbreviations used in this document

3 FUNCTION GROUP DESCRIPTION

3.1 Overview

The purpose of this specification is to provide the requirements for all vehicles that have the capability of updating through a USB port. The contents of the USB update package are dependent on the type of updates received from the Ford backend through the service or engineering website. The USB software update (regardless of the target module for that software update) shall be processed through a module that can be interfaced directly with a USB port.

3.2 Input Requirements

F-REQ-304793/D-###R_FNC_USB_003### Obtain USB Software Files

The Service website shall provide user interface for vehicle level OTA software updates.

For USB software update service, the service website shall have interface to VIN signed manifest.

The user shall be able to download the USB software update from the service website for all applicable regions and countries.

FUR-REQ-322154/B-###R_FNC_USB_007### Software Files Encryption and Decryption

The Ford Security Cloud System shall encrypt the software files that are identified as needing encryption before distributing the files through OTA.

Before transferring the software files to the target ECU, the vehicle client module shall decrypt the encrypted software files.

F-REQ-304798/D-###R_FNC_USB_008### Manifest Signing

All USB manifest file(s) shall be VIN signed by the method identified for each ECU.

Note: Development software shall support generic manifest files which shall not be required to be VIN signed.

3.3 Assumptions & Constraints

This functional specification assume that SYNC or other similar modules have a direct connection with a USB port. This specification is not limited to the SYNC module but also includes other Infotainment Systems with a USB connection to the ECG.

4 FUNCTIONAL ARCHITECTURE

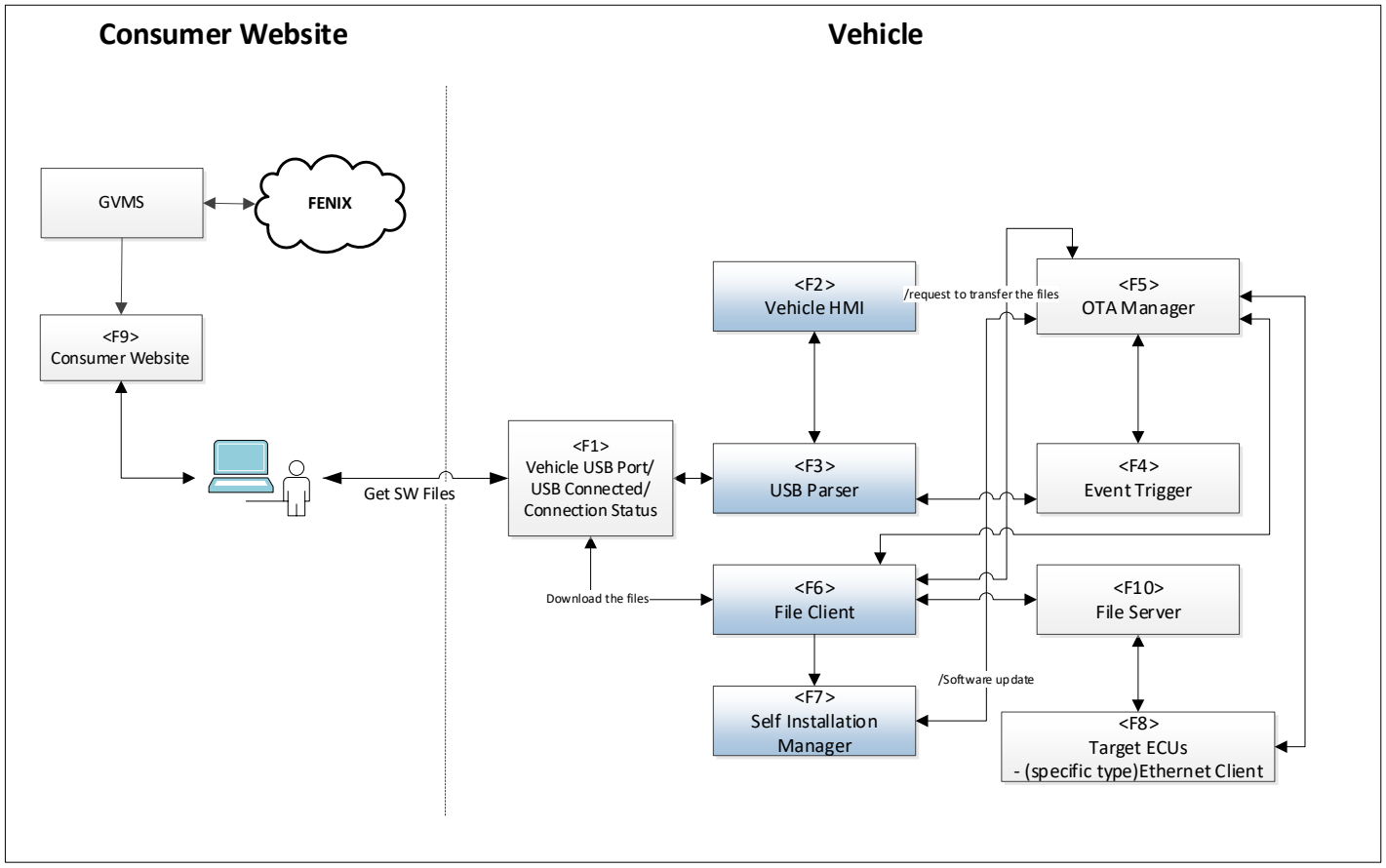

Figure 1: Overall USB Update Function Group

4.1 Function List

4.1.1 List of Logical Functions

Table 6: List of Logical Functions

Function ID	Function Name	Function Description
<F1>	USB connection Status	USB connection shall always monitor USB device status of being plugged or unplugged.
<F2>	Vehicle HMI	In vehicle HMI shall show progress until the software complete or failed.
<F3>	USB Parser	USB Parser shall detect USB device is connected to USB-Port with a valid Ford software update.
<F4>	Event IVSU Trigger	The Event IVSU trigger shall notify OTA manager that USB device is inserted with software updates.

<F5>	OTA Manager	The OTA Manager interface with the vehicle HMI (more details present in the OTA Manager Specification Document).
<F6>	File Client	Upon a request the from file server, the File Client shall transfer the all the requested files.
<F7>	Self-Installation Manager	If USB update is for self-Install module, then it shall wait until it receives all required information from OTA Manager.
<F8>	Target ECUs	The OTA Manager shall install the respective software on target ECUs.
<F10>	File Server	The File server has an interface with the File client to download all the software files onto ECG.

5 LOGICAL FUNCTIONS

5.1 USB Software Update

5.1.1 Function Description

The purpose of this function is to provide all the requirements for USB software updates such as to re-flash the full OS/App, re-flash differential OS/APP or any other software files pertaining to the vehicle components.

5.1.2 Function Scope

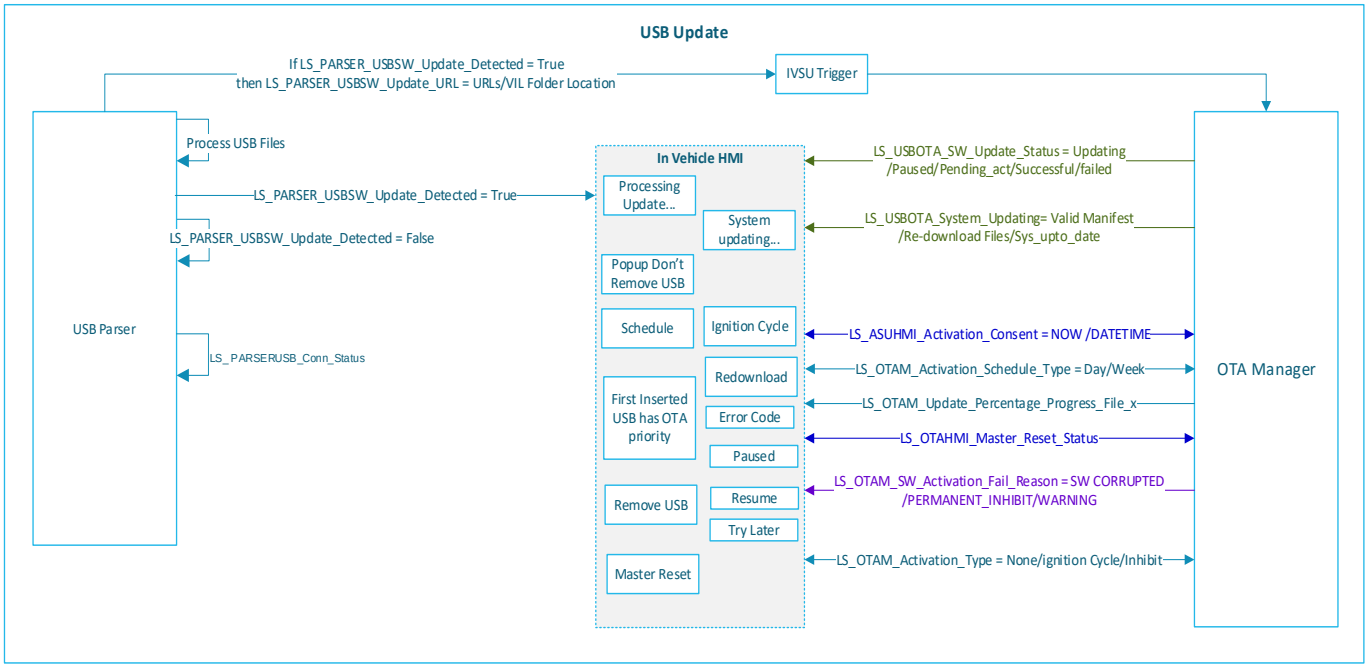

Figure 2: Function USB Update

5.1.3 Function Interfaces

5.1.3.1 Logical Inputs

Table 7: Logical Inputs

Signal Name	Description
LS_USBOTA_System_Updating	Valid Manifest

Signal Name	Description
~~LS_ PARSERUSB_Conn_Status~~	~~Maintain USB connection status (plugged, unplugged, and re-plugged (same or different) USB device.~~
LS_PARSER_USBSW_Update_Detected	The USB Parser shall notify the Event trigger with files types and location of software files and VIL folder. USB parser shall provide an initial flag to HMI when validating the USB for OTA software updates.
LS_OTAHMI_Master_Reset_Status	Master Reset or No Master Reset
	02 – Re-download Files 03 – System update to date
LS_USBOTA_SW_Update_Status	Downloading In Progress Pending for activation Failed Successful Paused Resumed
LS_OTAM_Activation_Type	Activation Type: - No Ignition Cycle Ignition Cycle Inhibit
LS_ASUHMI_Activation_Consent	NOW DATETIME
LS_OTAM_Update_Percentage_OverallProgress	Overall software update progress
LS_OTAM_OTAUSB_Number_of_Files	This signal shall show total and remaining number of files.
LS_OTAM_Activation_Schedule_Type	Day or Week
LS_OTAM_Activation_Time	Progress of activation
LS_OTAM_HMI_OTAUSB_Clear	USB update is paused and USB configuration time expire or OTA Manager abort the update, then set this signal to clear progress bar and other USB related HMIs.

5.1.3.2 Logical Outputs

Table 8: Logical Outputs

5.1.4 Function Modeling

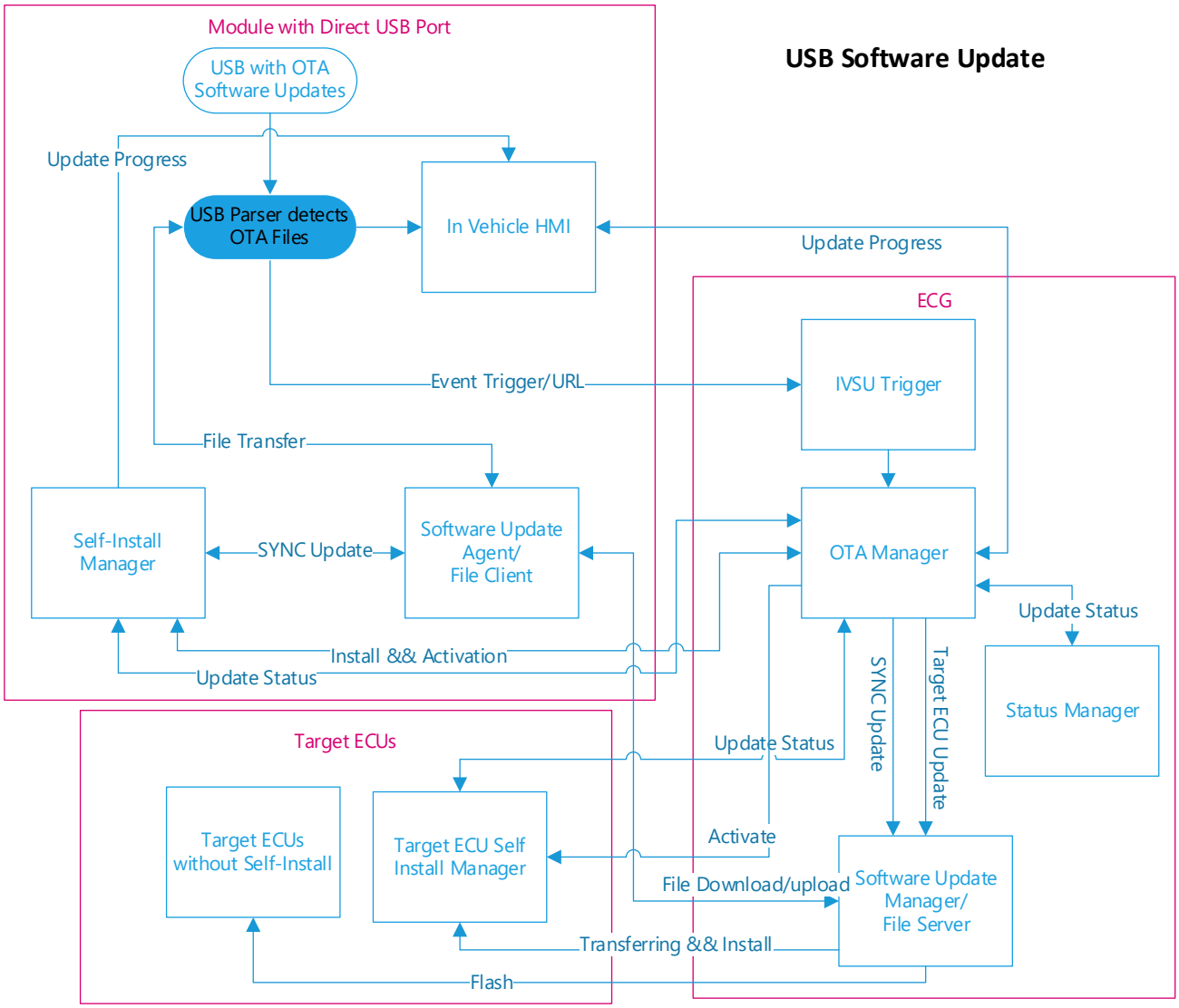

Figure 3: USB Software Update Flow

5.1.5 Function Requirements

5.1.5.1 Functional Requirements

REQ-329634/B-###R_FNC_USB_015### Types of Software Updates

The system shall allow the following types of USB Software updates:

- Software updates with normal ignition cycle
- Software updates without ignition cycle

F-REQ-304783/C-###R_FNC_USB_016### Software Packages

The Software package(s) in the USB device shall contain the downloaded software files that are released in IVS and/or VADR software repositories.

F-REQ-304761/E-###R_FNC_USB_018### USB Parser Detecting Software Update

When a USB device in inserted, the USB parser shall detect if USB device contains Ford release software files.

The USB parser shall detect the software updates by reading the folder and file names.

The USB Parser shall detect the following for USB Software Updates:

1. Correct vehicle level manifest file “VIN_FordSoftwareManifest.der” for that vehicle
2. Correct software folder

When the USB contains a software folder (unzipped), the USB Parser shall detect the “VIN_FordSoftwareUpdates” folder name.

For the consumer update, the following files shall be included inside the zip file “*VIN_FordSoftwareUpdates.zip*”:

- The “VIN_ FordSoftwareManifest.der” file which contains the encrypted manifest
- The “VIN_FordSoftwareUpdates” folder which contains the software binaries

For the service update, the following files shall be included inside the zip file “*FordSoftwareUpdates.zip*”:

- The “VIN_ FordSoftwareManifest.der” file which contains the encrypted manifest
- The “FordSoftwareUpdates” folder which contains the software binaries

FUR-REQ-324897/C-###R_FNC_USB_019### Multiple Manifest or Folders on USB device

When the USB device has multiple manifest files and/or software folders, the USB parser shall select the correct files and/or software folders (with the matching VIN) for the vehicle.

If the USB device contain manifest file(s) but the USB parser is unable to detect a valid manifest with the VIN matching the VIN of the vehicle, the HMI shall then notify the user to remove the files from the USB device and re-download the software package(s).

F-REQ-304760/E-###R_FNC_USB_020### Software Package Validation

The Service website shall provide an interface to download the software packages in advance for all vehicles.

The Service website shall provide an interface to download the VIN signed manifest file.

FUR-REQ-324900/B-###R_FNC_USB_026### Decrypt USB Software Files

If the USB Update Software files are encrypted, then the OTA system shall decrypt the software files before transferring them to target ECU.

FUR-REQ-322743/B-###R_FNC_USB_027### IVSU Authorization for USB Updates

For USB Software update in a vehicle authorization & PII data shall not be required.

FUR-REQ-324905/C-###R_FNC_USB_031### Navigation/Map or Large File Software Update

If the Software to be updated requires more memory than what is available on the ECU, then the ECG shall not download the files into the component cache and instead, directly install the files from the USB device.

During a software update, if the USB if removed, the update shall be cancelled.

F-REQ-304764/C-###R_FNC_USB_033### File transfer over Ignition cycles

The File Transfer shall start and maintain the copy process over unlimited ignition cycles.

When the transfer files request is made by the OTA Manager, all the respective files shall start transferring to the target location.

The USB update shall do a binary data transfer without impairing normal functionality.

FUR-REQ-324907/B-###R_FNC_USB_044### USB Update During Software Activation

While activation of a software package is in progress for an ECU through OTA, when a USB device is detected by the USB Parser, the USB Parser shall ignore the USB device with the software update until the activation is complete.

FUR-REQ-326796/B-###R_FNC_USB_046### eCall during USB software update in progress

While the USB software update is in progress, when an eCall occurs, the system shall pause the USB software update.

The system shall resume the USB software update after the eCall ends.

F-REQ-304773/C-###R_FNC_USB_048### USB connection Status

The USB parser shall keep track of the USB device connection status as plugged or unplugged.

FUR-REQ-324917/B-###R_FNC_USB_065### Software Update is Complete Remove the USB

When the software update is complete, the OTA manager shall trigger a flag to the HMI.

The HMI shall prompt the user to remove the USB device when the software update is completed (when the activation is successful). The HMI update details page shall contain the date and time for when the software update was completed.

5.1.5.2 Normal Operation

The USB software update shall be able to download and install without impairing normal vehicle functionality.

5.1.5.3 Error Handling

The following table shows common errors and expected behavior for the installer.

Table 9: Error Handling

Parameter Name	Description
Cache Full	Module doesn't have enough memory to process the update.
Manifest invalid	Manifest file is corrupted, older manifest file, and missing required files.
Installation package returned error	Fail the installation and reset installation mode.
USB Removed	USB Removed during manifest download and/or software download.
Install Signature	Fail the installation.
Reboot/Reset	Continue the installation session from the last saved point. A checkpoint should be saved for the download progress and after each installation file is processed.
Battery Voltage Low	The download/installation session of the specific file will restart once the vehicle is in full-power mode (saving offsets is not supported).
USB with Media Files or software files	Download in progress and second USB port user inserted Media USB device, files download shall continue.
Vehicle Modes	So long as the vehicle's ignition is ON, the USB update shall continue to download and install.
eCall/Phone	While the e-call in progress, the download/installation shall pause During the phone call, the download/installation shall continue.
Crash	If the vehicle is in crash mode, the software update shall be cancelled.
Ignition Cycle	Download will be cancelled after the is ignition OFF.
Master Reset	The Master reset shall cancel the USB download/installation of the software files (except during an Erase and Replace update process, where the Master Reset shall not be allowed until the activation is completed).

5.1.5.3.1 Master Reset During USB Update

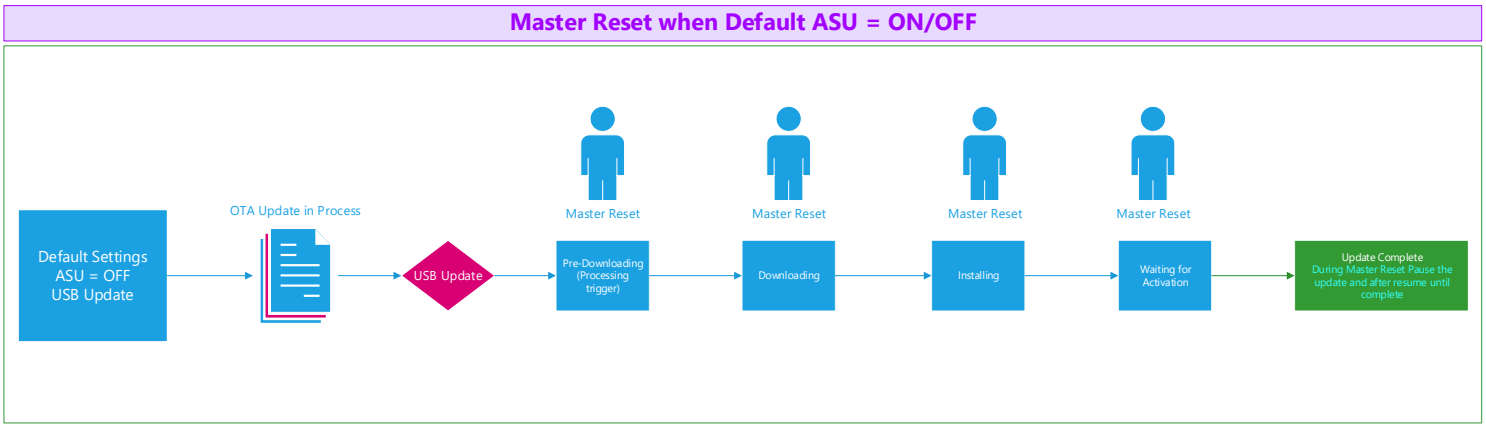

Figure 4: Master Reset during USB Software Update

5.1.5.4 Non-Functional Requirements

FUR-REQ-322903/B-###R_FNC_USB_002### Connectivity Status

The vehicle shall not require any type of internet connection for any USB software updates.

Note: CCS (Customer Connectivity Settings) settings is not an input to USB software update.

F-REQ-304795/C-###R_FNC_USB_001### User Consent

The OTA user consent shall not affect USB software update since the user consent is already provided when user logs in to service website and downloads the latest software.

F-REQ-304787/C-###R_FNC_USB_068### USB Parser Response

When a USB device is plugged in, the USB Parser shall respond within 10 seconds.

F-REQ-304788/C-###R_FNC_USB_069### System Updates progress display

When a system is being updated by a USB device, the user shall be provided an option to view the update progress in the 'Update Details' page of the HMI. The update progress shall be indicated via the number of files remaining e.g., 50/100 files remaining.