core Flight System (cFS) Limit Checker (LC) Application

Requirements Document

Version 1.2

August 25, 2017

1.0 Introduction

1.1 Document Purpose

The core Flight Software System (cFS) Limit Checker (LC) Application has been developed by the Flight Software Branch (FSB) of the Software Engineering Division (SED) at NASA's Goddard Space Flight Center. The purpose of this requirements specification is to define the requirements to be satisfied by the Limit Checker Application. This application has been developed for re-use. For this reason, several nomenclatures are used in this document to identify configurations for a mission.

The cFS is specified as a multi-platform product. Mission-specific features and customization requirements which are applicable for all platforms are tagged with <MISSION_DEFINED>. Platform-specific features and customizations requirements are tagged with either "<PLATFORM_DEFINED>" or "<OPTIONAL>." Additional nomenclature is used along with the tag to specify a cFS default value for the platform-specific feature: "<PLATFORM_DEFINED, Default_Value>". Reference platforms (single processor and multi-processor architectures) are defined to supply the default cFS application configuration. These configurations define the "maximum" cFS Application deployments such that any refined deployment is a subset of a reference platform.

1.2 Document Scope

The scope of this document is limited to the specification of requirements for the Limit Checker Software. These include functional, performance, qualification, and design requirements.

1.3 Document Organization

This document is organized into three additional sections and several appendices.

Section 2 gives the Limit Checker context.

Section 3 documents the Limit Checker system design decisions and constraints.

Section 4 contains the Limit Checker subsystem requirements

Section 5 contains the Limit Checker detailed functional and performance requirements.

Appendix A contains a list of abbreviations and acronyms used in this document.

1.4 Relevant Documents

1.4.1 Parent Documents

cFS Limit Checker Application Heritage Analysis 582-2008-007

1.4.2 Reference Documents

Operating System Abstraction Layer (OSAL) Library cFE Application Developer's Guide 582-2007-001

cFE User's Guide

2.0 cFS File Manager Application Context

In the most basic sense, the Limit Checker application monitors telemetry data points in the flight system and compares the values against predefined or computed threshold limits. When a threshold condition is encountered, an event message is issued and a Relative Time Sequence (RTS) command script may be initiated to respond to the threshold violation.

The definitions of the data to monitor and the actions to be taken are defined in 2 separate tables (see Figure 1.0):

- Watchpoint Definition Table (WDT) – defines the data to be evaluated (message id, offset, comparison value etc.)
- Actionpoint Definition Table (ADT) – defines the equations to use to evaluate the data and the actions to be taken

Figure 1.0 shows the context diagram for the cFS Limit Checker (LC) Application. During initialization, LC subscribes to messages from other applications as defined in the WDT. The Scheduler Application (SCH) sends periodic commands to LC as defined in the SCH Schedule Table. Ground commands come from the Command Ingest task (CI). Messages are routed to LC by the cFE SB Application. LC learns of ground updates to the LC tables through the cFE Table Services application. Actions taken by LC as defined in the ADT are sent to the cFS Stored Command (SC) Application. Messages generated by LC are routed to Housekeeping (HK) and Data Store (DS) (as long as the applications subscribe to them).

LC generates 2 dump-only tables:

- Watchpoint Results Table (WRT)– Contains the results of the Watchpoint evaluations defined in the WDT
- Actionpoint Results Table (ART) – Contains the results of the Actionpoint evaluations defined in the ADT

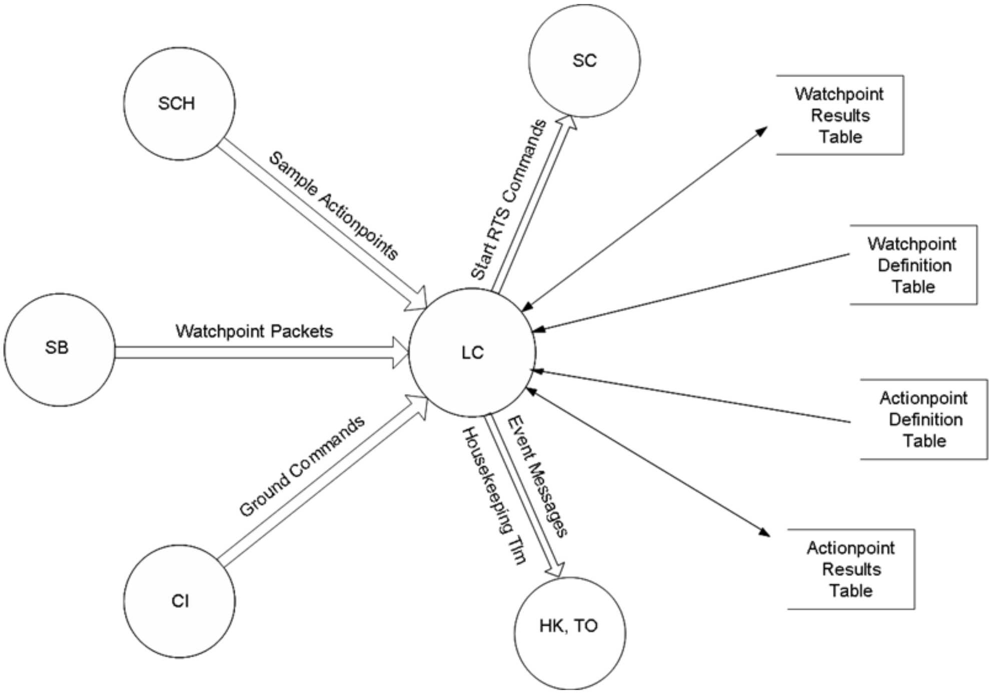

Figure 1.0 – cFS LC Context

2.1 Assumptions

The following list summarizes the assumptions made by the cFS Limit Checker Application:

- cFE API and OSAL are being used
- A command is sent to LC to schedule the execution of the application (from Scheduler Application)

3.0 Design Specifications

The Limit Checker Application's requirements and design are based on the results of the cFS heritage analysis effort and added support for "stale" Watchpoint and Actionpoint results. The results of the heritage analysis are documented in the cFS Limit Checker Application Heritage Analysis document.

- Receipt of messages. LC subscribes to the messages defined in the WDT. Upon receipt of each message, LC processes the messages based on the definitions in the WDT and will reset the age of the Watchpoint back to the limit specified in the WDT.
- Receipt of an LC Sample command. Typically this command is contained in the Scheduler Application table. When LC receives this command, if the Watchpoint Age Update argument is set to TRUE, LC will evaluate the actionpoints as defined in the ADT and decrement the Watchpoint Stale Results Counters. When a Watchpoint Stale Results Counter decrements to zero, LC will update the Watchpoint result for that Watchpoint to be "Stale".

Note that custom functions can be defined if the operators (<, <=, !=, =, >, >=) do not provide enough functionality.

As Actionpoints are evaluated, their results and activities are stored real-time in this dump-only Table.

It is the responsibility of the operator to ensure that changes to the WDT and/or the ADT result in a valid configuration.

Active - Active mode is the normal operation mode. In Active mode, the LC application performs all limit tests defined in the WDT and will invoke stored command sequences if the results of Watchpoint evaluations trigger an Actionpoint condition as defined in the ADT.

Passive - In Passive mode, the LC application behaves just like in Active mode except no stored command sequences will be invoked as a result of Actionpoint triggers.

Disabled - When disabled, the LC application will perform no Watchpoint or Actionpoint limit tests.\

In addition, LC provides the capability to set the state of individual Actionpoints to Active, Passive or Disable. Note that there is no capability to set the state of Watchpoints, only Actionpoints.

3.1 Design Constraints

Since the definition of the Watchpoints and Actionpoints is defined in two separate tables, careful attention needs to be paid to ensure that the tables are consistent with each other.

4.0 Subsystem Requirements

ID	ReqID	Text	Rational	Heritage Reference
7420	CFS-500	The cFS shall provide for on-board monitoring of telemetry points.	Provides for failure detection	LRO, SDO
7422	CFS-501	The cFS shall have the capability to modify the selection and configuration of monitored telemetry via ground command.	Want to enable/disable monitor points	LRO, SDO
7424	CFS-502	The selection of the monitored points shall be table driven and configurable via ground uploads.	Need to modify/add/remove monitors	LRO, SDO
7426	CFS-503	The cFS shall be able to execute autonomous recovery sequences upon detection of threshold failures in monitored telemetry.	Provides for failure recovery	LRO, SDO
7428	CFS-504	The contents of recovery sequences shall be configurable via ground uploads.	Need to be able to modify/add/remove actions	LRO, SDO

5.0 Detailed Requirements

The following are the detailed requirements for Limit Checker.

ID	ReqID	Text	Rational	Heritage Reference
		5.1 Basic Requirements The following requirements are basic requirements of Limit Checker Application. Some of them are included here to avoid repeating these requirements for each applicable requirement.		
7432	LC1000	Upon receipt of a No-Op command, LC shall increment the LC Valid Command Counter and generate an event message.	Debug command to verify application is alive	LRO
7434	LC1001	Upon receipt of a Reset command, LC shall reset the following housekeeping variables to a value of zero: a) Valid Command Counter b) Command Rejected Counter c) Passive RTS Execution Counter d) Actionpoint Sample Count e) TLM Count f) RTS Execution Counter	Important for testing and on-orbit flight operations in order to start with a "clean slate"	LRO, SDO
7436	LC1002	For all LC commands, if the length contained in the message header is not equal to the expected length, LC shall reject the command and issue an event message.	Basic command verification in the event of SEU or memory corruption	LRO, SDO
7438	LC1003	If LC accepts any command as valid, LC shall execute the command, increment the LC Valid Command Counter and issue an event message.	Operators require feedback on command execution. Note that this only applies to "ground commands" (i.e. does not include requests that come from the scheduler)	LRO, SDO
7440	LC1004	If LC rejects any command, LC shall abort the command execution, increment the LC Command Rejected Counter and issue an error event message.	Operators require feedback on command execution	LRO, SDO
		5.2 Watchpoints – Monitoring Requirements The following requirements relate to the monitoring of data. It should be noted that LC watchpoints cannot be individually disabled.		
7442	LC2000	The flight software shall monitor a maximum of <PLATFORM_DEFINED> Watchpoints.	Need to define a limit for defining data structures	LRO, SDO

7444	LC2001	For each Watchpoint specified in the Watchpoint Definition Table (WDT) LC shall specify an age value which indicates when the data becomes "stale".	Need to provide a means of ensuring evaluated data is not stale	MMS
7446	LC2002	For each Watchpoint specified in the Watchpoint Definition Table (WDT) LC shall maintain the age of the data.	Need to provide a means of ensuring evaluated data is not stale	MMS
7448	LC2003	Upon receipt of a message, LC shall compare the data in the message to the table-defined value using the table-defined comparison value and comparison operator for each data point defined in the Watchpoint Definition Table (WDT) if the LC Application State is one of the following: a) Active b) Passive	Telemetry Watchpoints shall be evaluated whenever the message (packet) containing the data to be examined is received. If LC App is Disabled, LC does not process WDT or ADT. Disabled state useful for making updates to LC tables (note that LC does not have to be disabled in order to make WDT or ADT updates, however, care must be taken to ensure the validity of the relationship between the 2 tables during updates). Note that individual Watchpoints can't be disabled; Individual Actionpoints can be disabled. Note also that WDT can have unused entries.	LRO, (loosely SDO)
7450	LC2003.1	LC shall support the following comparison values: a) = b) != c) > d) >= e) < f) <=	Watchdog comparison values	
7452	LC2003.2	If the WDT comparison operator specifies that a Custom Function shall be performed, LC shall apply the custom function to the data contained in the message.	Provides capability to do a calculation on the data instead of just a comparison	LRO, (loosely SDO)
7454	LC2003.3	If the comparison result for a Watchpoint results in a False, LC shall set the Number of Consecutive True values to zero.	Restarts the consecutive True values	LRO, SDO

7456	LC2003.4	If the Watchpoint cannot be evaluated, LC shall set the Watchpoint Results Table to ERROR for the erroneous Watchpoint.	Want to report that there was an erroneous watchpoint	LRO
7458	LC2004	For each Watchpoint, the flight software shall maintain the following statistics in the dump-only Watchpoint Results Table: a) The result of the last relational comparison (False, True, Error, or Stale) b) The number of times this Watchpoint has been compared c) The number of times this Watchpoint has crossed from the False to True result d) The number of consecutive times the comparison has yielded a True result e) The cumulative number of times the comparison has yielded a True result f) Most recent FALSE to TRUE transition value g) Most recent FALSE to TRUE transition timestamp h) Most recent TRUE to FALSE transition value i) Most recent TRUE to FALSE transition timestamp j) Most recent comparison age	Provides results for each evaluation of the WDT.	LRO, SDO
7460	LC2005	Upon receipt of a table update indication, LC shall validate the Watchpoint Definition Table for the following: a) Valid operator b) Data size c) Message ID	Valid the items that can be validated when an table is updated (some items cannot be valued until runtime)	LRO
		5.3 Actionpoints - Actions Requirements Below are the requirements associated with the Actionpoints.		
7462	LC3000	Limit Checker shall support up to a maximum of <PLATFORM_DEFINED> Actionpoints.	Used to size the actionpoint table	
7464	LC3001	Upon receipt of a Sample Request, LC shall process the request specified Actionpoints defined in the Actionpoint Definition Table (ADT) if the LC Application State is one of the following: a) Active b) Passive	Processing of the Actionpoint Definition Table is scheduled upon receipt of a command (usually from the CFS Scheduler Application). If LC App is Disabled, LC does not process WDT or ADT. Disabled state useful for making	LRO

			updates to LC tables. Note that individual Watchpoints can't be disabled; Individual Actionpoints can be disabled. Can sample individual or all actionpoints.	
7466	LC3001.1	LC shall support the following Reverse Polish Operators: a) And b) Or c) Xor d) Not e) equals		
7468	LC3001.2	If the equation result for an Actionpoint results in a Pass, LC shall set the Number of Consecutive Fail values to zero.	Used to track consecutive failures. If consecutive failures > max then take action. Need to reset if Actionpoint doesn't fail.	LRO, SDO
7470	LC3001.3	If the Actionpoint cannot be evaluated, LC shall set the Actionpoint Results Table to ERROR for the erroneous Actionpoint.	Potential problems with the Watchpoint or the equation.	LRO
7472	LC3002	For Each table-defined Actionpoint, LC shall store the results in the dump-only Actionpoint Results Table if the Actionpoint state is either: a) Active b) Passive	Actionpoint is not evaluated if state is unused, disabled or permanently disabled	LRO, SDO (loosely)
7474	LC3002.1	If the Actionpoint equation results in a transition from PASS to FAIL, LC shall issue an event message indicating the failure.	Want to see when an Actionpoint first exceeds the limit.	LRO, SDO
7476	LC3002.1.1	If the PASS to FAIL transition event message has been sent for the table-defined number of times, LC shall apply the table-defined event message filter.		
7478	LC3002.2	If the Actionpoint equation results in a transition from FAIL to PASS, LC shall issue an event message indicating that the actionpoint is now within limits.	Want to see if an actionpoint which had fails, go back within limits. Helps to see if there is data that is "bad" but not so bad that it fails enough times to cause the RTS to be executed	LRO, SDO
7480	LC3002.2.1	If the FAIL to PASS transition event message has been sent for the table-defined number of times, LC shall apply the table-defined event message filter.		

7482	LC3002.3	If the equation has yielded a Fail result for the table-defined consecutive number of times limit and the Actionpoint is currently Active, LC shall: a) generate an event message b) send a command to start the table-defined RTS c) Increment the counter indicating Total count of commands sent to SC task to start an RTS	Takes the action defined in the actionpoint definition table	LRO, SDO
7484	LC3002.3.1	Once an RTS is initiated, LC shall change the current state of the associated Actionpoint to Passive.	Prevents an RTS from getting initiated more than once	LRO, SDO (loosely)
7486	LC3002.4	If the equation has yielded a Fail result for the defined consecutive number of times and the Actionpoint is currently Passive, LC shall: a) generate an event message indicating that the Actionpoint Failed but the action was not taken b) Increment the Passive RTS Execution Counter	Informs the ground that a actionpoint tripped but that the action was not taken	LRO, SDO
7488	LC3003	If the Actionpoint is Disabled, LC shall skip processing that actionpoint.	Actionpoints can be individually enabled/disabled. If disabled, don't process the definition	LRO, SDO (loosely)
7490	LC3004	If the Actionpoint is Unused, LC shall skip processing that actionpoint	Unused Actionpoint entries can't be processed	LRO, SDO (loosely)
7492	LC3005	If the Actionpoint is Permanently Disabled, LC shall skip processing that actionpoint	Useful for separation sequence actionpoints	LRO
7494	LC3006	For each Actionpoint, the flight software shall maintain the following statistics in the dump-only Actionpoint Results Table: a) The result of the last Sample (Pass, Fail, Error, or Stale) b) The current state (PermOff, Disabled, Active, Passive, Unused) c) The number of times this Actionpoint has crossed from the Fail to Pass state d) The number of times this Actionpoint has crossed from the Pass to Fail state e) The number of consecutive times the equation result = Failed f) The cumulative number of times the equation result = Failed g) The cumulative count of the RTS executions h) Total number of event messages sent	Provides detailed results of the Actionpoint evaluation	LRO
7496	LC3007	Upon receipt of a table update indication, LC shall validate the Actionpoint Definition Table for the following:	Not much can be validated since much is at run-time	LRO

		a) valid default state b) RTS number (in range) c) Event Type (DEBUG, INFO, ERROR, CRITICAL) d) Failure Count (in range) e) Action Equation syntax		
		5.4 Command Requirements The following are commands used by the Limit Checker Application.		
7498	LC4000	Upon receipt of a Set LC Application State To Active Command, LC shall set the state of the LC Application to Active.	Nominal mode for LC. LC performs all limit tests defined in the watchpoint definition table and will invoke stored command sequences if the results of watchpoint evaluations trigger an actionpoint condition as defined in the actionpoint definition table.	LRO, SDO
7500	LC4001	Upon receipt of a Set LC Application State to Passive Command, LC shall set the LC Application State to Passive.	Provides the ability to monitor data but not take actions. LC still monitors telemetry and report statistics, however, no actions will be taken.	LRO, SDO (SDO used the term Watch Mode)
7502	LC4002	Upon receipt of a Set LC Application State to Disable Command, LC shall set the LC Application State to Disabled.	Provides capability to stop all LC monitoring. Useful when updating watchpoint and/or Actionpoint definition tables.	LRO, SDO
7504	LC4003	Upon receipt of a Set Actionpoint to Active Command, LC shall set the state for the command-specified Actionpoint to ACTIVE such that the actionpoint is evaluated and the table-defined actions are taken based on the evaluation.	Provides the ability to enable the actionpoint so that appropriate actions are taken on the basis of current watchpoint states	LRO, SDO
7506	LC4004	Upon receipt of a Set All Actionpoints to Active Command, LC shall set the state for all Actionpoints to ACTIVE such that the actionpoints are evaluated and the table-defined actions are taken based on the evaluation.	Provide the ability to enable ALL actionpoints	LRO, SDO
7508	LC4005	Upon receipt of a Set Actionpoint to Passive Command, LC shall set the state for the command-	Provides ability to monitor data associated	LRO, SDO

		specified Actionpoint to PASSIVE such that the actionpoint is evaluated, however, no actions are taken.	with an action point but not take the action defined for that actionpoint	
7510	LC4006	Upon receipt of a Set All Actionpoints to Passive Command, LC shall set the state for the all Actionpoints to PASSIVE such that all actionpoints are evaluated, however, no actions are taken.	Provides ability to quiet all actionpoints	LRO, SDO
7512	LC4007	Upon receipt of a Set Actionpoint to Disabled Command, LC shall set the state for the command-specified Actionpoint to DISABLED such that the actionpoints are not evaluated and no actions are taken.	Disable an action point so that no evaluation is performed for the particular actionpoint.	LRO, SDO
7514	LC4008	Upon receipt of a Set All Actionpoints to Disabled Command, LC shall set the state for all Actionpoint to DISABLED such that: a) the actionpoints are not evaluated b) no actions are taken c) no event messages generated.	Provides the ability to "turn off" all actionpoint evaluations	LRO, SDO
7516	LC4009	Upon receipt of a Set Actionpoint to Permanent Disable, LC shall mark the command-specified Actionpoint such that the Actionpoint cannot be Activated.	Useful for separation sequences or other actionpoints that should only be active for a limited period of time and never re-enabled by the ground.	LRO
7518	LC4009.1	If a command is received to Activate an Actionpoint which has been permanently disabled, the command shall be rejected.	The only way to activate a permanently disabled actionpoint is to restart LC application or load a new Actionpoint table. Note that the restart of the LC application will only activate this Actionpoint if the Save Critical Data parameter is set to "No" (see LC9005)	LRO
7520	LC4010	Upon receipt of a Reset Actionpoint Statistics Command, LC shall set to zero, all of the following Actionpoint Statistics for the command-specified Actionpoints: a) Total number of FAIL to PASS transitions b) Total number of PASS to FAIL transitions c) Number of consecutive FAIL results d) Total number of FAIL results e) Total number of RTS executions f) Total number of event messages sent relating to that Actionpoint	Provides ability to start monitoring "from scratch"	LRO, SDO

7522	LC4011	Upon receipt of a Reset All Actionpoint Statistics Command, LC shall set to zero, all of the following Actionpoint Statistics for all Actionpoints: a) Total number of FAIL to PASS transitions b) Total number of PASS to FAIL transitions c) Number of consecutive FAIL results d) Total number of FAIL results e) Total number of RTS executions f) Total number of event messages sent relating to that Actionpoint	Provides ability to start monitoring "from scratch"	LRO, SDO
7524	LC4012	Upon receipt of a Reset Watchpoint Statistics Command, LC shall set to zero all of the following Watchpoint Statistics for the command-specified Watchpoints: a) Total sample count for this watchpoint b) Number of times result transitioned from FALSE to TRUE c) Number of consecutive TRUE results d) Total number of TRUE results e) Most recent FALSE to TRUE transition value f) Most recent FALSE to TRUE transition timestamp g) Most recent TRUE to FALSE transition value h) Most recent TRUE to FALSE transition timestamp	Provides ability to start monitoring "from scratch"	LRO, SDO
7526	LC4013	Upon receipt of a Reset All Watchpoint Statistics Command, LC shall set to zero, all of the following Watchpoint Statistics for all Watchpoints: a) Total sample count for this watchpoint b) Number of times result transitioned from FALSE to TRUE c) Number of consecutive TRUE results d) Total number of TRUE results e) Most recent FALSE to TRUE transition value f) Most recent FALSE to TRUE transition timestamp g) Most recent TRUE to FALSE transition value h) Most recent TRUE to FALSE transition timestamp	Provides ability to start monitoring "from scratch"	LRO, SDO
		5.4 Status Reporting		
7528	LC8000	LC shall generate a housekeeping message containing the following: a) Valid Command Counter b) Command Rejected Counter	Housekeeping telemetry to indicate basic LC status	LRO, SDO (loosely)

		c) Passive RTS Execution Counter (Total count of RTS sequences not initiated because either the LC application state or the state of the actionpoint that failed is set to Passive) d) Current LC Application State (LC_ACTIVE, LC_PASSIVE, LC_DISABLED)... e) Total count of actionpoints sampled while LC_ACTIVE or LC_PASSIVE... f) Total count of packets monitored for watchpoints (cmd and tlm) g) RTS Execution Counter (Total count of commands sent to SC task to start an RTS) h) Selected data from watchpoint results table i) Selected data from actionpoint results table		
		5.5 Initialization Requirements The following are the requirements associated with Limit Checker on an Application reset, cFE Processor Reset, or a cFE Power-on Reset		
7530	LC9000	Upon cFE Power-On LC shall initialize the following Housekeeping data to Zero (or value specified): a) Valid Command Counter b) Command Rejected Counter c) Passive RTS Execution Counter d) Current LC State to <PLATFORM_DEFINED> Default Power-on State e) Actionpoint Sample Count f) TLM Count g) RTS Execution Counter h) Watch Results (bitmapped) i) Action Results (bitmapped)	Need to initialize values to a default state on cFE Power-on reset	Derived
7532	LC9001	Upon cFE Power-On LC shall initialize the following Watchpoint data to Zero (or value specified) for all Watchpoints: a) The result of the last watchpoint relational comparison to STALE b) The number of times this Watchpoint has been compared c) The number of times this Watchpoint has crossed from the False to True result d) The number of consecutive times the comparison has yielded a True result e) The cumulative number of times the comparison has yielded a True result f) The value that caused the last False-to-True crossing, and the crossing time stamp g) The value that caused the last True-to-False crossing, and the crossing time stamp		

7534	LC9002	Upon cFE Power-On LC shall initialize the following Actionpoint data to Zero (or value specified for all Actionpoints: a) The result of the last Actionpoint Sample to STALE b) The current state as defined in the ADT c) The number of times this Actionpoint has crossed from the Fail to Pass state d) The number of times this Actionpoint has crossed from the Pass to Fail state e) The number of consecutive times the equation result = Failed f) The cumulative number of times the equation result = Failed g) The cumulative count of the RTS executions h) Total number of event messages sent		
7536	LC9003	Upon a cFE Processor Reset or LC Application Reset, if the <PLATFORM_DEFINED> Save Critical Data parameter is set to NO, LC shall perform the same initialization as a cFE Power-on (see LC9000, LC9001, LC9002, and 9003)	Configuration parameter indicates if data should be saved across a cFE Processor Reset or LC Application Reset.	
7538	LC9004	Upon a cFE Processor Reset or LC Application Reset, if the <PLATFORM_DEFINED> Save Critical Data parameter is set to YES, LC shall restore the following data: a) LC housekeeping data b) WDT c) Watchpoint Statistics d) ADT e) Actionpoint Statistics	Provides ability to “fly thru” a cFE processor reset or LC Application reset.	
7540	LC9004.1	LC shall initialize the LC Application State to <PLATFORM_DEFINED> Default Reset State	Provides flexibility. May want to “fly thru” a reset by defaulting to LC ACTIVE or maybe defaulting to PASSIVE so that no actions are taken. May want to default to LC Disable.	
7542	LC9004.1.1	If the platform defined Default Reset State indicates to use the state of the LC Application prior to the reset, LC shall set the state of the LC Application to the state restored from the CDS	Provides ability to maintain the state of the LC Application prior to the reset instead of using the default value	
7544	LC9004.2	If LC determines the Critical Data is invalid, LC shall perform the same initialization as a cFE Power-on (see LC9000, LC9001 and LC9002)	Need to validate critical data to make sure that the data looks reasonable	
7546	LC9005	Upon any initialization, LC shall validate the Watchpoint Definition Table for the following:	Valid the items that can be validated when an	LRO

		a) valid operator b) data size c) Message ID	table is updated (some items cannot be validated until runtime)	
7548	LC9006	Upon any initialization, LC shall validate the Actionpoint Definition Table for the following: a) valid default state b) RTS number (in range) c) Event Type (DEBUG, INFO, ERROR, CRITICAL) d) Failure Count (in range) e) Action Equation syntax	Valid the items that can be validated when an table is updated (some items cannot be validated until runtime)	LRO
7550	LC9007	Upon any initialization, LC shall subscribe to the messages defined in the WDT.	Must subscribe to all of the messages defined in the Watchpoint Definition Table after any reset.	LRO
7552	LC9007.1	For a cFE Processor Reset, If the Save Critical Data parameter is YES, LC shall subscribe to the messages defined in the WDT restored from the CDS	Need to use the WDT that was being used prior to the cFE Processor Reset instead of the default WDT	
7554	LC9007.2	For an LC Application Reset, If the Save Critical Data parameter is YES, LC shall subscribe to the messages defined in the WDT restored from the CDS	Need to use the WDT that was being used prior to the LC App reset instead of the default WDT	

Appendix A Terminology

This appendix contains the list of terminology for the CFS Limit Checker Application used in this document

- **WATCHPOINT -** A piece of data whose value is to be compared against a predefined constant
- **TELEMETRY WATCHPOINT -** A watchpoint that is read from a telemetry packet
- **WATCHPOINT DEFINITION TABLE -** The input table that holds the information necessary to define all LC watchpoints
- **WATCHPOINT RESULTS TABLE -** The output (dump-only) table that holds the results of watchpoint evaluations
- **ACTIONPOINT** - A logical relationship between one or more watchpoint results, specified in Reverse Polish Notation
- **ACTIONPOINT DEFINITION TABLE** - The input table that holds the information necessary to define all LC actionpoints
- **ACTIONPOINT RESULTS TABLE -** The output (dump-only) table that holds the results of actionpoint evaluations
- **CUSTOM FUNCTION -** A mission specific function that can be used in place of a standard comparison operator in a watchpoint definition
- **REVERSE POLISH NOTATION (RPN)** - A method of describing a mathematical equation. Like other notations, it uses both operands and operators. What makes RPN different is the placement of the operands/operators within the equation. RPN eliminates the need for parenthesis to remove operator precedence ambiguity in an equation