Software Requirements Specification for Discrete Module of Engine Data Acquisition Unit of Airbus Helicopter Generic Vehicle Monitoring System (GVMS)

**Document Number: H398-002-001-DSC**

**Version No: 2.7**

Proprietary Notice

This document and the information disclosed herein are proprietary data of Howell Instruments, Inc. Neither this document nor the information contained herein shall be reproduced, used, or disclosed to others without the written authorization of Howell Instruments, Inc.

|  |  |
| --- | --- |
| Written By: | Prajwal R (Software Engineer, ALTEN Global Technologies Private Limited) |
| Reviewed By: | Pallavi T (Software Engineer, ALTEN Global Technologies Private Limited)    Pavitra D (Quality Engineer – Quality Assurance, ALTEN Global Technologies Private Limited) |
| Approved By: | Shyamala B ( Senior Director - Quality Assurance and Management, ALTEN Global Technologies Private Limited)  cid:image001.jpg@01D61E3F.6C97D0F0 |
| Released By: | Madhu Chandra Srinivas (Associate Director- Software, ALTEN Global Technologies Private Limited)  C:\Users\afreen.p.SWSYS\Pictures\SIGNATURE.PNG |



**ALTEN Global Technologies Private Limited**

Table of Contents

1. [Amendment Record 5](#_bookmark0)
2. [Overview 9](#_bookmark2)
3. [Objectives 13](#_bookmark5)
4. [Acronyms, Terms and Definitions 14](#_bookmark6)
5. [Scope 15](#_bookmark8)
6. [References 15](#_bookmark9)
7. [Assumptions 16](#_bookmark11)
8. [Outstanding Issues 16](#_bookmark12)
9. [Document Control 16](#_bookmark13)
10. [Approvals 16](#_bookmark14)
11. [Distribution 17](#_bookmark15)
12. [Traceability 17](#_bookmark16)
13. [Responsibilities 17](#_bookmark17)
14. [Input Documents 17](#_bookmark18)
15. [Output Documents 17](#_bookmark19)
16. [High Level requirements 18](#_bookmark20)
    1. [Initialization 18](#_bookmark21)
       1. [Initialization - Components 18](#_bookmark22)
       2. [Processor 18](#_bookmark23)
       3. [Watchdog timer 20](#_bookmark24)
       4. [System Timer 20](#_bookmark25)
       5. [Onchip CAN peripheral 20](#_bookmark26)
       6. [Nested Vector Interrupt Controller 21](#_bookmark27)
       7. [FSMC 21](#_bookmark28)
       8. [SPI peripheral 21](#_bookmark29)
    2. [Input/Output 22](#_bookmark30)
       1. [Discrete Inputs 22](#_bookmark31)
       2. [Discrete Outputs 23](#_bookmark33)
       3. [28V Output Current Monitor 24](#_bookmark35)
    3. [A825 Communication 24](#_bookmark36)
       1. [Logical Communication Channel 24](#_bookmark37)
       2. [Invalid A825 message 25](#_bookmark38)
       3. [NSC Channel 25](#_bookmark39)
          1. [NSC message format 25](#_bookmark40)
          2. [NSC messages 26](#_bookmark42)
          3. [Discrete Outputs 29](#_bookmark44)
          4. [Switch excitation 29](#_bookmark46)
          5. [Setting RCI value 30](#_bookmark48)
          6. [Reset Discrete CPU 30](#_bookmark49)
       4. [NOC Channel 30](#_bookmark50)
          1. [NOC message format 30](#_bookmark51)
          2. [NOC messages 31](#_bookmark53)
          3. [NOC message reply (EDAU Discrete inputs, outputs and chip detector inputs) 35](#_bookmark55)
          4. [NOC message reply (Part number and CRC to Gateway Module) 35](#_bookmark56)
    4. [Chip Detection 36](#_bookmark57)
       1. [Chip Detector Inputs 36](#_bookmark58)
    5. [Module Configuration Data 37](#_bookmark60)
       1. [Module Configuration Data - Definition 37](#_bookmark61)
       2. [Module Configuration Data format - Chip detector inputs 37](#_bookmark63)
       3. [Module Configuration Data format - Discrete Outputs 38](#_bookmark65)
    6. [BIT 39](#_bookmark67)
       1. [PBIT 39](#_bookmark68)
          1. [PBIT fault indication 40](#_bookmark69)
          2. [CPU and RAM Test 40](#_bookmark70)
          3. [CRC Test 40](#_bookmark71)
          4. [Module Configuration Data - Validity Test 41](#_bookmark72)
       2. [CBIT 41](#_bookmark73)
          1. [CBIT fault indication 42](#_bookmark74)
          2. [Stack Test 42](#_bookmark75)
    7. [Kernel and Scheduler 42](#_bookmark76)
       1. [Kernel/Scheduler - Initializations 42](#_bookmark77)
       2. [Scheduler - Algorithm 43](#_bookmark78)
       3. [Tasks 43](#_bookmark79)
       4. [Semaphores 44](#_bookmark81)
       5. [Task activation by semaphore 44](#_bookmark82)
       6. [Task structure 44](#_bookmark83)
    8. [Modes of Operation 45](#_bookmark84)
       1. [Operational Modes 45](#_bookmark85)
       2. [PBIT Mode 45](#_bookmark86)
       3. [Normal Mode 45](#_bookmark87)
       4. [Error Mode 46](#_bookmark88)
    9. [Software Part Number 46](#_bookmark90)
       1. [Software Part Number – Application Software 46](#_bookmark91)
       2. [Software Part Number – Module Configuration Data Binary 47](#_bookmark92)
    10. [Resource Utilization 47](#_bookmark93)
        1. [Memory Usage 47](#_bookmark94)
        2. [Throughput Usage 48](#_bookmark95)

List of Tables

[Table 1: Amendment Record 5](#_bookmark1)

[Table 2: Acronyms and Definitions 14](#_bookmark7)

[Table 3: References 15](#_bookmark10)

[Table 4: Discrete Inputs 22](#_bookmark32)

[Table 5: Discrete Outputs 24](#_bookmark34)

[Table 6: List of NSC messages 26](#_bookmark43)

[Table 7: A825 payload for Discrete Outputs 29](#_bookmark45)

[Table 8: Action for SFC = 49156 29](#_bookmark47)

[Table 9: List of Document Object Codes 31](#_bookmark54)

[Table 10: Chip Detector Inputs 36](#_bookmark59)

[Table 11: Module Configuration Data - sections 37](#_bookmark62)

[Table 12: Module Configuration Data Format 38](#_bookmark64)

[Table 13: Module Configuration Data for discrete outputs 39](#_bookmark66)

[Table 14: Tasks in Discrete Module 43](#_bookmark80)

[Table 15: Discrete Module - LED Blink Rate 46](#_bookmark89)

**List of Figures**

[Figure 1: Functional block diagram - Engine Instrument System 10](#_bookmark3)

[Figure 2: DAU - decomposition 12](#_bookmark4)

[Figure 3: NSC message format 26](#_bookmark41)

[Figure 4: NOC message format 31](#_bookmark52)

# Amendment Record

*Table 1: Amendment Record*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Version No.** | **Description of Amendment** | **Change Request No.** | **Changed By** | **Release Date** |
| 1.1 | Initial Release | NA | Vijaya Bhaskar | 17-Dec-  2021 |
| 1.2 | Updated to address review comments   * Corrected spelling mistakes throughout the document * Verification method is added all functional and derived requirements * Section 2 overview updated * Section 4 updated to remove unused acronyms and missing acronyms were added * Section 6 references updated * Section 13 responsibilities updated * Rationale updated for H398-SRS- DSC-DRQ-19, H398-SRS-DSC-DRQ- 20, H398-SRS-DSC-DRQ-21, H398- SRS-DSC-DRQ-22, H398-SRS-DSC- DRQ-24, H398-SRS-DSC-DRQ-25, H398-SRS-DSC-DRQ-41 * Updated requirement   H398-SRS-DSC-FNC-27 to remove  spare discrete inputs   * Updated requirement   H398-SRS-DSC-FNC-28 to add  signals names for discrete outputs and also to remove spare discrete outputs   * Updated requirement   H398-SRS-DSC-FNC-29, H398-SRS- DSC-DRQ-39, H398-SRS-DSC-DRQ- 40, H398-SRS-DSC-FNC-54 for  better readability   * Updated requirement   H398-SRS-DSC-FNC-35 to changes serverID and payload columns   * Updated requirement   H398-SRS-DSC-FNC-37 to change  action column   * Updated requirement   H398-SRS-DSC-FNC-44 to change  RCI, Description and Payload columns and also DOC ids 2730, 2731 | 100011 | Vijaya Bhaskar | 11-Jan-  2022 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | and 2732 are added   * Updated requirement   H398-SRS-DSC-FNC-47 to remove  spare chip detector inputs   * Updated requirement   H398-SRS-DSC-FNC-50 to change  Channel state column and removed note 4   * Updated requirement   H398-SRS-DSC-FNC-55 to remove  note 3   * Updated requirement   H398-SRS-DSC-FNC-56 to remove  shall keyword from note section   * Changed requirement   H398-SRS-DSC-FNC-57 to H398- SRS-DSC-DRQ-57   * Updated requirement   H398-SRS-DSC-FNC-59 to add point 2   * Missing fields for requirement H398-SRS-DSC-FNC-63 is added * Updated requirement   H398-SRS-DSC-DRQ-64 to change  the description column   * Updated requirement   H398-SRS-DSC-DRQ-65 to add  requirement description |  |  |  |
| 1.3 | Updated to address QA comments   * Updated serial numbers for Enable power interface clock in requirement H398-SRS-DSC-DRQ-19. * Updated serial numbers for Enable SPI2 peripheral in requirement H398- SRS-DSC-DRQ-25. * Updated Table 4: Discrete Inputs Disc. Gnd/Open In #92 signal name as EDAU Maint 2 (Maintenance Interlock) and active state as Ground (GSE Serial Port Enable). * Updated Acronyms, Terms and Definitions in Table 2: GPIOA, FTP, SVN,GPIOE, DRQ, DSC, NVIC, FSMC, SCK, MOSI, RCI, FNC and INTR. * Updated Table 9: DOC ID 1400 for RCI. * Updated In amendment record 11th point to Payload columns. | 100011 | Vijaya Bhaskar | 13-Jan-  2022 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | * Updated Table 4: Discrete Inputs active state as N/A for Disc. Gnd/Open In #94 to Disc. Gnd/Open In #99 . * Updated Table 7: text highlight colour for table column heading. * Updated figure 2 to specify DLU as part of CMU+ * H398-SRS-DSC-FNC-29 updated to reframe the requirement to add “failed” |  |  |  |
| 1.4 | Updated to address DER audit observations and due to self-review.  Modified Requirement: H398-SRS-DSC-FNC-45 H398-SRS-DSC-DRQ-51  New Requirements: H398-SRS-DSC-FNC-76 H398-SRS-DSC-FNC-77 | 100039 | Vijaya Bhaskar | 24-Feb-  2022 |
| 1.5 | Updated to address Latest Spec observations and due to self-review.  Modified Requirement:  H398-SRS-DSC-DRQ-19 H398-SRS-DSC-FNC-29 H398-SRS-DSC-FNC-34 H398-SRS-DSC-FNC-35 H398-SRS-DSC-FNC-43 H398-SRS-DSC-FNC-44 H398-SRS-DSC-FNC-45 H398-SRS-DSC-FNC-56 H398-SRS-DSC-DRQ-64 H398-SRS-DSC-DRQ-67 H398-SRS-DSC-DRQ-49 H398-SRS-DSC-DRQ-75  Delete Requirements:  H398-SRS-DSC-FNC-76 H398-SRS-DSC-DRQ-39 H398-SRS-DSC-DRQ-40  Section 6: Updated Spec version in reference table. | 100054 | Vijaya Bhaskar | 30-Mar-  2022 |
| 1.6 | Updated to address QA comments and self review  Section 1: updated Amendment record of 1.5 version | PR100054 | Vijaya Bhaskar | 06-Apr-  2022 |

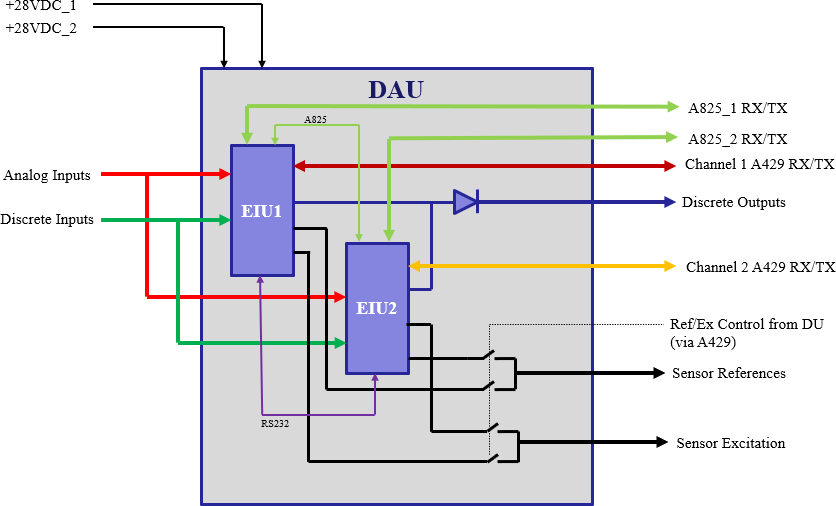
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | H398-SRS-DSC-DRQ-40 – updated the section number to 16.3.3.6  H398-SRS-DSC-DRQ-21 – updated the  system timer to 10000 interrupts |  |  |  |
| 1.7 | Updated to address DER SOI#2 action and self review  Modified Requirements:  H398-SRS-DSC-FNC-37 H398-SRS-DSC-FNC-44 H398-SRS-DSC-FNC-77 H398-SRS-DSC-DRQ-64  Added Requirements:  H398-SRS-DSC-DRQ-79 H398-SRS-DSC-DRQ-80  General requirements:  Section 16.10 Resource Utilization is added Amendment record is updated | PR100078 | Vijaya Bhaskar | 12-Apr-  2022 |
| 2.0 | The following Sections are modified: Section 2 Overview  Section 3 Objectives  Section 4 Acronyms, Terms and Definitions Section 6 References  The following requirement is modified:  H398-SRS-DSC-FNC-27 | PR100169 | Prajwal R | 31-01-  2024 |
| 2.1 | The following section is Updated to address review comments:   1. Section 6 References   The following section is updated as per self review:   1. Section15 Output Documents | PR100169 | Prajwal R | 10-02-  2024 |
| 2.2 | The following sections are updated to address QA comments:  Section 6 References  Section 10 Approvals  Section 13 Responsibilities Section 14 Input Documents | PR100169 | Prajwal R | 14-02-  2024 |
| 2.3 | The following requirement is modified as per self- review  H398-SRS-DSC-FNC-27 | PR100213 | Prajwal R | 20-05-  2024 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 2.4 | The following requirements are modified as per review comments  H398-SRS-DSC-FNC-28 H398-SRS-DSC-FNC-36 H398-SRS-DSC-FNC-44 H398-SRS-DSC-FNC-35 H398-SRS-DSC-FNC-27 H398-SRS-DSC-FNC-45 H398-SRS-DSC-DRQ-74  H398-SRS-DSC-FNC-29 is updated as H398- SRS-DSC-DRQ-29  H398-SRS-DSC-FNC-47 is updated as H398- SRS-DSC-DRQ-47  The following sections got modified as per review comments:  Section:2 Overview Section:6 References | PR100213 | Prajwal R | 08-07-  2024 |
| 2.5 | The following sections are Updated to address QA comments:  Front Page  Section:1 Updated Version.2.4 in Amendment Record  Section:6 References Section:10 Approvals | PR100213 | Prajwal R | 10-07-  2024 |
| 2.6 | The below requirements are modified H398-SRS-DSC-FNC-27  H398-SRS-DSC-FNC-28 H398-SRS-DSC-FNC-36 H398-SRS-DSC-FNC-35  H398-SRS-DSC-FNC-44 | PR100243 | Prajwal R | 01-08-  2024 |
| 2.7 | The below requirements are Updated as per QA comments  H398-SRS-DSC-FNC-27 H398-SRS-DSC-FNC-28 | PR100243 | Prajwal R | 17-08-  2024 |

# Overview

The EIS consist of Display Unit (DU), Engine Data Acquisition Unit (EDAU), Configuration Management Unit plus NVM (CMU+) and Data Logging Unit (DLU). It will be complemented by pilot input devices and interconnecting harness. The EIS will interface with the Airbus AS532U2 engine sensors and other applicable aircraft systems.

Figure: Functional block diagram - Engine Instrument System describes the functional blocks of Engine Instrument System (EIS).



*Figure 1: Functional block diagram - Engine Instrument System*

## Engine Data Acquisition Unit (EDAU)

The EDAU shall function to process airframe and engine data and transmit this information to be graphically presented on a cockpit display. The EDAU will process analog and discrete inputs, apply user defined logic to those inputs, and may generate aural and visual Caution and Warning alerts.

The EDAU is utilized to perform the following functions:

1. Provide analog to digital conversion of aircraft and engine systems
2. Provide discrete bit information via A429 for Crew Alerts (CAS)
3. Provide field loadable software update via maintenance port
4. Process data
5. Filter analog and digital signal data
6. Receive/Transmit data
7. Provide discrete strapping for multiple aircraft and engine configurations

To increase the reliability of the system, EDAU will have two Data Acquisition Units (DAU) packed together. Both the Data Acquisition Units (DAU) will run the data acquisition application software, process the same inputs and produce same outputs independently at the same time.

The primary function of the DAU will be acquiring analog engine parameters, discrete inputs, and other aircraft system information for processing and converting to a digital format for display via ARINC 429 to four Display Units (DU).

Each DAU will contain three Boards/Modules, each with its own CPU: The Gateway Board (BH35112) for digital buses, the Analog Board (BH35113) for analog inputs and the Discrete Board (BH35114) for discrete I/O and chip burner. Inter-board communication and communication with CMU+ / DLU will be on CAN bus based on ARINC 825 protocol.

## Configuration Module Unit Plus NVM (CMU+)

The CMU+ stores Aircraft Configuration Data (ACD). The ACD includes options related to the engine parameters, sensor interface, and airframe configurations. The DAU accesses CMU+ on system initialization to determine the current configuration and active interfaces.

The CMU+ is utilized to perform the following functions:

* 1. Provide configuration data to each DAU utilizing separate, isolated hardware for each DAU
  2. Support the DAU system configuration so that a replacement DAU will function the same as the DAU being replaced after installation and initialization
  3. The EDAU shall provide the capability to support different engine and aircraft configurations.
  4. The CMU+ shall store item configuration data for entire EIS
  5. The CMU+ shall contain system NVM

The CMU+ is internally redundant with two separate configuration sections packaged into one enclosure (separate power supplies). One section CMU+ #1 is connected to DAU#1 through ARINC825 (CAN #1), and the other section CMU+ #2 is connected to DAU#2 through ARINC825 (CAN #1). Internally CMU+ #1 and CMU+ #2 are connected through an independent ARINC825 (CAN #2).

The CMU+ installed in position #1 will have a USB device interface for programming/accessing configuration information in Maintenance Mode.

Figure: DAU - decomposition illustrates the decomposition of the DAU into its various internal modules:

EDAU

A429 Output RS232 Output Audio Output



USB Input/Output

CMU+ Module / DLU Module

RS422 Output A429 Input

Discrete Input

Gateway Module

ARINC825

28V Power Input

Analog Input RS232 Input

Analog Input Discrete Input 28V power Input

Analog Output

Discrete Input 28V Power Input Discrete Output

Analog Module

Discrete Module

ARINC825

*Figure 2: DAU - decomposition*

The Discrete Module consists of five components namely, Application Software, Boot Loader, Boot Configuration, Module Configuration Data and Maintenance Software. The Boot Loader’s responsibility is to pass control to either Application Software or Maintenance Software at Power On based on the boot configuration. If USB connection is found active, control will be passed to Maintenance Software (Maintenance Mode), else to Application Software (Normal Mode). The Maintenance Software is responsible for uploading the Aircraft Configuration Data. Howell Instruments, Inc. will be developing this

Software. The Aircraft Configuration Data is a configuration file which describes the multiple systems/equipment variations as well as engine parameter. This data is needed so that the DAU Application Software can support multiple aircraft configurations.

The Application Software is responsible for reading all the discrete inputs and set the discrete outputs. The software for the Discrete module follows a cyclic model with interrupts enabled. The main functionality of the Discrete Module is to perform the following:

* + 1. Read and Process Discrete Inputs
    2. Service Discrete Outputs

Read Discrete Inputs: Read the inputs. Apply inversion, persistence, and debouncing logic and update the values accordingly.

Service Discrete Outputs: Read the outputs from the CAN bus buffer. These outputs are transmitted by the Gateway. Send these values to the corresponding discrete output devices.

## Can bus communication:

The Discrete Module uses a CAN bus to communicate with the Gateway Module. Whenever a CAN message is received an interrupt is triggered. The handler routine copies the received message to a temporary buffer. It then identifies the Logical Communication Channel (LCC) of the received message. For each category of LCC, there are separate queues. The handler inserts the received message in the respective buffer. It then parses the payload, performs the required action and transmits the appropriate response.

## Scheduler

The Application Software uses a kernel to create and schedule jobs. The kernel has a fixed priority pre- emptive scheduler. The scheduler ensures that at any given time, the processor executes the highest priority task of all those tasks that are currently ready to execute. The pre-emptive scheduler has a clock interrupt task that provides the scheduler with an option to switch after the task has had a given period to execute—the time slice.

The kernel uses semaphores to synchronize the tasks. Three operations can be performed on a semaphore: Creation, Pending, and Posting. Each task has to run periodically. A unique semaphore is created for every task. A task desiring to run will perform a PEND on its corresponding semaphore. The scheduler is responsible for releasing the semaphores at correct intervals so that the tasks execute at the required frequency. The scheduler releases the semaphore for a task by doing a POST on the semaphore. Once the semaphore is released, the task is made to run. The task then executes its set of instructions and makes the next PEND. It then waits for a POST from the scheduler. This process goes indefinitely.

# Objectives

The objective of this document is to specify the Software High Level Requirements for the Discrete Module of the Engine Data Acquisition Unit (EDAU) for Airbus Helicopter model AS532U2 aircraft.

# Acronyms, Terms and Definitions

The list of Acronyms used in this document are described in Table: Acronyms and Definitions.

*Table 2: Acronyms and Definitions*

|  |  |
| --- | --- |
| **Term** | **Definition** |
| A825 | ARINC Specification 825 |
| ADC | Analog to Digital Convertor |
| ARINC | Aeronautical Radio, Incorporated |
| BIT | Built-In Test |
| CAN | Controller Area Network |
| CAS | Crew Alert System |
| CBIT | Continuous Built-In Test |
| CMU+ | Configuration Management Unit Plus NVM |
| CPU | Central Processing Unit |
| CRC | Cyclic Redundancy Check |
| DRQ | Derived Requirements |
| DSC | Discrete |
| EDAU | Engine Data Acquisition Unit |
| DLC | Data Length Count |
| DOC | Data Object Code |
| FSMC | Flexible Static Memory Controller |
| FTP | File Transfer Protocol |
| FNC | Functional |
| GPIO | General Purpose Input and Output |
| GPIOA | General Purpose Input and Output-Port A |
| GPIOE | General Purpose Input and Output-Port E |
| HCLK | High Speed Clock |
| INTR | Interrupt |
| LCC | Logical Communication Channel |
| LCL | "Local" Bit |
| MCD | Module Configuration Data |
| MISO | Master In Slave Out |
| MOPS | Minimum Operational Performance Standards |
| MOSI | Master Out Slave In |
| NOC | Normal Operation Channel |
| NSC | Node Service Channel |

|  |  |
| --- | --- |
| NVM | Non-Volatile Memory |
| NVIC | Nested Vectored Interrupt Controller |
| PBIT | Power-On Built-In Test |
| PCLK | Peripheral Clock |
| PLL | Phase Lock Loop |
| PLLCLK | Phase Lock Loop Clock |
| PVT | "Private" Bit |
| QA | Quality Assurance |
| RAM | Random Access Memory |
| RCI | Redundancy Code Identifier |
| RPM | Revolutions Per Minute |
| RSD | "Reserved" Bit |
| SCK | Serial clock |
| SFC | Service Function Code |
| SMT | Service Message Type |
| SVN | SubVersion |

# Scope

The Scope of this document is limited to specifying the Software High Level Requirements of the Discrete Module Application software of the EDAU.

# References

The Table: References provides the Reference documents used for Software High Level Requirements of the Discrete Module of the EDAU of Airbus AS532U2 Engine Instrument System.

*Table 3: References*

|  |  |  |
| --- | --- | --- |
| **Source** | **Document No.** | **Title** |
| RTCA | DO-178B | Software Considerations in Airborne Systems and Equipment Certification |
| Howell | SYS2160SRS | Airbus Helicopter AS532U2 Retrofit Engine Instrument System Requirements Specification |
| Howell Instruments, | HE0398DHA | Hardware Architecture of Discrete Board |

|  |  |  |
| --- | --- | --- |
| Inc. |  |  |
| Howell Instruments, Inc. | BH35114 | PCB Assembly of Discrete Board |
| Aeronautical Radio, Inc. | ARINC SPECIFICATION 825-2 | GENERAL STANDARDIZATION OF CAN (CONTROLLER AREA NETWORK) BUS PROTOCOL FOR AIRBORNE USE |
| ALTEN Global Technologies Private Limited | H398-001-002 | Software Development Plan |
| ALTEN Global Technologies Private Limited | H398-001-006 | Software Requirement Standards |

# Assumptions

None

# Outstanding Issues

None

# Document Control

This document is under change control. After baseline, any changes to this document shall be carried out in accordance with H398-001-004 (Software Configuration Management Plan).

# Approvals

The document will be reviewed to meet the objectives of DO-178B level A and approved by ALTEN GT QA.

# Distribution

This document will be distributed over a secure FTP server to Howell Instruments, Inc.

# Traceability

Traceability to SES (System Equipment Specification) is provided in H398-002-002-DSC.

# Responsibilities

1. Software Development Lead shall write the Software High Level Requirements in ReMa.
2. Reviewers shall review the requirements based on the DO-178B SRS check points provided as attributes in ReMa. Developers to respond to these comments and then change the status to <Looked Into> in ReMa.
3. Once all the comments from Reviewer are made to 'Looked Into' and the requirement status is Closed in ReMa,
4. Project leader is responsible for baselining the document to SVN through ReMa as well as checking in the exported document into the path

<http://192.168.1.230/svn/A21HOWBLDAU/V2_H398/ACCORD/SW/Trunk/Documents/SRS>

# Input Documents

* SYS2160SRS
* Software Development Plan (H398-001-002)
* Software Requirements Standards (H398-001-006)

# Output Documents

The Output documents are as mentioned below

* Software Requirements Specification for Discrete Module of Engine Data Acquisition Unit of Airbus EDAU (H398-002-001-DSC)
* Traceability Matrix (H398-002-002-DSC)

# High Level requirements

This section explains the Software High Level Requirements of Discrete module of the EDAU unit.

## Initialization

This section specifies the Software High Level Requirements for the initialization of the Discrete Module.

* + 1. Initialization - Components

The Discrete Module initializes the following after Power On :

1. Processor
2. Watchdog timer
3. System Timer
4. On-chip CAN peripheral
5. Nested Vector Interrupt Controller
6. SPI 2
7. FSMC
   * 1. Processor

Requirement ID: H398-SRS-DSC-DRQ-19 MOPS: No

Safety Requirement: No

Rationale: The Processor initialization required to execute the software is derived from hardware architecture document HE0398DHA and to execute the application software in normal mode.

Verification Method: Testing

The Discrete Module shall initialize the processor as follows:

1. Reset the RCC Clock Configuration to the default reset state
2. Enable HSE (high speed external clock).
3. Enable power interface clock
4. Configure power regulator voltage scale 1
5. Set CPU clock as System Clock
6. Set PCLK1 = HCLK clock/ 4
7. Set PCLK2 = HCLK clock/ 2
8. Set 5 wait states as the latency period on Flash
9. Enable instruction cache
10. Enable data cache
11. Enable flash prefetch buffer
12. Set PLLCLK to 168 MHz
13. Enable PLL.
14. Select PLL as system clock source.
15. Initialize the interrupt vector table in RAM.
16. Set the Vector Table base address at RAM
17. Configure the group priority equal sixteen and sub-priority is zero
18. Set up the system tick interrupt into vector table in RAM
19. Set the priority of the system tick interrupt
20. Set up the PendSV interrupt into vector table in RAM
21. Set the priority of the PendSV interrupt
22. Enable CRC peripheral clock
23. Initializing the FPU context save not in lazy mode
24. Enable floating point coprocessor
25. Enable FSMC Clock, system configuration clock and GPIO Clock.
26. Enable Memory Management Fault, Bus Fault and Usage Fault exceptions in SHCSR register
27. Enable Usage Fault in Configuration Control Register
28. Enable GPIOA clock.
29. Configure GPIO Pin 0 to Pin 6 of Port A as discrete input chip select 1 to discrete input chip select 7 respectively to enable the discrete input ICs of discrete-to-digital converters (speed = 50Mhz, mode = output mode, pull-up/pull-down as no pull-up/pull down, output type as output push pull)
30. Enable GPIOB clock.
31. Configure GPIO Pin 2 of Port B to toggle heartbeat LED (speed = 50Mhz, mode = output mode, pull- up/pull-down as no pull-up/pull down, output type as output push pull)
32. Configure GPIO Pin 10 of Port B as discrete output to reset all discrete outputs on power initialization (mode = output mode, pull-up/pull-down as no pull-up/pull down, output type as output push pull)
33. Configure GPIO Pin 11 and Pin 12 of Port B as Discrete Input Select 0 and Discrete Input Select 1 to control the discrete input circuit selected SPI function (mode = output mode, pull-up/pull-down as no pull- up/pull down, output type as output push pull)
34. Enable GPIOC clock.
35. Configure GPIO Pin 10 of Port C as reference control for the chip detectors which is used to enable the reference voltage (mode = output mode, pull-up/pull-down as no pull-up/pull down, output type as output push pull)
36. Enable GPIOD clock
37. Configure GPIO pins 0, 1, 8, 9, 10, 14 and 15 of port D as FSMC alternate function output pins for data

bus bits 2, 3, 13, 14, 15, 0 and 1 respectively (speed = 50Mhz, mode = alternate function mode, pull-up/pull- down as no pull-up/pull down, output type as output push pull)

1. Configure GPIO pins 4, 5 and 7 of port D as FSMC OE, FSMC WE and FSMC NE1 FSMC alternate function output pins respectively (speed = 50Mhz, mode = alternate function mode, pull-up/pull-down as no pull-up/pull down, output type as output push pull)
2. Enable GPIOE clock
3. Configure GPIO pins 7, 8, 9, 10, 11, 12, 13, 14 and 15 of port E as FSMC alternate function output pins

for data bus bits 4, 5, 6, 7, 8, 9, 10, 11 and 12 respectively (speed = 50Mhz, mode = alternate function mode, pull-up/pull-down as no pull-up/pull down, output type as output push pull)

1. Enable GPIOF clock
2. Configure GPIO pins 0, 1, 2, 3, 4 and 5 of port F as FSMC alternate function output pins for address

bus bits 0, 1, 2, 3, 4 and 5 respectively (speed = 50Mhz, mode = alternate function mode, pull-up/pull-down as no pull-up/pull down, output type as output push pull)

* + 1. Watchdog timer

Requirement ID: H398-SRS-DSC-DRQ-20 MOPS: No

Safety Requirement: No

Rationale: The Watchdog Timer initialization is required to monitor the processor idle state Verification Method: Testing

The Discrete Module shall initialize the Watchdog Timer for 1 second.

* + 1. System Timer

Requirement ID: H398-SRS-DSC-DRQ-21 MOPS: No

Safety Requirement: No

Rationale: The System Timer initialization is required to perform periodic scheduling of tasks. Verification Method: Testing

The Discrete Module shall initialize the System Timer to generate 10000 interrupts every second.

* + 1. Onchip CAN peripheral

Requirement ID: H398-SRS-DSC-DRQ-22 MOPS: No

Safety Requirement: No

Rationale: The initialization of CAN peripheral is required to communicate to Gateway module. Verification Method: Testing

The Discrete Module shall initialize the on chip CAN peripheral to communicate bi-directionally.

* + 1. Nested Vector Interrupt Controller

Requirement ID: H398-SRS-DSC-DRQ-23 MOPS: No

Safety Requirement: No

Rationale: The initialization of Nested Vector Interrupt Controller required to execute the software Verification Method: Testing

The Discrete Module shall initialize the Interrupt Vector Table to reside in RAM and set the NVIC Priority Group to 4.

* + 1. FSMC

Requirement ID: H398-SRS-DSC-DRQ-24 MOPS: No

Safety Requirement: No

Rationale: The FSMC initialization required to interact with external memory for discrete outputs, chip detector inputs and 28V discrete output current monitors

Verification Method: Testing

The Discrete Module shall initialize the FSMC to interact with discrete outputs, chip detector inputs and 28V discrete output current monitors.

* + 1. SPI peripheral

Requirement ID: H398-SRS-DSC-DRQ-25 MOPS: No

Safety Requirement: No

Rationale: The SPI peripheral initialization required to interface with discrete to digital converters for reading the discrete inputs.

Verification Method: Testing

The Discrete Module shall initialize the SPI2 peripheral as follows:

1. Enable SPI2 peripheral clock
2. Set GPIO Pins #13 and #15 for SCK and MOSI as Alternate Function mode
3. Set GPIO Pin #14 for MISO as Alternate Function mode
4. Configure the SPI2 global interrupt
5. Configure SPI2 peripheral as Direction is Full Duplex, SPI mode is Master mode, Data size is 16 bits, Clock Polarity as Low, Clock phase as Edge1, Slave as soft. Baud rate pre scalar to 16, first bit transmission is MSB
6. Set up SPI2 interrupt
7. Enable SPI2 peripheral.

## Input/Output

This section specifies Software High Level Requirements for Discrete Inputs and Discrete Outputs.

* + 1. Discrete Inputs

Requirement ID: H398-SRS-DSC-FNC-27 MOPS: No

Safety Requirement: Yes Rationale: NA

Verification Method: Testing

The Discrete Module shall read and process the Discrete inputs mentioned in Table: Discrete Inputs for every 100ms

*Table 4: Discrete Inputs*

|  |  |  |  |
| --- | --- | --- | --- |
| **Disc Ref** | **Identifier** | **Active State** |  |
| **Super Puma** |
| **AS532U2** |
| DI\_1 | ON\_GROUND | GND | DI003 |
| DI\_2 | OEI30\_ENG1 | GND | DI003 |
| DI\_3 | OEI30\_ENG2 | GND | DI003 |
| DI\_4 | OEI2\_ENG1 | GND | DI003 |
| DI\_5 | OEI2\_ENG2 | GND | DI003 |
| DI\_6 | BVALVE1\_IN | GND | DI003 |
| DI\_7 | BVALVE2\_IN | GND | DI003 |
| DI\_8 | Spare |  |  |
| DI\_91 | DAY/NIGHT | GND = Night | DI003 |
| DI\_40 | Spare |  |  |
| DI\_92 | NORMAL/NVG | GND = Normal | DI003 |
| DI\_9 | ENG\_START1 | 28V | DI28 |
| DI\_10 | ENG\_START2 | 28V | DI28 |
| DI\_11 | ENG\_GOV1 | GND | DI003 |
| DI\_12 | ENG\_GOV2 | GND | DI003 |
| DI\_13 | ENG\_CHIP1 | GND | DI003 |

|  |  |  |  |
| --- | --- | --- | --- |
| DI\_14 | ENG\_CHIP2 | GND | DI003 |
| DI\_15 | T1 | GND | DI003 |
| DI\_16 | T2 | GND | DI003 |
| DI\_17 | BLEED OFFSET 1 | GND | DI003 |
| DI\_18 | BLEED OFFSET 2 | GND | DI003 |
| DI\_19 | XMSN\_M\_PRESS | GND | DI003 |
| DI\_20 | XMSN\_SB\_PRESS | GND | DI003 |
| DI\_21 | XMSN\_MGBT | GND | DI003 |
| DI\_22 | XMSN\_CHIP | GND | DI003 |
| DI\_23 | XMSN\_IGBT | GND | DI100 |
| DI\_24 | XMSN\_TGBT | GND | DI100 |
| DI\_25 | HYD\_AUX.PUMP\_PRESS | GND | DI003 |
| DI\_26 | HYD\_AP\_PRESS | GND | DI003 |
| DI\_27 | HYD\_MAIN\_PRESS | GND | DI003 |
| DI\_28 | HYD\_MAIN\_LEVEL | GND | DI003 |
| DI\_29 | HYD\_RH\_PRESS | GND | DI003 |
| DI\_30 | HYD\_RH\_LOW\_LEVEL | GND | DI003 |
| DI\_31 | PFT\_TEST | GND | DI003 |
| DI\_35 | AUX PUMP FAIL | GND |  |
| DI\_32 | PWR.C | GND | DI003 |
| DI\_33 | THRT | GND | DI003 |
| DI\_34 | SERVO | GND | DI003 |
| DI\_36 | REGM1 |  | DI003 |
| DI\_37 | REGM2 |  | DI003 |
| DI\_38 | MAN1\_IDLE\_DETENT | 28V | DI28 |
| DI\_39 | MAN2\_IDLE\_DETENT | 28V | DI28 |

* + 1. Discrete Outputs

Requirement ID: H398-SRS-DSC-FNC-28 MOPS: No

Safety Requirement: Yes Rationale: NA

Verification Method: Testing

The Discrete Module shall set the Discrete Outputs mentioned in Table: Discrete Outputs for every 100ms

*Table 5: Discrete Outputs*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Parameter** | **Active State** | **Super Puma AS532U2** | **Channel** | **Default value** |
| ΔNG1\_THRESH | GND | DI003 | Disc. Out #10 (DOG400m\_2) | N/A |
| ΔNG2\_THRESH | GND | DI003 | Disc. Out #11 (DOG400m\_3) | N/A |

* + 1. 28V Output Current Monitor

Requirement ID: H398-SRS-DSC-DRQ-29 MOPS: No

Safety Requirement: Yes

Rationale: Required to define Output status monitor Verification Method: Testing

The Discrete Module shall determine the status of 28V Discrete Output current monitor for every 100ms as follows

1. All 28V discrete outputs (Disc. Out #1 to Disc. Out #8) are disabled and set 28V Discrete Output Fail to True, upon short detection.
2. After 5 seconds from detection of 28V Discrete Output Fail to True, all 28V outputs are re- enabled and set 28V Discrete Output Fail to False.
3. Upon a subsequent short detection, set 28V Discrete Output Fail to True and set all 28V discrete outputs to disabled/failed state until a power cycle

Note:

1. Refer discrete hardware architecture document (HE0398DHA) to know where the 28V Discrete Output current monitors are located.
2. Discrete Module transmits 28V Discrete Output Fail status to Gateway module via NOC messages (Doc Id = 750 )

## A825 Communication

This section specifies Software High Level Requirements for the A825 communication.

* + 1. Logical Communication Channel

Requirement ID: H398-SRS-DSC-FNC-31 MOPS: No

Safety Requirement: No Rationale: NA

Verification Method: Testing

The Discrete Module shall use the following Logical Communication Channels to communicate over the A825 bus

1. Node Service Channel (NSC)
2. Normal Operation Channel (NOC)
   * 1. Invalid A825 message

Requirement ID: H398-SRS-DSC-FNC-32 MOPS: No

Safety Requirement: No Rationale: NA

Verification Method: Testing

The Discrete Module shall ignore an A825 message if it is invalid.

Note: an invalid message is one which does not satisfy the specified NSC/NOC message formats.

* + 1. NSC Channel

This section specifies requirements for messages sent over NSC channel of the A825 bus.

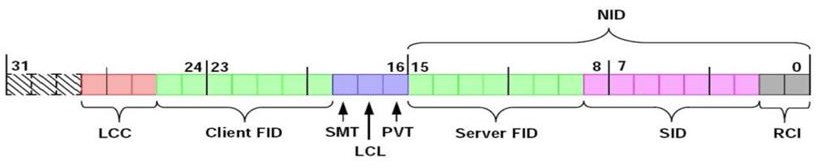
* + - 1. NSC message format

Requirement ID: H398-SRS-DSC-FNC-34 MOPS: No

Safety Requirement: No Rationale: NA

Verification Method: Testing

The Discrete Module shall use the message structure illustrated in Figure: NSC message format while using the NSC channel



*Figure 3: NSC message format*

Note:

1. LCC - Indicates the Logical Communication Channel
2. Client FID - always 42
3. SMT - Indicates whether message is Request (1) or Response (0)
4. LCL - messages is designated only for the network the transmitting node resides in - always 1
5. PVT - messages which have no meaning to nodes other than those which are specifically programmed to use them - always 1
6. Server FID - Always 42
7. Server ID - Specifies the recipient node for the message
8. RCI - Redundancy Channel Identifier
   * + 1. NSC messages

Requirement ID: H398-SRS-DSC-FNC-35 MOPS: No

Safety Requirement: No Rationale: NA

Verification Method: Testing

The Discrete Module shall use the Service Function Codes specified in Table: List of NSC messages in its A825 communication over the Node Service Channel

*Table 6: List of NSC messages*

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **SFC** | **Clien t FID** | **SMT** | **LCL** | **PVT** | **Server FID** | **Server ID** | **Rx/ Tx** | **RCI** | **DLC** | **Description** | **Payload** |
| NSC\_SE T\_RCI (49152) | 42 | 1 | 1 | 1 | 42 | 3 or 0 | Rx | 3 | 3 | Gateway to command the discrete module to  adopt the specified RCI | The first two bytes of the payload contain the SFC.  The next byte |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |  |  | contains the desired RCI.  Once this data has been received  , each data acquisiti on node will use this RCI for ARINC 825  commun ication |
| NSC\_SE T\_DOUT1 (49153) | 42 | 1 | 1 | 1 | 42 | 3 or 0 | Rx | 0 | 8 | Command to the Discrete Module to set output discrete signals with the value indicated by the payload. | The first two bytes of the payload contain the SFC and the desired discrete output status of the first 16  discrete outputs in next four bytes. . 0=OFF,  1 = ON,  and 2 = FLASH.  The status of discrete output number  one is |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |  |  | containe d in the least significa nt two  bits of the third byte of the payload. The status of discrete output number two is containe d in the 3rd and 4th least significa nt bits of the third byte of the payload,  etc. |
| NSC\_EX CITE\_SW ITCH (49156) | 42 | 1 | 1 | 1 | 42 | 3 or 0 | Rx | 0 | 3 | Command to Discrete Module to switch on/off the excitation to the chip detectors. (Note: This is not applicable as of now and it is retained  for future use) | The first two bytes of the payload contain the SFC.  The third byte is  (OFF = 0, ON =  1). |
| NSC\_RE SET(4454 4) | 42 | 1 | 1 | 1 | 42 | 3 | Rx | 0 | 3 | Command to Discrete Module to  reset the Discrete CPU. | The first two bytes of the payload contain the SFC |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |  |  | ‘NSC\_R ESET'. |

Note:

1. Rx/Tx - Indicates whether Discrete module Receives (Rx) or Transmits (Tx) the message
2. DLC - Data Length Count - size of the payload in bytes.
   * + 1. Discrete Outputs

Requirement ID: H398-SRS-DSC-FNC-36 MOPS: No

Safety Requirement: No Rationale: NA

Verification Method: Testing

The Discrete shall receive the values of the Discrete Outputs mentioned in Table: A825 payload for Discrete Outputs from the payload of the A825 message with LCC = NSC and SFC = 49153.

*Table 7: A825 payload for Discrete Outputs*

|  |  |  |
| --- | --- | --- |
| **Output Signal** | **Payload Byte #** | **Bit Position** |
| ΔNG1\_THRESH | 3 | 3, 4 |
| ΔNG2\_THRESH | 3 | 5, 6 |

* + - 1. Switch excitation

Requirement ID: H398-SRS-DSC-FNC-37 MOPS: No

Safety Requirement: No Rationale: NA

Verification Method: Testing

The Discrete Module shall start performing the below actions when it receives an A825 message with LCC

= NSC and SFC = 49156 from Gateway module.

* + - * 1. Processing of discrete inputs, setting discrete outputs.
        2. Processing of chip detector inputs as mentioned in Table: Action for SFC = 49156 based on the payload

*Table 8: Action for SFC = 49156*

|  |  |
| --- | --- |
| **Payload Byte#3 value** | **Action** |
| 0 | Switch off excitation to chip detector inputs |

|  |  |
| --- | --- |
|  | Note: Default value of chip detector input is 1 |
| Other than 0 | Switch on excitation to chip detector inputs |

* + - 1. Setting RCI value

Requirement ID: H398-SRS-DSC-FNC-38 MOPS: No

Safety Requirement: No Rationale: NA

Verification Method: Testing

The Discrete Module shall set its RCI to the value received from Gateway via A825 message (LCC = NSC and SFC = 49152).

* + - 1. Reset Discrete CPU

Requirement ID: H398-SRS-DSC-DRQ-41 MOPS: No

Safety Requirement: No

Rationale: Resetting the discrete board is required to support software verification activities Verification Method: Testing

For any NSC message with SFC = NSC\_RESET (44544), the Discrete Module shall reset the Discrete Module CPU

* + 1. NOC Channel

This section specifies Software High Level Requirements for messages sent over NOC channel of the CAN bus.

* + - 1. NOC message format

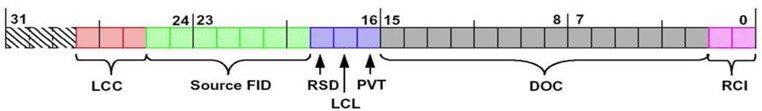
Requirement ID: H398-SRS-DSC-FNC-43 MOPS: No

Safety Requirement: No

Rationale: NA

Verification Method: Testing

The Discrete Module shall use the message structure illustrated in Figure: NOC message format while using the NOC channel



*Figure 4: NOC message format*

Note:

1. LCC - Indicates the Logical Communication Channel
2. Source FID - always 42
3. RSD - Reserved - always 0
4. LCL - messages is designated only for the network the transmitting node resides in - always 1
5. PVT - messages which have no meaning to nodes other than those which are specifically programmed to use them - always 1
6. DOC - Data Object Code
7. RCI - Redundancy Channel Identifier
   * + 1. NOC messages

Requirement ID: H398-SRS-DSC-FNC-44 MOPS: No

Safety Requirement: No Rationale: NA

Verification Method: Testing

The Discrete Module shall use the Document Object Codes specified in Table: List of Document Object Codes in its A825 communication over the Normal Operation Channel:

*Table 9: List of Document Object Codes*

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **DO C** | **Source FID** | **RSD** | **LCL** | **PVT** | **RCI** | **Rx/Tx** | **DLC** | **Description** | **Payload** |
| 0 | 42 | 0 | 1 | 1 | Ref er Not e 3 | Rx | 0 | Gateway to command Discrete Module to send status of discrete inputs and | NA |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  | discrete outputs |  |
| 100 | 42 | 0 | 1 | 1 | Ref er Not e 3 | Tx | 8 | Values of discrete inputs 1  through 64 | Each discrete input is reported as a single bit of the payload. Discrete input number 1 appears in the least significant bit of the 1st payload byte. Discrete input number 64 appears in the most significant bit of the 8th payload byte. |
| 101 | 42 | 0 | 1 | 1 | Ref er Not e 3 | Tx | 6 | Values of discrete inputs 65  through 112 | Each discrete input is reported as a single bit of the payload. Discrete input number 65 appears in the least significant bit of the 1st payload byte. Discrete input number 112 appears in the  most significant bit of the 6th payload byte. |
| 200 | 42 | 0 | 1 | 1 | Ref er Not e 3 | Tx | 4 | Values of chip detector inputs 1  through 32 (Note: This is not applicable as of now and it is retained  for future use) | Each chip detector input is reported as a single bit of the payload. Chip detector input number 1 appears in the least significant bit of the 1st payload byte. Discrete input number 32 appears in the most significant bit of the 4th payload byte. |
| 300 | 42 | 0 | 1 | 1 | Ref er Not e 3 | Tx | 2 | Present values of discrete output 1  through 16  (bit 10 – DeltaNG1\_T hreshold, bit  11 –  DeltaNG2\_T hreshold) | Each discrete output value is represented by two bits of payload. 0 = OFF, 1 = ON, 2  = FLASH, and 3 is reserved. Discrete output number 1 appears in the least significant two bits of the 1st payload byte. Discrete output number 16 appears in the most significant two bits of the  6th payload byte. |
| 750 | 42 | 0 | 1 | 1 | Ref er Not e 3 | Tx | 1 | Value of 28V discrete output current monitor (Note: This is not | 28V discrete output current monitor is represented by single bit of payload.  0 = 28V Discrete Output Fail is False  1 = 28V Discrete Output Fail  is True |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  | applicable as of now and it is retained  for future  use) |  |
| 270  0 | 42 | 0 | 1 | 1 | Ref er Not e 3 | Rx | 2 | Gateway to command Discrete Module to send Application Software, Boot Software, Boot Configuration  , Software Loader and Module Configuration Part Number and CRC | NA |
| 272  2 | 42 | 0 | 1 | 1 | Ref er Not e 3 | Tx | 8 | Discrete Module Boot Software Part number  "H108E-21" | Values of Discrete Module Boot Software Part number first 8 bytes |
| 272  3 | 42 | 0 | 1 | 1 | Ref er Not e 3 | Tx | 8 | Discrete Module Boot Software Part number "7 -x.yy " | Values of Discrete Module Boot Software Part number next 8 bytes  Note: where,  x is major release number yy is minor release number |
| 272  4 | 42 | 0 | 1 | 1 | Ref er Not e 3 | Tx | 8 | Discrete Module Boot Software CRC value | 8 bytes containing  Hexadecimal Discrete Module Boot Software CRC value. |
| 272  5 | 42 | 0 | 1 | 1 | Ref er Not e 3 | Tx | 8 | Discrete Module Application Software Part number  "H108E-80" | Values of Discrete Module Application Software Part number first 8 bytes |
| 272  6 | 42 | 0 | 1 | 1 | Ref er Not e 3 | Tx | 8 | Discrete Module Application Software Part number "9 -x.yy " | Values of Discrete Module Application Software Part number next 8 bytes  Note: where,  x is major release number yy is minor release number |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |  |
| 272  7 | 42 | 0 | 1 | 1 | Ref er Not e 3 | Tx | 8 | Discrete Module Application Software CRC value | 8 bytes containing  Hexadecimal Discrete Module Application Software CRC value. |
| 272  8 | 42 | 0 | 1 | 1 | Ref er Not e 3 | Tx | 8 | Discrete Module Configuration Part number "H108E-67" | Values of Discrete Module Configuration Part number first 8 bytes |
| 272  9 | 42 | 0 | 1 | 1 | Ref er Not e 3 | Tx | 8 | Discrete Module Configuration Software Part number "3 -x.yy " | Values of Discrete Module Configuration Part number next 8 bytes  Note: where,  x is major release number yy is minor release number |
| 273  0 | 42 | 0 | 1 | 1 | Ref er Not e 3 | Tx | 8 | Discrete Module Configuration CRC value | 8 bytes containing  Hexadecimal Discrete Module Configuration CRC value. |
| 273  1 | 42 | 0 | 1 | 1 | Ref er Not e 3 | Tx | 8 | Discrete Module Boot Configuration Part number  "H108E-64" | Values of Discrete Module Boot Configuration Part number first 8 bytes |
| 273  2 | 42 | 0 | 1 | 1 | Ref er Not e 3 | Tx | 8 | Discrete Module Boot Configuration Software Part number "8 -x.yy " | Values of Discrete Module Boot Configuration Part number next 8 bytes  Note: where,  x is major release number yy is minor release number |
| 273  3 | 42 | 0 | 1 | 1 | Ref er Not e 3 | Tx | 8 | Discrete Module Boot Configuration CRC value | 8 bytes containing  Hexadecimal Discrete Module Boot Configuration CRC value. |
| 273  7 | 42 | 0 | 1 | 1 | Ref er Not e 3 | Tx | 8 | Discrete Module Software Loader Part number "  H108E-64" | Values of Discrete Module Software Loader Part number first 8 bytes |
| 273  8 | 42 | 0 | 1 | 1 | Ref er | Tx | 8 | Discrete Module Software Loader Part | Values of Discrete Module Software Loader Part number next 8 bytes  Note: where, |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  | Not e 3 |  |  | number "7 - x.yy " | x is major release number yy is minor release number |
| 273  9 | 42 | 0 | 1 | 1 | Ref er Not e 3 | Tx | 8 | Discrete Module Software Loader CRC value | 8 bytes containing  Hexadecimal Discrete Module Software Loader CRC value. |

Note:

1. Rx/Tx - Indicates whether Discrete module Receives (Rx) or Transmits (Tx) the message
2. DLC - Data Length Count - size of the payload in bytes.
3. When SFC 'NSC\_SET\_RCI' in First two bytes and RCI in Third byte is received from Gateway module, Discrete module will use this RCI for ARINC 825 communication otherwise RCI is considered as 3.
   * + 1. NOC message reply (EDAU Discrete inputs, outputs and chip detector inputs) Requirement ID: H398-SRS-DSC-FNC-45

MOPS: No

Safety Requirement: No Rationale: NA

Verification Method: Testing

When the Discrete Module receives an A825 message with LCC = NOC and DOC = 0 then it shall perform the following:

1. Transmit values of all EDAU Discrete inputs to Gateway via A825 bus using LCC = NOC and DOC =

{100, 101}

1. Transmit values of all EDAU Chip Detector Inputs to Gateway via A825 bus using LCC = NOC and DOC = {200}, if chip excitation command (49156) received from Gateway module atleast once, otherwise return zero.
2. Transmit status of all EDAU Discrete outputs to Gateway via A825 bus using LCC = NOC and DOC =

{300}

Note: Refer H398-SRS-DSC-FNC-27 for discrete inputs, H398-SRS-DSC-FNC-28 for discrete outputs and H398-SRS-DSC-DRQ-47 for chip detector inputs.

* + - 1. NOC message reply (Part number and CRC to Gateway Module) Requirement ID: H398-SRS-DSC-FNC-77

MOPS: No

Safety Requirement: No Rationale: NA

Verification Method: Testing

When the Discrete Module receives an A825 message from Gateway Module with LCC = NOC and DOC

= 2700 then it shall perform the following:

1. Transmit Part number of Discrete Boot Module via A825 bus using LCC = NOC and DOC = {2722, 2723}
2. Transmit CRC of Discrete Boot Module via A825 bus using LCC = NOC and DOC = {2724}
3. Transmit Part number of Discrete Application Module via A825 bus using LCC = NOC and DOC =

{2725, 2726}

1. Transmit CRC of Discrete Application Module via A825 bus using LCC = NOC and DOC = {2727}
2. Transmit Part number of Discrete Configuration Module via A825 bus using LCC = NOC and DOC =

{2728, 2729}

1. Transmit CRC of Discrete Configuration Module via A825 bus using LCC = NOC and DOC = {2730}
2. Transmit Part number of Discrete Boot Configuration Data via A825 bus using LCC = NOC and DOC

= {2731, 2732}

1. Transmit CRC of Discrete Boot Configuration Data via A825 bus using LCC = NOC and DOC = {2733}
2. Transmit Part number of Discrete Software Loader Module via A825 bus using LCC = NOC and DOC

= {2737, 2738}

1. Transmit CRC of Discrete Software Loader Module via A825 bus using LCC = NOC and DOC ={2739}
   1. Chip Detection

This section specifies the Software High Level Requirements on the Chip Detection functionality.

* + 1. Chip Detector Inputs

Requirement ID: H398-SRS-DSC-DRQ-47 MOPS: No

Safety Requirement: No

Rationale: Required to define chip detectors Verification Method: Testing

The Discrete Module shall read and process the Chip Detector inputs mentioned in Table: Chip Detector Inputs every 100ms

*Table 10*: Chip Detector Inputs

|  |  |  |
| --- | --- | --- |
| Chip Detector Output | Signal Name | Debounce (ms) |
| Chip Detector Input #1 (CD150\_1) | Engine 1 Chip Detect | 2800 |
| Chip Detector Input #2 (CD150\_2) | Engine 2 Chip Detect | 2800 |
| Chip Detector Input #3 (CD150\_3) | C-BOX Chip Detect | 2800 |

|  |  |  |
| --- | --- | --- |
| Chip Detector Input #4 (CD150\_4) | Mast Chip Detect | 2800 |
| Chip Detector Input #5 (CD150\_5) | T/R 90 Deg. GB Chip Detect | 2800 |
| Chip Detector Input #6 (CD150\_6) | T/R 42 Deg. GB Chip Detect | 2800 |
| Chip Detector Input #7 (CD150\_7) | Monitor Chip Detect | 2800 |
| Chip Detector Input #8 (CD150\_8) | Planetary Chip Detect | 2800 |
| Chip Detector Input #9 (CD150\_9) | Engine 2 RGB Chip Detect | 2800 |
| Chip Detector Input #10 (CD150\_10) | Engine 1 RGB Chip Detect | 2800 |
| Chip Detector Input #11 (CD150\_11) | Sump Chip Detect | 2800 |
| Chip Detector Input #12 (CD150\_12) | Spare | 2800 |

## Module Configuration Data

This section provides the Software High Level Requirements for the Module Configuration Data of the Discrete Module.

* + 1. Module Configuration Data - Definition

Requirement ID: H398-SRS-DSC-DRQ-49 MOPS: No

Safety Requirement: No

Rationale: Required to define the various sections of Module Configuration Data used to make the software configurable.

Verification Method: Testing

The Module Configuration Data for the Discrete Module shall be a binary and have the sections specified in Table: Module Configuration Data - sections

*Table 11: Module Configuration Data - sections*

|  |
| --- |
| **Section** |
| MCD Software Part Number (Refer section 16.9.2) |
| On time in milliseconds for flashing discrete output. |
| Off time in milliseconds for flashing discrete output. |
| Configuration for Chip detector inputs. |
| Configuration for discrete outputs. |
| Cyclic Redundancy Check for MCD |

Note: The CRC field is filled during build process.

* + 1. Module Configuration Data format - Chip detector inputs

Requirement ID: H398-SRS-DSC-DRQ-50 MOPS: No

Safety Requirement: No

Rationale: Required to define the Chip detector inputs section of Module Configuration Data Verification Method: Testing

The Module Configuration Data for Chip detector inputs of Discrete Module shall have the format specified in Table: Module Configuration Data Format

*Table 12: Module Configuration Data Format*

|  |  |  |
| --- | --- | --- |
| **Chip Detector** | **Channel\_state** | **Chip\_debounce** |
| 1 | <value>  1 – Enabled  0 – Disabled | 28 (2800ms) |
| 2 | <value>  1 – Enabled  0 – Disabled | 28 (2800ms) |
| 3 | <value>  1 – Enabled  0 – Disabled | 28 (2800ms) |
| …..  …..  …. | …..  …..  …. | …..  …..  …. |
| 31 | <value>  1 – Enabled  0 – Disabled | 28 (2800ms) |
| 32 | <value>  1 – Enabled  0 – Disabled | 28 (2800ms) |

Note:

1. The first column will not be part of the MCD table. It is illustrated here to identify the columns.
2. The second column represents channel state (ENABLE(1) / DISABLE(0))
3. The third column represents debounce period for each chip detector input
   * 1. Module Configuration Data format - Discrete Outputs Requirement ID: H398-SRS-DSC-DRQ-51

MOPS: No

Safety Requirement: No

Rationale: Required to define the discrete outputs section of Module Configuration Data Verification Method: Testing

The Module Configuration Data for discrete outputs of Discrete Module shall have the format specified in Table: Module Configuration Data for discrete outputs

*Table 13: Module Configuration Data for discrete outputs*

|  |  |  |
| --- | --- | --- |
| **Channel** | **Discrete\_Output\_State** | **Range** |
| 1 | <value> | 1. – OFF 2. – ON 3. – FLASH |
| 2 | <value> | 1. – OFF 2. – ON 3. – FLASH |
| 3 | <value> | 1. – OFF 2. – ON 3. – FLASH |
| …..  …..  …. | …..  …..  …. | …..  …..  …. |
| 23 | <value> | 1. – OFF 2. – ON 3. – FLASH |
| 24 | <value> | 1. – OFF 2. – ON 3. – FLASH |

## BIT

This section provides the Software High Level Requirements for the Built In Tests performed by the Discrete Module. i.e

1. Power on Built-In Test
2. Continuous Built-In Test
   * 1. PBIT

Requirement ID: H398-SRS-DSC-FNC-53 MOPS: No

Safety Requirement: No Rationale: NA

Verification Method: Testing

The Discrete Module shall perform the following tests as part of PBIT after power-on:

1. Test CPU and RAM
2. CRC of the discrete module and Module Configuration Data
3. Validity Test on Module Configuration Data (MCD)
   * + 1. PBIT fault indication

Requirement ID: H398-SRS-DSC-FNC-54 MOPS: No

Safety Requirement: No Rationale: NA

Verification Method: Testing

The Discrete Module shall enter into error mode for any one of the following during PBIT:

1. CPU and RAM Test Failure as defined in H398-SRS-DSC-FNC-55
2. CRC Test Failure as defined in H398-SRS-DSC-FNC-56
3. MCD Validity Test Failure as defined in H398-SRS-DSC-DRQ-57
4. Failure in creation of task by OS kernel as defined in H398-SRS-DSC-DRQ-64.
   * + 1. CPU and RAM Test

Requirement ID: H398-SRS-DSC-FNC-55 MOPS: No

Safety Requirement: No Rationale: NA

Verification Method: Testing

The Discrete Module shall perform PBIT as follows:

1. Perform the following operations with known operands

Arithmetic Operators - addition, subtraction, multiplication, division Logical Operators - logical AND (&&), Logical OR ( || ) and exclusive OR

1. Test Random Access Memory (RAM)

Note: Flags - Carry, Overflow, Zero flags are also tested while performing the above operations.

* + - 1. CRC Test

Requirement ID: H398-SRS-DSC-FNC-56 MOPS: No

Safety Requirement: No Rationale: NA

Verification Method: Testing

The Discrete Module shall perform the CRC test as follows:

1. Calculate the CRC over the required memory region. (CRC polynomial is X^32+X^26+X^23+X^22+X^16+X^12+X^11+X^10+X^8+X^7+X^5+X^4+X^2+X^1+X^0)
2. Compare the calculated CRC with the CRC present in the memory.Refer Note 2
3. If both values match then CRC test is PASS else FAIL

Note :

1. Discrete Module perform the CRC test on Module configuration data region and Application software region.
2. Module configuration data CRC is present at 0x80BFFFC, Application Software CRC is present at 0x803FFFC
   * + 1. Module Configuration Data - Validity Test

Requirement ID: H398-SRS-DSC-DRQ-57 MOPS: No

Safety Requirement: No

Rationale: Derived to define the validity check of the Module Configuration Data Verification Method: Testing

The Discrete Module shall verify the parameters of the Module Configuration Data are within the ranges for Discrete output configuration states, Chip Detector Channel States and Chip Detector Debounce Time during PBIT.

* + 1. CBIT

Requirement ID: H398-SRS-DSC-FNC-58 MOPS: No

Safety Requirement: No Rationale: NA

Verification Method: Testing

As part of CBIT the Discrete Module shall perform the following tests:

1. Stack Test of the following tasks Init Task

Application Task ARINC 825 comm task. CBIT Task

Idle task

* + - 1. CBIT fault indication

Requirement ID: H398-SRS-DSC-FNC-59 MOPS: No

Safety Requirement: No Rationale: NA

Verification Method: Testing

The Discrete Module shall enter into error mode for any one of the following during CBIT:

1. Stack test failure
2. Detection of NonMaskable interrupt, HardFault interrupt, MemManage interrupt, BusFault interrupt SpuriousInterrupt interrupt and UsageFault interrupt.
   * + 1. Stack Test

Requirement ID: H398-SRS-DSC-FNC-60 MOPS: No

Safety Requirement: No Rationale: NA

Verification Method: Testing

The Discrete shall perform the Stack test as follows:

1. Read the location at 70% of the stack length
2. If the value read is not equal to 0xDEADDEAD then the test is failed else it is passed Note: Stack is initially filled with 0xDEADDEAD

## Kernel and Scheduler

This section specifies the Software High Level Requirements for the kernel and scheduler used in the Discrete Module.

* + 1. Kernel/Scheduler - Initializations

Requirement ID: H398-SRS-DSC-DRQ-62 MOPS: No

Safety Requirement: No

Rationale: This requirement has been derived to ensure proper initialisation so that all tasks are functioning correctly

Verification Method: Testing

The Discrete Module shall initialize the data structures used by the kernel and scheduler after PBIT.

* + 1. Scheduler - Algorithm

Requirement ID: H398-SRS-DSC-DRQ-63 MOPS: No

Safety Requirement: No

Rationale: This requirement has been derived to ensure that all tasks are functioning correctly Verification Method: Testing

The Discrete Module shall use a priority based pre-emptive scheduler.

* + 1. Tasks

Requirement ID: H398-SRS-DSC-DRQ-64 MOPS: No

Safety Requirement: No

Rationale: This requirement has been derived to differentiate the activities that are to be performed by different tasks

Verification Method: Testing

The Discrete Module shall create the tasks described in Table: Tasks in Discrete Module during initialization:

*Table 14: Tasks in Discrete Module*

|  |  |  |  |
| --- | --- | --- | --- |
| **Tasks** | **Frequency** | **Priority** | **Description** |
| ARINC825  communication Task | Invoked by ISR | 6 | This task initializes to handle commands coming across the ARINC 825 bus. |
| Application Task | 100ms | 5 | This is the main task of the Discrete Module. It performs processing of discrete inputs, chip detector inputs, 28v discrete output current monitors and setting  discrete outputs. |
| Init Task | 500ms | 7 | This task initializes the peripherals, scheduler, kernel and creates other tasks. |
| Idle Task | - | 63 | It is executed when there are no other Ready tasks. |
| CBIT Task | 600ms | 8 | It performs stack test for all created Tasks. |

* + 1. Semaphores

Requirement ID: H398-SRS-DSC-DRQ-65 MOPS: No

Safety Requirement: No

Rationale: This requirement has been derived to specify the mechanism used for task activation Verification Method: Testing

The Discrete Module shall use separate counting semaphore for each task to activate the task.

* + 1. Task activation by semaphore

Requirement ID: H398-SRS-DSC-DRQ-66 MOPS: No

Safety Requirement: No

Rationale: This requirement has been derived to specify the mechanism used for task activation Verification Method: Testing

The Discrete Module shall use the System Timer interrupt to release the semaphore for tasks as per the timing mentioned in H398-SRS-DSC-DRQ-64.

* + 1. Task structure

Requirement ID: H398-SRS-DSC-DRQ-67 MOPS: No

Safety Requirement: No

Rationale: This has been derived to ensure that all tasks have a common structure for executing their defined activities

Verification Method: Testing

The tasks in Discrete Module shall behave as per the following steps:

1. Wait for the semaphore
2. On semaphore release, perform the activities once
3. Infinitely repeat Steps#1 and #2.

Note:

* 1. The Idle task is an exception. It is activated when there are no other tasks which are ready.
  2. Semphore for ARINC825 communication Task is released upon reception of ARINC 825 data triggered by ISR.

## Modes of Operation

This section specifies the requirements for the operational modes of the Discrete Module.

* + 1. Operational Modes

The Discrete Module operates in different mode to perform its various responsibilities. The different modes are

1. Power-On BIT mode
2. Normal Mode
3. Error Mode
   * 1. PBIT Mode

Requirement ID: H398-SRS-DSC-DRQ-70 MOPS: No

Safety Requirement: No

Rationale: This is required to identify the PBIT mode of application software Verification Method: Testing

The Discrete Module shall perform the PBIT tests during PBIT Mode.

* + 1. Normal Mode

Requirement ID: H398-SRS-DSC-DRQ-71 MOPS: No

Safety Requirement: No

Rationale: This is required to identify the normal mode of application software Verification Method: Testing

The Discrete Module shall perform the following activities in this mode:

1. Process Discrete Inputs & Chip Detector Inputs
2. Set Discrete Outputs
3. Perform A825 communication with Gateway module
4. Perform CBIT
   * 1. Error Mode

Requirement ID: H398-SRS-DSC-DRQ-72 MOPS: No

Safety Requirement: No

Rationale: This is required to identify the error mode of application software Verification Method: Testing

In this mode, the Discrete Module shall perform the following

1. cease all current activities
2. switch the Heartbeat LED as steady on.

*Table 15: Discrete Module - LED Blink Rate*

|  |  |  |
| --- | --- | --- |
| **Error Type** | **Heartbeat LED state** | **Description** |
| Application CRC Error | Steady On | Mismatch in the computed and Stored Flight Application CRC value. |
| MCD CRC Error | Steady On | Mismatch in the computed and Stored MCD Application CRC value. |
| RAM Error | Steady On | Failure of RAM test in PBIT. |
| CPU Error | Steady On | Failure in HSE start-up OR Processor Test Failure during PBIT. |
| MCD Error | Steady On | Error in configuring the Module Configuration data (MCD). |
| Stack Error | Steady On | Failure of Stack Test during CBIT. |
| OS Kernel Error | Steady On | Failure in Creating a Task. |
| INTR Error | Steady On | Failure due to following Fault interrupts: NonMaskable, HardFault, MemManage, BusFault, SpuriousInterrupt and UsageFault. |

## Software Part Number

The Discrete Module consists of two binaries. Each binary has a unique part number to identify itself.

* + 1. Software Part Number – Application Software Requirement ID: H398-SRS-DSC-DRQ-74

MOPS: No

Safety Requirement: No

Rationale: Required to define the Software Part Number to identify the Application Software Verification Method: Testing

The Discrete Module shall define a 16 character string at a prefixed location in memory (0x0803FFEC) to identify its Software Part Number.

Format: <H108E-809><space><-><x><.><yy><space> where,

x is major release number yy is minor release number

* + 1. Software Part Number – Module Configuration Data Binary Requirement ID: H398-SRS-DSC-DRQ-75

MOPS: No

Safety Requirement: No

Rationale: Required to define the Software Part Number to identify the MCD Software Verification Method: Testing

The Discrete Module shall define a 16-character string at a prefixed location in memory (0x080A0000) to identify its Module Configuration Data (MCD) Software Part Number.

Format:

<H108E-673><space><-><x><.><yy><space> where,

x is major release number yy is minor release number

## Resource Utilization

This section specifies the Software High Level Requirements for the resource utilization of the Discrete Module.

* + 1. Memory Usage

Requirement ID: H398-SRS-DSC-DRQ-79 MOPS: No

Safety Requirement: No

Rationale: This margin is required for future enhancements Verification Method: Analysis

The Discrete Module software shall not utilize more than 80% of the available FLASH and RAM memory for the initial certification of the product.

* + 1. Throughput Usage

Requirement ID: H398-SRS-DSC-DRQ-80 MOPS: No

Safety Requirement: No

Rationale: This margin is required for future enhancements Verification Method: Analysis

The Discrete Module software shall not utilize more than 80% of the available processor throughput for the initial certification of the product.