Software Low Level Requirements for Gateway Module of Engine Data Acquisition Unit of Airbus Helicopter Generic Vehicle Monitoring System and CMU+

Document Number: H398-003-011-GWY

Version No: 2.13

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: | Afreen P(Software Engineer, ALTEN Global Technologies Private Limited) |
| Reviewed By: | Sruthi D (Lead Engineer, ALTEN Global Technologies Private Limited)  C:\Users\afreen.p.SWSYS\AppData\Local\Microsoft\Windows\INetCache\Content.MSO\A65C1828.tmp  Pravalika K (Quality Engineer – Quality Assurance, ALTEN Global Technologies Private Limited) |
| Approved By: | Devarani R (Manager - Quality Assurance and Management, ALTEN Global Technologies Private Limited) |
| Released By: | Madhuchandra Srinivas (Associate Director – Software, ALTEN Global Technologies **Private Limited**)  C:\Users\afreen.p.SWSYS\Pictures\SIGNATURE.PNG |

A logo with a triangle and a triangle in a red yellow and blue rectangle

Description automatically generated

ALTEN Global Technologies ****Private Limited****

Table of Contents

[1 Amendment Record 13](#_Toc210985650)

[2 Objective 51](#_Toc210985651)

[3 Scope 51](#_Toc210985652)

[4 References 51](#_Toc210985653)

[5 Acronyms and Definitions 52](#_Toc210985654)

[6 Document Control 55](#_Toc210985655)

[7 Responsibilities 55](#_Toc210985656)

[8 Distributions 56](#_Toc210985657)

[9 Traceability 56](#_Toc210985658)

[10 Software Low Level Requirements- Gateway Module Application Software 56](#_Toc210985659)

[10.1 daugwya429 56](#_Toc210985660)

[10.1.1 A429StatusMiscompare 56](#_Toc210985661)

[10.1.2 A429OutTask 58](#_Toc210985662)

[10.1.3 LabelPriority 63](#_Toc210985663)

[10.1.4 ProcessLabel 68](#_Toc210985664)

[10.1.5 ProcessArincWord 73](#_Toc210985665)

[10.1.6 ReceiveArinc429 84](#_Toc210985666)

[10.1.7 TransmitArinc429 87](#_Toc210985667)

[10.1.8 A429Task 93](#_Toc210985668)

[10.1.9 InitChannels 103](#_Toc210985669)

[10.1.10 Inita429Parameters 105](#_Toc210985670)

[10.1.11 InitTimmingA429 108](#_Toc210985671)

[10.1.12 A429Init 111](#_Toc210985672)

[10.1.13 CalcArincWordChecksum 114](#_Toc210985673)

[10.1.13.6 Other CSUs called by this CSU 115](#_Toc210985674)

[10.2 daugwya825comm 116](#_Toc210985675)

[10.2.1 A8251CommTask 117](#_Toc210985676)

[10.2.2 A825CommInit 120](#_Toc210985677)

[10.3 daugwya825comm2 123](#_Toc210985678)

[10.3.1 A8251CommTask2 123](#_Toc210985679)

[10.3.2 A825CommInit2 126](#_Toc210985680)

[10.4 daugwyapp 129](#_Toc210985681)

[10.4.1 DoMirrorCheck 129](#_Toc210985682)

[10.4.2 AppInit 132](#_Toc210985683)

[10.4.3 AppU32Noc 134](#_Toc210985684)

[10.4.4 AppU32Nsc 144](#_Toc210985685)

[10.4.5 ClearLog 150](#_Toc210985686)

[10.4.6 WragleFilter 152](#_Toc210985687)

[10.4.9 SendLog 155](#_Toc210985688)

[10.4.10 AppSet825P2PMessage 158](#_Toc210985689)

[10.4.11 AppPow 160](#_Toc210985690)

[10.4.12 AppTask 162](#_Toc210985691)

[10.4.13 DoInit 168](#_Toc210985692)

[10.4.14 DoBit 170](#_Toc210985693)

[10.4.15 DoAcquire 176](#_Toc210985694)

[10.4.16 DoAcquireCrcPn 178](#_Toc210985695)

[10.4.17 DoAcquireCmuNvram 180](#_Toc210985696)

[10.4.18 DoAppUpdate 182](#_Toc210985697)

[10.4.24 ExciteStatus 185](#_Toc210985698)

[10.4.25 AppExciteConditions 188](#_Toc210985699)

[10.4.26 DoNormal 189](#_Toc210985700)

[10.4.27 CheckMissing 194](#_Toc210985701)

[10.4.28 AppGetAnalog 197](#_Toc210985702)

[10.4.29 DoCheckup 201](#_Toc210985703)

[10.4.30 DoExcite 203](#_Toc210985704)

[10.4.31 AppExciteOff 208](#_Toc210985705)

[10.4.32 AppExciteON 210](#_Toc210985706)

[10.4.33 DoError 213](#_Toc210985707)

[10.4.34 DoCmDownload 215](#_Toc210985708)

[10.4.35 ReCorrctPointers 220](#_Toc210985709)

[10.4.36 CorrctPointers 223](#_Toc210985710)

[10.4.37 ProcessEngineNoCMU 226](#_Toc210985711)

[10.4.38 ProcessExciteNoCMU 227](#_Toc210985712)

[10.4.39 ProcessEngine 229](#_Toc210985713)

[10.4.40 ProcessExcite 231](#_Toc210985714)

[10.4.41 ReconfigurationFromDU 233](#_Toc210985715)

[10.5 daugwyapplogic 243](#_Toc210985716)

[10.5.1 TransmitGwyPartNums 243](#_Toc210985717)

[10.5.2 TransmitAnalogPartNums 254](#_Toc210985718)

[10.5.3 TransmitDiscretePartNums 265](#_Toc210985719)

[10.5.4 TransmitSystemCheckSum 275](#_Toc210985720)

[10.5.5 PacketizedDataToDisplay 276](#_Toc210985721)

[10.5.6 DateTimeCmp 279](#_Toc210985722)

[10.5.7 DoNvmCRCcheck 283](#_Toc210985723)

[10.5.8 APPu32NOCEx 285](#_Toc210985724)

[10.5.9 TransmitFaultstoCmu 290](#_Toc210985725)

[10.5.10 TransmitA429toCMU 292](#_Toc210985726)

[10.5.11 InputLabelTx 297](#_Toc210985727)

[10.5.12 TxBNRLabelsToCMU 299](#_Toc210985728)

[10.5.13 TxDDWLabelsToCMU 301](#_Toc210985729)

[10.5.14 RxLabelsToCMU 303](#_Toc210985730)

[10.5.15 NonLabelParametersToCMU 306](#_Toc210985731)

[10.6 daugwyarinc825 308](#_Toc210985732)

[10.6.1 RxQueueRemove 308](#_Toc210985733)

[10.6.2 A825GetMessage 314](#_Toc210985734)

[10.6.3 RxQueueInsert 317](#_Toc210985735)

[10.6.4 A825ReceiveMessage 320](#_Toc210985736)

[10.6.5 TxQueueInsert 324](#_Toc210985737)

[10.6.6 BuildBroadcastExtId 328](#_Toc210985738)

[10.6.7 BuildPeerToPeerExtId 330](#_Toc210985739)

[10.6.8 A825Xmit 332](#_Toc210985740)

[10.6.9 XmitFromBuff 336](#_Toc210985741)

[10.6.10 A825BufferCheck 338](#_Toc210985742)

[10.6.11 CanConfig 341](#_Toc210985743)

[10.6.12 NvicConfig 346](#_Toc210985744)

[10.6.13 A825Init 349](#_Toc210985745)

[10.7 daugwyarinc8252 351](#_Toc210985746)

[10.7.1 RxQueueRemove2 351](#_Toc210985747)

[10.7.2 A825GetMessage2 356](#_Toc210985748)

[10.7.3 RxQueueInsert2 358](#_Toc210985749)

[10.7.4 A825ReceiveMessage2 361](#_Toc210985750)

[10.7.5 TxQueueInsert2 365](#_Toc210985751)

[10.7.6 BuildBroadcastExtId2 368](#_Toc210985752)

[10.7.7 BuildPeerToPeerExtId2 370](#_Toc210985753)

[10.7.8 A825Xmit2 373](#_Toc210985754)

[10.7.9 XmitFromBuff2 376](#_Toc210985755)

[10.7.10 A825BufferCheck2 378](#_Toc210985756)

[10.7.11 CanConfig2 381](#_Toc210985757)

[10.7.12 NvicConfig2 385](#_Toc210985758)

[10.7.13 A825Init2 388](#_Toc210985759)

[10.8 daugwybittest 390](#_Toc210985760)

[10.8.1 RamTest 390](#_Toc210985761)

[10.8.2 ProcessorTest 393](#_Toc210985762)

[10.8.3 CrcCheck 395](#_Toc210985763)

[10.8.4 XRamTest 398](#_Toc210985764)

[10.8.5 PbitCheck 401](#_Toc210985765)

[10.9 daugwybittesta 403](#_Toc210985766)

[10.9.1 AluTest 403](#_Toc210985767)

[10.10 daugwycbit 407](#_Toc210985768)

[10.10.1 StackTest 408](#_Toc210985769)

[10.10.2 ContinuousBitTask 410](#_Toc210985770)

[10.10.3 ContinuousBitTaskInit 412](#_Toc210985771)

[10.11 daugwycomm 415](#_Toc210985772)

[10.11.1 CommPutChars 415](#_Toc210985773)

[10.11.2 CommGetChars 417](#_Toc210985774)

[10.12 daugwycrc 420](#_Toc210985775)

[10.12.1 UpdCrc32 420](#_Toc210985776)

[10.13 daugwycrt0 422](#_Toc210985777)

[10.13.1 Crt0TransferData 422](#_Toc210985778)

[10.14 daugwydtoa 425](#_Toc210985779)

[10.14.1 DtoaHwInit 425](#_Toc210985780)

[10.15 daugwyetitimer 430](#_Toc210985781)

[10.15.1 ETIinit 430](#_Toc210985782)

[10.15.2 TimerTask 433](#_Toc210985783)

[10.16 daugwyexmon 435](#_Toc210985784)

[10.16.1 CheckCondition 435](#_Toc210985785)

[10.16.2 CheckRange 440](#_Toc210985786)

[10.16.3 Chklimit 443](#_Toc210985787)

[10.16.4 ValidateParam 449](#_Toc210985788)

[10.16.5 ExceedancePacket 452](#_Toc210985789)

[10.16.6 ExceedanceToCMU 454](#_Toc210985790)

[10.16.7 Limmon 459](#_Toc210985791)

[10.16.8 SetMonLimits 467](#_Toc210985792)

[10.17 daugwyhw 471](#_Toc210985793)

[10.17.1 InitIO 471](#_Toc210985794)

[10.17.2 HwInit 479](#_Toc210985795)

[10.17.3 HwCopy 488](#_Toc210985796)

[10.17.4 HwMemset 490](#_Toc210985797)

[10.17.5 HwAbs32 492](#_Toc210985798)

[10.18 daugwyiadc 494](#_Toc210985799)

[10.18.1 IadcRead 494](#_Toc210985800)

[10.18.2 TransferCompleteIntr 496](#_Toc210985801)

[10.18.3 ConfigureDma 498](#_Toc210985802)

[10.18.4 TriggerInternalAdc 502](#_Toc210985803)

[10.18.5 IadcInit 504](#_Toc210985804)

[10.19 daugwyinit 509](#_Toc210985805)

[10.19.1 InitTask 509](#_Toc210985806)

[10.19.2 InitInit 515](#_Toc210985807)

[10.20 daugwyintr 518](#_Toc210985808)

[10.20.1 IntrInit 518](#_Toc210985809)

[10.20.2 IntrInstall 520](#_Toc210985810)

[10.20.3 IsrReset 523](#_Toc210985811)

[10.20.4 NonMaskable 525](#_Toc210985812)

[10.20.5 HardFault 527](#_Toc210985813)

[10.20.6 MemManage 529](#_Toc210985814)

[10.20.7 BusFault 531](#_Toc210985815)

[10.20.8 UsageFault 533](#_Toc210985816)

[10.20.9 SpuriousInterrupt 535](#_Toc210985817)

[10.21 daugwylogic 537](#_Toc210985818)

[10.21.1 CheckMinMax 537](#_Toc210985819)

[10.21.2 HwCalibrationOffset 539](#_Toc210985820)

[10.21.3 EngData 543](#_Toc210985821)

[10.21.4 AircraftData 548](#_Toc210985822)

[10.21.5 LogicCustApp 552](#_Toc210985823)

[10.21.6 MessageAvailable 554](#_Toc210985824)

[10.21.8 TransmitMessage 563](#_Toc210985825)

[10.21.13 CrossChannelCmpDis 569](#_Toc210985826)

[10.21.16 CrossChannelComparison 572](#_Toc210985827)

[10.21.17 LogicRS232 576](#_Toc210985828)

[10.21.19 LogicTask 580](#_Toc210985829)

[10.21.20 LogicTaskcom4 583](#_Toc210985830)

[10.21.21 LogicInit 587](#_Toc210985831)

[10.21.22 CrossChannelEssential 589](#_Toc210985832)

[10.21.23 CrossChannelCritical 594](#_Toc210985833)

[10.21.24 CrossChannelAirbusAIF 598](#_Toc210985834)

[10.21.25 DiscreteFilter 601](#_Toc210985835)

[10.22 daugwylogic2 603](#_Toc210985836)

[10.22.2 ProcessAllLogics 604](#_Toc210985837)

[10.22.5 DUDiscreteInputsStatus 606](#_Toc210985838)

[10.22.11 CalcSysChecksum 608](#_Toc210985839)

[10.25 daugwylookup 609](#_Toc210985840)

[10.25.1 LookUpTableLookUp 610](#_Toc210985841)

[10.26 daugwymain 617](#_Toc210985842)

[10.26.1 MainFunc 617](#_Toc210985843)

[10.27 daugwymontab 620](#_Toc210985844)

[10.27.1 Brief Description 620](#_Toc210985845)

[10.27.2 List of HLRs allocated 620](#_Toc210985846)

[10.27.3 List of global variables accessed and modified 620](#_Toc210985847)

[10.27.4 Parameter list (Input/Output) 620](#_Toc210985848)

[10.27.5 Return Value 620](#_Toc210985849)

[10.27.6 Other CSUs called by this CSU 621](#_Toc210985850)

[10.27.7 Description of list of LLRs allocated 621](#_Toc210985851)

[10.28 daugwyoscpu 621](#_Toc210985852)

[10.28.1 OsTaskCreate 621](#_Toc210985853)

[10.29 daugwyoscpua 625](#_Toc210985854)

[10.29.1 OsCtxSw 625](#_Toc210985855)

[10.29.2 OsIntCtxSw 626](#_Toc210985856)

[10.29.3 SaveStatusReg 628](#_Toc210985857)

[10.29.4 RestoreStatusReg 629](#_Toc210985858)

[10.29.5 OsStartHighRdy 630](#_Toc210985859)

[10.29.6 PendSvHandler 633](#_Toc210985860)

[10.30 daugwyqueue 636](#_Toc210985861)

[10.30.1 InitQueue 636](#_Toc210985862)

[10.30.2 IsEmpty 637](#_Toc210985863)

[10.30.3 IsFull 639](#_Toc210985864)

[10.30.4 EnQueue 640](#_Toc210985865)

[10.30.5 DeQueue 642](#_Toc210985866)

[10.31 daugwyrevno 643](#_Toc210985867)

[10.31.1 Brief Description 644](#_Toc210985868)

[10.31.2 List of HLRs allocated 644](#_Toc210985869)

[10.31.3 List of global variables accessed and modified 644](#_Toc210985870)

[10.31.4 Parameter list (Input/Output) 644](#_Toc210985871)

[10.31.5 Return Value 644](#_Toc210985872)

[10.31.6 Other CSUs called by this CSU 644](#_Toc210985873)

[10.31.7 Description of list of LLRs allocated 644](#_Toc210985874)

[10.32 daugwyrs232 645](#_Toc210985875)

[10.32.1 TxEnable 645](#_Toc210985876)

[10.32.2 Rs232TxBlock 647](#_Toc210985877)

[10.32.3 Rs232RxBlock 650](#_Toc210985878)

[10.32.4 InitComData 652](#_Toc210985879)

[10.32.5 TransmitChar 654](#_Toc210985880)

[10.32.6 ResetCOM 656](#_Toc210985881)

[10.32.7 DisableTxIntr 658](#_Toc210985882)

[10.32.8 ReceiveChar 659](#_Toc210985883)

[10.32.9 DisableRxIntr 661](#_Toc210985884)

[10.32.10 Rs232TxIsr 663](#_Toc210985885)

[10.32.11 Rs232RxIsr 665](#_Toc210985886)

[10.32.12 COM2Intr 667](#_Toc210985887)

[10.32.13 InitCOM2 669](#_Toc210985888)

[10.32.14 COM4Intr 673](#_Toc210985889)

[10.32.15 InitCOM4 676](#_Toc210985890)

[10.32.16 Rs232Init 680](#_Toc210985891)

[10.33 daugwyrterr 683](#_Toc210985892)

[10.33.1 RterrError 683](#_Toc210985893)

[10.33.2 RterrForever 689](#_Toc210985894)

[10.33.3 CheckValidErrorFlash 690](#_Toc210985895)

[10.33.4 UpdateNVRam 692](#_Toc210985896)

[10.33.5 WriteErrorToFlash 694](#_Toc210985897)

[10.33.6 EraseErrSection 698](#_Toc210985898)

[10.33.7 ProgramWord 701](#_Toc210985899)

[10.34 daugwysound 703](#_Toc210985900)

[10.34.1 SoundSynthesize 703](#_Toc210985901)

[10.34.2 AudioTest 706](#_Toc210985902)

[10.34.3 SoundTask 718](#_Toc210985903)

[10.34.4 SoundInit 724](#_Toc210985904)

[10.35 daugwytach 726](#_Toc210985905)

[10.35.1 TachRead 726](#_Toc210985906)

[10.35.2 TachDriver 728](#_Toc210985907)

[10.35.3 TachInit 732](#_Toc210985908)

[10.36 daugwytbase 735](#_Toc210985909)

[10.36.1 TbaseTaskSignaling 736](#_Toc210985910)

[10.36.2 TbaseIntrHandler 738](#_Toc210985911)

[10.36.3 TbaseInit 740](#_Toc210985912)

[10.37 daugwytmr 741](#_Toc210985913)

[10.37.1 Timer7Intr 741](#_Toc210985914)

[10.37.2 TmrInitDac 743](#_Toc210985915)

[10.37.3 TmrInitTim6 746](#_Toc210985916)

[10.37.4 TmrInitTim12 748](#_Toc210985917)

[10.37.5 GetDivConfig 751](#_Toc210985918)

[10.37.6 TmrInitTach1 753](#_Toc210985919)

[10.37.7 TmrInitTach2 757](#_Toc210985920)

[10.37.8 TmrInitTach3 762](#_Toc210985921)

[10.37.9 TmrInitTach4 766](#_Toc210985922)

[10.37.10 TmrInitTach5 770](#_Toc210985923)

[10.38 daugwyucos 775](#_Toc210985924)

[10.38.1 OsInit 775](#_Toc210985925)

[10.38.2 OsTaskIdle 777](#_Toc210985926)

[10.38.3 OsStart 779](#_Toc210985927)

[10.38.4 OsSched 782](#_Toc210985928)

[10.38.5 OsTcbInit 784](#_Toc210985929)

[10.38.6 OsIntEnter 788](#_Toc210985930)

[10.38.7 OsIntExit 789](#_Toc210985931)

[10.38.8 OsTimeDly 792](#_Toc210985932)

[10.38.9 OsTimeTick 794](#_Toc210985933)

[10.38.10 OsSemCreate 797](#_Toc210985934)

[10.38.11 OsSemPend 799](#_Toc210985935)

[10.38.12 OsSemPost 802](#_Toc210985936)

[10.39 daugwywdog 805](#_Toc210985937)

[10.39.1 WdogInit 805](#_Toc210985938)

[10.39.2 WdogKickWatchDog 808](#_Toc210985939)

[10.40 daugwyxram 810](#_Toc210985940)

[10.40.1 XramInit 810](#_Toc210985941)

[10.41 daugwylogicairbus 812](#_Toc210985942)

[10.41.1 EngineStateLogic 812](#_Toc210985943)

[10.41.2 CompositeEngineStateLogic 816](#_Toc210985944)

[10.41.3 EngineOEIStateLogic 822](#_Toc210985945)

[10.41.4 DiscreteSignals 824](#_Toc210985946)

[10.41.5 GroundState 826](#_Toc210985947)

[10.41.6 ExternalInterfaces 827](#_Toc210985948)

[10.41.7 BleedValue 832](#_Toc210985949)

[10.41.8 EngineState 835](#_Toc210985950)

[10.41.9 EngineChip 838](#_Toc210985951)

[10.41.10 MGBOil 839](#_Toc210985952)

[10.41.11 MGBChip 841](#_Toc210985953)

[10.41.12 IGBOilTemp 843](#_Toc210985954)

[10.41.13 TBGOilTemp 844](#_Toc210985955)

[10.41.14 HydraulicCautions 846](#_Toc210985956)

[10.41.15 TxProcessAirbusLabel 849](#_Toc210985957)

[10.41.16 ConvertRange 852](#_Toc210985958)

[10.41.17 TRQ12DisplayClass 853](#_Toc210985959)

[10.41.18 CalculateEmbeddedResolution 856](#_Toc210985960)

[10.41.20 TxProcessLabel034SysMsgStat 858](#_Toc210985961)

[10.41.21 TxProcessLabel050TRQ 863](#_Toc210985962)

[10.41.22 TxProcessLabel051TRQAngle 867](#_Toc210985963)

[10.41.23 TxProcessLabel052TRQAEOYellow 869](#_Toc210985964)

[10.41.24 TxProcessLabel053TRQAEOYelRec 871](#_Toc210985965)

[10.41.25 TxProcessLabel054TRQAEORed 873](#_Toc210985966)

[10.41.26 TxProcessLabel055TRQAEORdDi 875](#_Toc210985967)

[10.41.27 TxProcessLabel057TRQOEIYelDash 877](#_Toc210985968)

[10.41.28 TxProcessLabel060TRQOEIRedDash 879](#_Toc210985969)

[10.41.29 TxProcessLabel061TRQOEIRedLine 880](#_Toc210985970)

[10.41.30 TxProcessLabel174HYD 882](#_Toc210985971)

[10.41.31 TxProcessLabel175HYDAng 885](#_Toc210985972)

[10.41.32 TxProcessLabel176MGBOilPress 889](#_Toc210985973)

[10.41.33 TxProcessLabel177MGBOilPrsAng 891](#_Toc210985974)

[10.41.34 TxProcessLabel200MGBOilTemp 893](#_Toc210985975)

[10.41.35 TxProcessLabel201MGBOilTempAng 895](#_Toc210985976)

[10.41.36 TxProcessLabel262FuelPress 898](#_Toc210985977)

[10.41.37 T41DisplayClass 900](#_Toc210985978)

[10.41.38 T42DisplayClass 904](#_Toc210985979)

[10.41.39 TxProcessLabel265T4 908](#_Toc210985980)

[10.41.40 TxProcessLabel266T4Angle 911](#_Toc210985981)

[10.41.41 TxProcessLabel270T4AEOYellow 913](#_Toc210985982)

[10.41.42 TxProcessLabel271T4AEOYelRec 915](#_Toc210985983)

[10.41.43 TxProcessLabel272T4AEORed 917](#_Toc210985984)

[10.41.44 TxProcessLabel273T4AEORedDiode 919](#_Toc210985985)

[10.41.45 TxProcessLabel274T4OEIYelDash 921](#_Toc210985986)

[10.41.46 TxProcessLabel275T4OEIRedDash 922](#_Toc210985987)

[10.41.47 TxProcessLabel276T4OEIRedLine 924](#_Toc210985988)

[10.41.48 TxProcessLabel300T4STARTWhiTri 926](#_Toc210985989)

[10.41.49 TxProcessLabel301T4STRTRdTriLo 927](#_Toc210985990)

[10.41.50 TxProcessLabel302T4STARTRedTriHig 929](#_Toc210985991)

[10.41.51 TxProcessLabel316EOT 931](#_Toc210985992)

[10.41.52 TxProcessLabel317EOTAng 933](#_Toc210985993)

[10.41.53 TxProcessLabel320EOP 936](#_Toc210985994)

[10.41.54 TxProcessLabel321EOPAng 938](#_Toc210985995)

[10.41.55 DeltaNG1DisplayClass 939](#_Toc210985996)

[10.41.56 DeltaNG1OEITopping 943](#_Toc210985997)

[10.41.57 DeltaNG2DisplayClass 946](#_Toc210985998)

[10.41.58 DeltaNG2OEITopping 950](#_Toc210985999)

[10.41.59 TxProcessLabel343NG 953](#_Toc210986000)

[10.41.60 TxProcessLabel344DeltaNGAngle 956](#_Toc210986001)

[10.41.61 TxProcessLabel345DltNGAEOYl 958](#_Toc210986002)

[10.41.62 TxProcessLabel346DltNGAEOYlRc 960](#_Toc210986003)

[10.41.63 TxProcessLabel347DeltaNGAEORed 962](#_Toc210986004)

[10.41.64 TxProcessLabel350DltNGAEORdDi 964](#_Toc210986005)

[10.41.65 TxProcessLabel351DltNGOEIYlDsh 966](#_Toc210986006)

[10.41.66 TxProcessLabel352DltNGOEIRdDsh 967](#_Toc210986007)

[10.41.67 TxProcessLabel353DltNGOEIRdLn 969](#_Toc210986008)

[10.41.68 TxProcessLabel012WarningStatus 971](#_Toc210986009)

[10.41.69 TxProcessLabel013WarningStatus 973](#_Toc210986010)

[10.41.70 TxProcessLabel356EngineStatus1 977](#_Toc210986011)

[10.41.71 TxProcessLabel357EngineStatus2 979](#_Toc210986012)

[10.41.72 TxProcessLabel030ModeMgmt 981](#_Toc210986013)

[10.41.73 TxProcessLabel077LightingState 983](#_Toc210986014)

[10.41.74 TxProcessLabel035DiscrepRpt 985](#_Toc210986015)

[10.41.76 T4Roundup 988](#_Toc210986016)

[10.41.77 TxProcessLabel062DAUBIT 990](#_Toc210986017)

[10.41.82 DeltaNGThreshDiscreteOutput 994](#_Toc210986018)

[10.41.83 ArincLineStatus 996](#_Toc210986019)

[10.41.84 TxProcessLabel064CKSUMMSB 998](#_Toc210986020)

[10.41.85 TxProcessLabel065CKSUMLSB 999](#_Toc210986021)

[10.41.86 TxProcessLabel031Units 1001](#_Toc210986022)

[10.41.87 HwAbsFlt 1002](#_Toc210986023)

[10.41.88 TRQRoundup 1004](#_Toc210986024)

[11 Software Low Level Requirements - Gateway Module Library 1006](#_Toc210986025)

[11.1 daulibmisc 1006](#_Toc210986026)

[11.1.1 NvicPriorityGroupConfig 1006](#_Toc210986027)

[11.1.2 NvicInit 1007](#_Toc210986028)

[11.1.3 NvicSetVectorTable 1010](#_Toc210986029)

[11.2 daulibstm32f4xxadc 1011](#_Toc210986030)

[11.2.1 AdcInit 1011](#_Toc210986031)

[11.2.2 AdcCommonInit 1014](#_Toc210986032)

[11.2.3 AdcCmd 1016](#_Toc210986033)

[11.2.4 AdcRegularChannelConfig 1017](#_Toc210986034)

[11.2.5 AdcSoftwareStartConv 1021](#_Toc210986035)

[11.2.6 AdcDmaCmd 1022](#_Toc210986036)

[11.2.7 AdcDmaReqAfterLastTransferCmd 1024](#_Toc210986037)

[11.3 daulibstm32f4xxcan 1026](#_Toc210986038)

[11.3.1 CanDeInit 1026](#_Toc210986039)

[11.3.2 CanInit 1028](#_Toc210986040)

[11.3.3 CanFilterInit 1034](#_Toc210986041)

[11.3.4 CanTransmit 1037](#_Toc210986042)

[11.3.5 CanReceive 1041](#_Toc210986043)

[11.3.6 CanItConfig 1045](#_Toc210986044)

[11.4 daulibstm32f4xxcrc 1046](#_Toc210986045)

[11.4.1 CrcResetDr 1046](#_Toc210986046)

[11.4.2 CrcCalcBlockCrc 1048](#_Toc210986047)

[11.5 daulibstm32f4xxdac 1049](#_Toc210986048)

[11.5.1 DacInit 1049](#_Toc210986049)

[11.5.2 DacStructInit 1051](#_Toc210986050)

[11.5.3 DacCmd 1053](#_Toc210986051)

[11.6 daulibstm32f4xxdma 1055](#_Toc210986052)

[11.6.1 DmaDeInit 1055](#_Toc210986053)

[11.6.2 DmaInit 1060](#_Toc210986054)

[11.6.3 DmaStructInit 1062](#_Toc210986055)

[11.6.4 DmaCmd 1063](#_Toc210986056)

[11.6.5 DmaClearItPendingBit 1065](#_Toc210986057)

[11.6.6 DmaItConfig 1067](#_Toc210986058)

[11.7 daulibstm32f4xxflash 1069](#_Toc210986059)

[11.7.1 FlashSetLatency 1069](#_Toc210986060)

[11.7.2 FlashPrefetchBufferCmd 1071](#_Toc210986061)

[11.7.3 FlashInstructionCacheCmd 1072](#_Toc210986062)

[11.7.4 FlashDataCacheCmd 1074](#_Toc210986063)

[11.8 daulibstm32f4xxfsmc 1076](#_Toc210986064)

[11.8.1 FsmcNorSramInit 1076](#_Toc210986065)

[11.8.2 FsmcNorSramCmd 1079](#_Toc210986066)

[11.9 daulibstm32f4xxgpio 1081](#_Toc210986067)

[11.9.1 GpioInit 1081](#_Toc210986068)

[11.9.2 GpioSetBits 1083](#_Toc210986069)

[11.9.3 GpioResetBits 1084](#_Toc210986070)

[11.9.4 GpioReadInputDataBit 1086](#_Toc210986071)

[11.9.5 GpioToggleBits 1087](#_Toc210986072)

[11.9.6 GpioPinAFConfig 1089](#_Toc210986073)

[11.10 daulibstm32f4xxiwdg 1091](#_Toc210986074)

[11.10.1 IwdgWriteAccessCmd 1091](#_Toc210986075)

[11.10.2 IwdgSetPrescaler 1092](#_Toc210986076)

[11.10.3 IwdgSetReload 1094](#_Toc210986077)

[11.10.4 IwdgReloadCounter 1095](#_Toc210986078)

[11.10.5 IwdgEnable 1096](#_Toc210986079)

[11.11 daulibstm32f4xxpwr 1098](#_Toc210986080)

[11.11.1 PwrMainRegulatorModeConfig 1098](#_Toc210986081)

[11.12 daulibstm32f4xxrcc 1100](#_Toc210986082)

[11.12.1 RccDeInit 1100](#_Toc210986083)

[11.12.2 RccHseConfig 1102](#_Toc210986084)

[11.12.3 RccWaitForHseStartUp 1103](#_Toc210986085)

[11.12.4 RccPllConfig 1105](#_Toc210986086)

[11.12.5 RccPllCmd 1107](#_Toc210986087)

[11.12.6 RccSysClkConfig 1108](#_Toc210986088)

[11.12.7 RccGetSysClkSource 1110](#_Toc210986089)

[11.12.8 RccHclkConfig 1111](#_Toc210986090)

[11.12.9 RccPclk1Config 1113](#_Toc210986091)

[11.12.10 RccPclk2Config 1114](#_Toc210986092)

[11.12.11 RccGetClocksFreq 1115](#_Toc210986093)

[11.12.12 RccAhb1PeriphClockCmd 1119](#_Toc210986094)

[11.12.13 RccAhb3PeriphClockCmd 1120](#_Toc210986095)

[11.12.14 RccApb1PeriphClockCmd 1122](#_Toc210986096)

[11.12.15 RccGetFlagStatus 1124](#_Toc210986097)

[11.12.16 RccApb1PeriphResetCmd 1125](#_Toc210986098)

[11.12.17 RccApb2PeriphResetCmd 1127](#_Toc210986099)

[11.12.18 RccApb2PeriphClockCmd 1128](#_Toc210986100)

[11.13 daulibstm32f4xxtim 1130](#_Toc210986101)

[11.13.1 TimPrescalerConfig 1130](#_Toc210986102)

[11.13.2 TimSetAutoReload 1132](#_Toc210986103)

[11.13.3 TimCmd 1133](#_Toc210986104)

[11.13.4 TimItConfig 1135](#_Toc210986105)

[11.13.5 TimSetIC1Prescaler 1136](#_Toc210986106)

[11.13.6 TimSetIC2Prescaler 1138](#_Toc210986107)

[11.13.7 TimSetIC3Prescaler 1139](#_Toc210986108)

[11.13.8 TimSetIC4Prescaler 1141](#_Toc210986109)

[11.13.9 TimDmaCmd 1143](#_Toc210986110)

[11.13.10 Ti1Config 1144](#_Toc210986111)

[11.13.11 Ti2Config 1147](#_Toc210986112)

[11.13.12 Ti3Config 1149](#_Toc210986113)

[11.13.13 Ti4Config 1151](#_Toc210986114)

[11.13.14 TimICInit 1154](#_Toc210986115)

[11.13.15 TimClearITPendingBit 1156](#_Toc210986116)

[11.13.16 TimSelectOutputTrigger 1157](#_Toc210986117)

[11.14 daulibstm32f4xxusart 1159](#_Toc210986118)

[11.14.1 UsartDeInit 1159](#_Toc210986119)

[11.14.2 UsartInit 1162](#_Toc210986120)

[11.14.3 UsartCmd 1165](#_Toc210986121)

[11.14.4 UsartSendData 1167](#_Toc210986122)

[11.14.5 UsartReceiveData 1168](#_Toc210986123)

[11.14.6 UsartItConfig 1170](#_Toc210986124)

[11.14.7 UsartGetItStatus 1172](#_Toc210986125)

[12 Software Low Level Requirements - MCD Software 1175](#_Toc210986126)

[12.1 daugwyappdata.c 1175](#_Toc210986127)

[12.1.1 Brief Description 1175](#_Toc210986128)

[12.1.2 List of HLRs allocated 1175](#_Toc210986129)

[12.1.3 Description of list of LLRs allocated 1176](#_Toc210986130)

[13 Appendix A : Data Dictionary 1180](#_Toc210986131)

[14 Appendix B: Data Constants 1180](#_Toc210986132)

List of Tables

[Table 1 : Amendment Record 13](#_Toc210986133)

[Table 2 : Reference document 51](#_Toc210986134)

[Table 3 : List of Abbreviation 52](#_Toc210986135)

[Table 4 : RS232 Configuration parameters 1176](#_Toc210986136)

[Table 5 : Analog Sensor lookup table configuration 1176](#_Toc210986137)

[Table 6 : ARINC 429 channel configuration parameters 1177](#_Toc210986138)

[Table 7 : Audio tones output Configuration format 1178](#_Toc210986139)

[Table 8 : Analog Parameter configuration format for TACH Channels 1179](#_Toc210986140)

# 1 Amendment Record

Table : Amendment Record

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Version Number** | **Description of Amendment** | **Change**  **Request No.** | **Changed By** | **Release**  **Date** |
| 1.1 | Initial Release | NA | Chaithra J | 01-Feb-2022 |
| 1.2 | 1. Below Sections are updated based on LLD Review comments:  10.3.1.4,10.16.6.4 , 10.16.6.6 ,10.16.7.3,10.17.2.6,10.17.4,10.18.3.6,  10.18.5.6,10.19.1.6,10.19.2.6,10.19.1.3, 10.23.33.5,10.23.32.6,10.23.18.6,  10.23.16.1 ,10.22.89.1,10.22.54.1, 10.22.53.1, 10.22.52.1, 10.33.4.3  2. Below Requirement ID’s are updated based on LLD Review comments:  H398-LLD-GWY-FNC-1518,H398-LLD-GWY-FNC-1519,H398-LLD-GWY-FNC-1520,  H398-LLD-GWY-FNC-1521,H398-LLD-GWY-FNC-1522,H398-LLD-GWY-FNC-1523,  H398-LLD-GWY-FNC-1524,H398-LLD-GWY-FNC-1525,H398-LLD-GWY-FNC-1526,  H398-LLD-GWY-FNC-1527,H398-LLD-GWY-FNC-1556,H398-LLD-GWY-FNC-1557,  H398-LLD-GWY-FNC-1589,H398-LLD-GWY-FNC-1590,H398-LLD-GWY-FNC-1583,  H398-LLD-GWY-FNC-1615,H398-LLD-GWY-FNC-651,H398-LLD-GWY-FNC-1621,  H398-LLD-GWY-FNC-1635,H398-LLD-GWY-FNC-1636,H398-LLD-GWY-FNC-1637,  H398-LLD-GWY-FNC-1641,H398-LLD-GWY-FNC-1760,H398-LLD-GWY-FNC-1770,  H398-LLD-GWY-FNC-1771,H398-LLD-GWY-FNC-1772,H398-LLD-GWY-FNC-1775,  H398-LLD-GWY-FNC-3687,H398-LLD-GWY-FNC-4146,H398-LLD-GWY-FNC-3932,  H398-LLD-GWY-FNC-3945,H398-LLD-GWY-FNC-4440,H398-LLD-GWY-FNC-4450,  H398-LLD-GWY-FNC-2650,H398-LLD-GWY-FNC-2670,H398-LLD-GWY-FNC-3328,  H398-LLD-GWY-FNC-3329,H398-LLD-GWY-FNC-3342,H398-LLD-GWY-FNC-3507,  H398-LLD-GWY-FNC-3508,H398-LLD-GWY-FNC-4599,H398-LLD-GWY-FNC-4616,  H398-LLD-GWY-FNC-4768,H398-LLD-GWY-FNC-4842  3.Below Requirement ID’s are deleted based on LLD Review comments:  H398-LLD-GWY-FNC-4353  H398-LLD-GWY-FNC-4354  H398-LLD-GWY-FNC-4355,  4.All instances of DoMirrorcheck is replaced with DoMirrorCheck  5.Below Requirement ID’s are updated as part of self review comments:  H398-LLD-GWY-FNC-862,H398-LLD-GWY-FNC-5762,H398-LLD-GWY-FNC-5745,  H398-LLD-GWY-FNC-5778,H398-LLD-GWY-FNC-5404,H398-LLD-GWY-FNC-5405,  H398-LLD-GWY-FNC-5474,H398-LLD-GWY-FNC-321,H398-LLD-GWY-FNC-5746,  H398-LLD-GWY-FNC-4389,H398-LLD-GWY-FNC-4202,H398-LLD-GWY-FNC-2095,  H398-LLD-GWY-FNC-2096,H398-LLD-GWY-FNC-2837,H398-LLD-GWY-FNC-3096,  H398-LLD-GWY-FNC-3339,H398-LLD-GWY-FNC-3340,H398-LLD-GWY-FNC-3341,  H398-LLD-GWY-FNC-3342,H398-LLD-GWY-FNC-3351,H398-LLD-GWY-FNC-5777,  H398-LLD-GWY-FNC-3352,H398-LLD-GWY-FNC-3353,H398-LLD-GWY-FNC-3362,  H398-LLD-GWY-FNC-3363,H398-LLD-GWY-FNC-3364,H398-LLD-GWY-FNC-3373,  H398-LLD-GWY-FNC-3374,H398-LLD-GWY-FNC-3375,H398-LLD-GWY-FNC-3376,  H398-LLD-GWY-FNC-3385,H398-LLD-GWY-FNC-3386,H398-LLD-GWY-FNC-3387,  H398-LLD-GWY-FNC-3388,H398-LLD-GWY-FNC-3391,H398-LLD-GWY-FNC-3392,  H398-LLD-GWY-FNC-3393,H398-LLD-GWY-FNC-3394,H398-LLD-GWY-FNC-3395,  H398-LLD-GWY-FNC-3396,H398-LLD-GWY-FNC-3405,H398-LLD-GWY-FNC-3406,  H398-LLD-GWY-FNC-3614,H398-LLD-GWY-DRQ-3657,H398-LLD-GWY-FNC-3239,  H398-LLD-GWY-FNC-3244,H398-LLD-GWY-FNC-3248,H398-LLD-GWY-FNC-3258,  H398-LLD-GWY-FNC-4377,H398-LLD-GWY-FNC-4379,H398-LLD-GWY-FNC-4790,  H398-LLD-GWY-FNC-4781,H398-LLD-GWY-FNC-4907,H398-LLD-GWY-FNC-4908,  H398-LLD-GWY-FNC-5091,H398-LLD-GWY-FNC-5092,H398-LLD-GWY-FNC-5190,  H398-LLD-GWY-FNC-5779,H398-LLD-GWY-FNC-5463,  H398-LLD-GWY-FNC-5326,H398-LLD-GWY-FNC-4067,H398-LLD-GWY-FNC-4035, H398-LLD-GWY-DRQ-1666, H398-LLD-GWY-DRQ-1675, H398-LLD-GWY-DRQ-1684.  6. Newly added Requirement ID's as per code review comments:  H398-LLD-GWY-FNC-5784, H398-LLD-GWY-FNC-5781,  LLD-GWY-FNC-5782, LLD-GWY-FNC-5783  7.Updated as part of code review comments:  H398-LLD-GWY-FNC-2095,H398-LLD-GWY-FNC-2096,  H398-LLD-GWY-FNC-4128,H398-LLD-GWY-FNC-4129,  H398-LLD-GWY-FNC-4130,H398-LLD-GWY-FNC-4202,  H398-LLD-GWY-FNC-4339,H398-LLD-GWY-FNC-4340,  H398-LLD-GWY-FNC-4341,H398-LLD-GWY-FNC-4342,  H398-LLD-GWY-FNC-4343,H398-LLD-GWY-FNC-4344,  H398-LLD-GWY-FNC-1773,H398-LLD-GWY-FNC-321,  H398-LLD-GWY-FNC-322,H398-LLD-GWY-FNC-349,  H398-LLD-GWY-FNC-520,H398-LLD-GWY-FNC-376,  H398-LLD-GWY-FNC-522,H398-LLD-GWY-FNC-573,  H398-LLD-GWY-FNC-574,H398-LLD-GWY-FNC-3530,  H398-LLD-GWY-FNC-3539,H398-LLD-GWY-FNC-3548,  H398-LLD-GWY-FNC-3557,H398-LLD-GWY-FNC-3558,  H398-LLD-GWY-FNC-3579,H398-LLD-GWY-FNC-3580,  H398-LLD-GWY-FNC-3583,H398-LLD-GWY-FNC-3584,  H398-LLD-GWY-FNC-3585,H398-LLD-GWY-FNC-3586,  H398-LLD-GWY-FNC-3588,H398-LLD-GWY-FNC-3625,  H398-LLD-GWY-FNC-3626,H398-LLD-GWY-FNC-3627,  H398-LLD-GWY-FNC-3628,H398-LLD-GWY-FNC-3629,  H398-LLD-GWY-FNC-3630,H398-LLD-GWY-FNC-3631,  H398-LLD-GWY-FNC-3632,H398-LLD-GWY-FNC-5761, H398-LLD-GWY-FNC-575, H398-LLD-GWY-FNC-3577  8. Function name changed from CVRRecordingEnableOutput to CvrRecordingEnableOutput.based on Code review comments.  9.DD and Dc updated as part of code review comments. | 100034/100035 | Chaithra J | 11-Feb-2022 |
| 1.3 | DD and DC updated as per QA Review Comments.  Below Requirement ID’s are updated as part of QA review comments:  H398-LLD-GWY-FNC-1524  H398-LLD-GWY-FNC-1537  H398-LLD-GWY-FNC-1538  H398-LLD-GWY-FNC-1948  H398-LLD-GWY-FNC-21  H398-LLD-GWY-FNC-1788  H398-LLD-GWY-FNC-1763  H398-LLD-GWY-FNC-1772  H398-LLD-GWY-FNC-1777  H398-LLD-GWY-FNC-1929  H398-LLD-GWY-FNC-328  H398-LLD-GWY-FNC-329  H398-LLD-GWY-FNC-347  H398-LLD-GWY-FNC-5752  H398-LLD-GWY-FNC-3798  Below sections are updated based on QA review comments:  10.16.2.4, 10.16.5.1, 10.23.36.4 | 100034/100035 | Chaithra J | 11-Feb-2022 |
| 1.4 | Updated sections/requirements as mentioned in PR100098 | PR100098 | P Afreen | 28-Dec-2022 |
| 1.5 | Updated as per review comments and self-review  H398-LLD-GWY-FNC-4459  H398-LLD-GWY-FNC-4461  H398-LLD-GWY-FNC-4464  H398-LLD-GWY-FNC-4465  H398-LLD-GWY-FNC-4489  H398-LLD-GWY-FNC-4491  H398-LLD-GWY-FNC-4494  H398-LLD-GWY-FNC-4495  H398-LLD-GWY-FNC-4516  H398-LLD-GWY-FNC-4517  H398-LLD-GWY-FNC-4518  H398-LLD-GWY-FNC-4523  H398-LLD-GWY-FNC-4525  H398-LLD-GWY-FNC-4527  H398-LLD-GWY-FNC-4536  H398-LLD-GWY-FNC-4537  H398-LLD-GWY-FNC-4538  H398-LLD-GWY-FNC-4543  H398-LLD-GWY-FNC-4545  H398-LLD-GWY-FNC-4547  H398-LLD-GWY-FNC-4555  H398-LLD-GWY-FNC-4556  H398-LLD-GWY-FNC-4557  H398-LLD-GWY-FNC-4562  H398-LLD-GWY-FNC-4564  H398-LLD-GWY-FNC-4566  H398-LLD-GWY-FNC-4575  H398-LLD-GWY-FNC-4576  H398-LLD-GWY-FNC-4577  H398-LLD-GWY-FNC-4582  H398-LLD-GWY-FNC-4584  H398-LLD-GWY-FNC-4586  H398-LLD-GWY-FNC-4595  H398-LLD-GWY-FNC-4596  H398-LLD-GWY-FNC-4597  H398-LLD-GWY-FNC-4602  H398-LLD-GWY-FNC-4604  H398-LLD-GWY-FNC-4606  H398-LLD-GWY-FNC-4277  H398-LLD-GWY-FNC-4344  H398-LLD-GWY-FNC-4163  H398-LLD-GWY-FNC-3919  H398-LLD-GWY-FNC-3932  H398-LLD-GWY-FNC-3945  H398-LLD-GWY-FNC-3958  H398-LLD-GWY-FNC-3971  H398-LLD-GWY-FNC-6083  H398-LLD-GWY-FNC-6250  H398-LLD-GWY-FNC-6252  H398-LLD-GWY-FNC-2146  H398-LLD-GWY-FNC-2365  H398-LLD-GWY-FNC-6278  The below requirements are updated as per self review  H398-LLD-GWY-FNC-5922  H398-LLD-GWY-FNC-5923  H398-LLD-GWY-FNC-6209  H398-LLD-GWY-FNC-6283  H398-LLD-GWY-FNC-2817  H398-LLD-GWY-FNC-2930  H398-LLD-GWY-FNC-6401  H398-LLD-GWY-FNC-2994  H398-LLD-GWY-FNC-6340  H398-LLD-GWY-FNC-6082  H398-LLD-GWY-FNC-6190  H398-LLD-GWY-FNC-2493  H398-LLD-GWY-FNC-6210  H398-LLD-GWY-FNC-6079  H398-LLD-GWY-FNC-6190  H398-LLD-GWY-FNC-6247  H398-LLD-GWY-FNC-6080  H398-LLD-GWY-FNC-6083  H398-LLD-GWY-FNC-6150  H398-LLD-GWY-FNC-169  H398-LLD-GWY-FNC-91  H398-LLD-GWY-FNC-171  H398-LLD-GWY-FNC-180  H398-LLD-GWY-FNC-179  H398-LLD-GWY-FNC-178  H398-LLD-GWY-FNC-177  H398-LLD-GWY-FNC-6189  H398-LLD-GWY-FNC-6188  H398-LLD-GWY-FNC-176  H398-LLD-GWY-FNC-174  H398-LLD-GWY-FNC-175  H398-LLD-GWY-FNC-173  H398-LLD-GWY-FNC-172  H398-LLD-GWY-FNC-170  Modified section's  10.4.25 AppExciteConditions  10.4.25.1 Brief Description  Data constant and Data Dictionary sheet  Newly added ID's  H398-LLD-GWY-FNC-6424  H398-LLD-GWY-FNC-6425  H398-LLD-GWY-FNC-6426  Deleted ID's and sections  H398-LLD-GWY-FNC-505  H398-LLD-GWY-FNC-506  H398-LLD-GWY-FNC-507  H398-LLD-GWY-FNC-508  10.4.26 AppExciteConditions  10.4.26.1 Brief Description  10.4.26.2 List of HLRs allocated  10.4.26.3 List of global variables accessed and modified  10.4.26.4 Parameter list (Input/Output)  10.4.26.5 Return Value  10.4.26.6 Other CSUs called by this CSU  10.4.26.7 Description of list of LLRs allocated | PR100098 | P Afreen | 3-Jan-2023 |
| 1.6 | The below mention requirements and sections are updated as per QA comments:  H398-LLD-GWY-FNC-2855  H398-LLD-GWY-FNC-5909  H398-LLD-GWY-FNC-5689  H398-LLD-GWY-FNC-5949  H398-LLD-GWY-FNC-5924  H398-LLD-GWY-FNC-5927  H398-LLD-GWY-FNC-5928  H398-LLD-GWY-FNC-5942  H398-LLD-GWY-FNC-5943  H398-LLD-GWY-DRQ-5868  H398-LLD-GWY-FNC-3585  H398-LLD-GWY-FNC-3637  H398-LLD-GWY-FNC-3638  H398-LLD-GWY-FNC-37  H398-LLD-GWY-FNC-38  H398-LLD-GWY-FNC-39  H398-LLD-GWY-FNC-40  H398-LLD-GWY-FNC-41  H398-LLD-GWY-FNC-3390  Appendix A : Data Dictionary  Appendix B: Data Constants  Deleted requirements  H398-LLD-GWY-FNC-6148  H398-LLD-GWY-FNC-5908 | PR100098 | P Afreen | 7-Jan-2023 |
| 1.7 | The below mentioned requirements are updated as per LLRT observations.  H398-LLD-GWY-FNC-6182  H398-LLD-GWY-FNC-694  H398-LLD-GWY-FNC-695  The below mentioned requirements and section are updated as per changes for HSIT observations.  H398-LLD-GWY-FNC-6164  H398-LLD-GWY-FNC-5817  H398-LLD-GWY-FNC-6173  H398-LLD-GWY-FNC-5752  Appendix B : Data Constant | PR100143 | P Afreen | 19-Jan-2023 |
| 2.0 | The below sections got modified  according to the self review:  Section 2: Objective  Section 4:References  Section 9: Traceability  Section 6:Document control  Appendix A : Data Dictionary  Appendix B : Data Constant  The below sections got added:  10.4.41.ReconfigurationFromDU  10.21.22.CrossChannelEssential  10.21.23.CrossChannelCritical  10.21.24.CrossChannelAirbusAIF  10.4.daugwylogicairbus.c  The below requirements got  modified according to the  self review:  H398-LLD-GWY-FNC-442  H398-LLD-GWY-FNC-6079  H398-LLD-GWY-FNC-518  H398-LLD-GWY-FNC-6112  H398-LLD-GWY-FNC-570  H398-LLD-GWY-FNC-1760  H398-LLD-GWY-FNC-1776  H398-LLD-GWY-FNC-2054  H398-LLD-GWY-FNC-2055  H398-LLD-GWY-FNC-2056  H398-LLD-GWY-FNC-2057  H398-LLD-GWY-FNC-2058  H398-LLD-GWY-FNC-2059  H398-LLD-GWY-FNC-2060  H398-LLD-GWY-FNC-2061  H398-LLD-GWY-FNC-2062  H398-LLD-GWY-FNC-6092  H398-LLD-GWY-FNC-2114  H398-LLD-GWY-FNC-5842  H398-LLD-GWY-FNC-2129  H398-LLD-GWY-FNC-2130  H398-LLD-GWY-FNC-5990  H398-LLD-GWY-FNC-2145  H398-LLD-GWY-FNC-2146  H398-LLD-GWY-FNC-2755  H398-LLD-GWY-FNC-2828  H398-LLD-GWY-FNC-3039  The below requirements got added:  H398-LLD-GWY-FNC-6444  H398-LLD-GWY-FNC-6445  H398-LLD-GWY-FNC-6446  H398-LLD-GWY-FNC-6447  H398-LLD-GWY-FNC-6448  H398-LLD-GWY-FNC-6449  H398-LLD-GWY-FNC-6459  H398-LLD-GWY-FNC-6460  H398-LLD-GWY-FNC-6461  H398-LLD-GWY-FNC-6462  H398-LLD-GWY-FNC-6463  H398-LLD-GWY-FNC-6464  H398-LLD-GWY-FNC-6465  H398-LLD-GWY-FNC-6466  H398-LLD-GWY-FNC-6467  H398-LLD-GWY-FNC-6468  H398-LLD-GWY-FNC-6469  H398-LLD-GWY-FNC-6470  H398-LLD-GWY-FNC-6471  H398-LLD-GWY-FNC-6472  H398-LLD-GWY-FNC-6473  H398-LLD-GWY-FNC-6474  H398-LLD-GWY-FNC-6475  H398-LLD-GWY-FNC-6476  H398-LLD-GWY-FNC-6485  H398-LLD-GWY-FNC-6495  H398-LLD-GWY-FNC-6504  H398-LLD-GWY-FNC-6513  H398-LLD-GWY-FNC-6514  H398-LLD-GWY-FNC-6515  H398-LLD-GWY-FNC-6516  H398-LLD-GWY-FNC-6517  H398-LLD-GWY-FNC-6518  H398-LLD-GWY-FNC-6519  H398-LLD-GWY-FNC-6520  H398-LLD-GWY-FNC-6529  H398-LLD-GWY-FNC-6530  H398-LLD-GWY-FNC-6531  H398-LLD-GWY-FNC-6532  H398-LLD-GWY-FNC-6541  H398-LLD-GWY-FNC-6542  H398-LLD-GWY-FNC-6543  H398-LLD-GWY-FNC-6544  H398-LLD-GWY-FNC-6545  H398-LLD-GWY-FNC-6546  H398-LLD-GWY-FNC-6555  H398-LLD-GWY-FNC-6556  H398-LLD-GWY-FNC-6565  H398-LLD-GWY-FNC-6566  H398-LLD-GWY-FNC-6567  H398-LLD-GWY-FNC-6576  H398-LLD-GWY-FNC-6585  H398-LLD-GWY-FNC-6594  H398-LLD-GWY-FNC-6603  H398-LLD-GWY-FNC-6604  H398-LLD-GWY-FNC-6605  H398-LLD-GWY-FNC-6606  H398-LLD-GWY-FNC-6607  H398-LLD-GWY-FNC-6608  H398-LLD-GWY-FNC-6609  H398-LLD-GWY-FNC-6610  H398-LLD-GWY-FNC-6619  H398-LLD-GWY-FNC-6620  H398-LLD-GWY-FNC-6629  H398-LLD-GWY-FNC-6639  H398-LLD-GWY-FNC-6640  H398-LLD-GWY-FNC-6641  H398-LLD-GWY-FNC-6651  H398-LLD-GWY-FNC-6660  H398-LLD-GWY-FNC-6661  H398-LLD-GWY-FNC-6676  H398-LLD-GWY-FNC-6677  H398-LLD-GWY-FNC-6678  H398-LLD-GWY-FNC-6679  H398-LLD-GWY-FNC-6680  H398-LLD-GWY-FNC-6681  H398-LLD-GWY-FNC-6682  H398-LLD-GWY-FNC-6683  H398-LLD-GWY-FNC-6684  H398-LLD-GWY-FNC-7406  H398-LLD-GWY-FNC-6693  H398-LLD-GWY-FNC-6694  H398-LLD-GWY-FNC-6695  H398-LLD-GWY-FNC-6696  H398-LLD-GWY-FNC-6697  H398-LLD-GWY-FNC-6698  H398-LLD-GWY-FNC-6699  H398-LLD-GWY-FNC-6708  H398-LLD-GWY-FNC-6717  H398-LLD-GWY-FNC-6726  H398-LLD-GWY-FNC-6735  H398-LLD-GWY-FNC-6744  H398-LLD-GWY-FNC-6753  H398-LLD-GWY-FNC-6762  H398-LLD-GWY-FNC-6771  H398-LLD-GWY-FNC-6781  H398-LLD-GWY-FNC-6782  H398-LLD-GWY-FNC-6791  H398-LLD-GWY-FNC-6792  H398-LLD-GWY-FNC-6793  H398-LLD-GWY-FNC-6794  H398-LLD-GWY-FNC-6795  H398-LLD-GWY-FNC-6796  H398-LLD-GWY-FNC-6797  H398-LLD-GWY-FNC-6798  H398-LLD-GWY-FNC-6799  H398-LLD-GWY-FNC-6800  H398-LLD-GWY-FNC-6809  H398-LLD-GWY-FNC-6818  H398-LLD-GWY-FNC-6819  H398-LLD-GWY-FNC-6828  H398-LLD-GWY-FNC-6829  H398-LLD-GWY-FNC-6838  H398-LLD-GWY-FNC-6839  H398-LLD-GWY-FNC-6840  H398-LLD-GWY-FNC-6849  H398-LLD-GWY-FNC-6850  H398-LLD-GWY-FNC-6860  H398-LLD-GWY-FNC-6862  H398-LLD-GWY-FNC-6863  H398-LLD-GWY-FNC-6864  H398-LLD-GWY-FNC-6865  H398-LLD-GWY-FNC-6866  H398-LLD-GWY-FNC-6876  H398-LLD-GWY-FNC-6877  H398-LLD-GWY-FNC-6878  H398-LLD-GWY-FNC-6879  H398-LLD-GWY-FNC-6880  H398-LLD-GWY-FNC-6881  H398-LLD-GWY-FNC-6890  H398-LLD-GWY-FNC-6891  H398-LLD-GWY-FNC-6892  H398-LLD-GWY-FNC-6893  H398-LLD-GWY-FNC-6904  H398-LLD-GWY-FNC-6913  H398-LLD-GWY-FNC-6914  H398-LLD-GWY-FNC-6915  H398-LLD-GWY-FNC-6916  H398-LLD-GWY-FNC-6917  H398-LLD-GWY-FNC-6926  H398-LLD-GWY-FNC-6935  H398-LLD-GWY-FNC-6944  H398-LLD-GWY-FNC-6953  H398-LLD-GWY-FNC-6962  H398-LLD-GWY-FNC-6971  H398-LLD-GWY-FNC-6981  H398-LLD-GWY-FNC-6990  H398-LLD-GWY-FNC-6999  H398-LLD-GWY-FNC-7008  H398-LLD-GWY-FNC-7009  H398-LLD-GWY-FNC-7010  H398-LLD-GWY-FNC-7012  H398-LLD-GWY-FNC-7013  H398-LLD-GWY-FNC-7014  H398-LLD-GWY-FNC-7023  H398-LLD-GWY-FNC-7024  H398-LLD-GWY-FNC-7033  H398-LLD-GWY-FNC-7043  H398-LLD-GWY-FNC-7052  H398-LLD-GWY-FNC-7054  H398-LLD-GWY-FNC-7055  H398-LLD-GWY-FNC-7056  H398-LLD-GWY-FNC-7057  H398-LLD-GWY-FNC-7058  H398-LLD-GWY-FNC-7067  H398-LLD-GWY-FNC-7068  H398-LLD-GWY-FNC-7069  H398-LLD-GWY-FNC-7078  H398-LLD-GWY-FNC-7080  H398-LLD-GWY-FNC-7081  H398-LLD-GWY-FNC-7082  H398-LLD-GWY-FNC-7083  H398-LLD-GWY-FNC-7084  H398-LLD-GWY-FNC-7093  H398-LLD-GWY-FNC-7094  H398-LLD-GWY-FNC-7095  H398-LLD-GWY-FNC-7105  H398-LLD-GWY-FNC-7106  H398-LLD-GWY-FNC-7107  H398-LLD-GWY-FNC-7108  H398-LLD-GWY-FNC-7120  H398-LLD-GWY-FNC-7121  H398-LLD-GWY-FNC-7130  H398-LLD-GWY-FNC-7131  H398-LLD-GWY-FNC-7132  H398-LLD-GWY-FNC-7142  H398-LLD-GWY-FNC-7151  H398-LLD-GWY-FNC-7161  H398-LLD-GWY-FNC-7171  H398-LLD-GWY-FNC-7180  H398-LLD-GWY-FNC-7189  H398-LLD-GWY-FNC-7198  H398-LLD-GWY-FNC-7207  H398-LLD-GWY-FNC-7208  H398-LLD-GWY-FNC-7209  H398-LLD-GWY-FNC-7210  H398-LLD-GWY-FNC-7339  H398-LLD-GWY-FNC-7219  H398-LLD-GWY-FNC-7228  H398-LLD-GWY-FNC-7237  H398-LLD-GWY-FNC-7238  H398-LLD-GWY-FNC-7239  H398-LLD-GWY-FNC-7248  H398-LLD-GWY-FNC-7249  H398-LLD-GWY-FNC-7250  H398-LLD-GWY-FNC-7259  H398-LLD-GWY-FNC-7268  H398-LLD-GWY-FNC-7277  H398-LLD-GWY-FNC-7278  H398-LLD-GWY-FNC-7349  H398-LLD-GWY-FNC-7350  H398-LLD-GWY-FNC-7351  H398-LLD-GWY-FNC-7352  H398-LLD-GWY-FNC-7353  H398-LLD-GWY-FNC-7354  H398-LLD-GWY-FNC-7355  H398-LLD-GWY-FNC-7356  H398-LLD-GWY-FNC-7357  H398-LLD-GWY-FNC-7366  H398-LLD-GWY-FNC-7375  H398-LLD-GWY-FNC-7384  H398-LLD-GWY-FNC-7393  H398-LLD-GWY-FNC-7402  H398-LLD-GWY-FNC-7403  H398-LLD-GWY-FNC-7404  H398-LLD-GWY-FNC-7405  H398-LLD-GWY-FNC-7321  H398-LLD-GWY-FNC-7325  H398-LLD-GWY-FNC-7326  H398-LLD-GWY-FNC-7327  H398-LLD-GWY-FNC-7328  H398-LLD-GWY-FNC-7329  H398-LLD-GWY-FNC-7330  H398-LLD-GWY-FNC-7331  H398-LLD-GWY-FNC-7322  H398-LLD-GWY-FNC-7323  H398-LLD-GWY-FNC-7324  H398-LLD-GWY-FNC-7287  H398-LLD-GWY-FNC-7288  H398-LLD-GWY-FNC-7289  H398-LLD-GWY-FNC-7290  H398-LLD-GWY-FNC-7291  H398-LLD-GWY-FNC-7292  H398-LLD-GWY-FNC-7293  H398-LLD-GWY-FNC-7294  H398-LLD-GWY-FNC-7295  H398-LLD-GWY-FNC-7296  H398-LLD-GWY-FNC-7297  H398-LLD-GWY-FNC-7298  H398-LLD-GWY-FNC-7299  H398-LLD-GWY-FNC-7300  H398-LLD-GWY-FNC-7301  H398-LLD-GWY-FNC-7302  H398-LLD-GWY-FNC-7303  H398-LLD-GWY-FNC-7304  H398-LLD-GWY-FNC-7407  H398-LLD-GWY-FNC-7408  H398-LLD-GWY-FNC-7409  H398-LLD-GWY-FNC-7410  H398-LLD-GWY-FNC-7411  H398-LLD-GWY-FNC-7412  H398-LLD-GWY-FNC-7332  H398-LLD-GWY-FNC-7333  H398-LLD-GWY-FNC-7334  H398-LLD-GWY-FNC-7335  H398-LLD-GWY-FNC-7336  H398-LLD-GWY-FNC-7413  H398-LLD-GWY-FNC-7414  H398-LLD-GWY-FNC-7415  H398-LLD-GWY-FNC-7416  H398-LLD-GWY-FNC-7313  H398-LLD-GWY-FNC-7314  H398-LLD-GWY-FNC-7315  H398-LLD-GWY-FNC-7316  H398-LLD-GWY-FNC-7317  H398-LLD-GWY-FNC-7318  H398-LLD-GWY-FNC-7319  H398-LLD-GWY-FNC-7320  H398-LLD-GWY-FNC-7425  H398-LLD-GWY-FNC-7426  H398-LLD-GWY-FNC-7427  H398-LLD-GWY-FNC-7428  H398-LLD-GWY-FNC-7429  H398-LLD-GWY-FNC-7430  H398-LLD-GWY-FNC-7431  H398-LLD-GWY-FNC-7432  H398-LLD-GWY-FNC-7433  H398-LLD-GWY-FNC-7442  H398-LLD-GWY-FNC-7443  H398-LLD-GWY-FNC-7444  H398-LLD-GWY-FNC-7445 | PR100171 | Nabirasul A H | 31-Jan-2024 |
| 2.1 | Update as per self review  Modified Desiginations in Front page | PR100171 | SRAVANTHI MUPPINENI | 12-02-2024 |
| 2.2 | The below requirements and sections got updated as per “QA”review comments  H398-LLD-GWY-FNC-3144  H398-LLD-GWY-FNC-3145  H398-LLD-GWY-FNC-6449  H398-LLD-GWY-FNC-6444  H398-LLD-GWY-FNC-6445  H398-LLD-GWY-FNC-6446  H398-LLD-GWY-FNC-6447  H398-LLD-GWY-FNC-6448 Section 10.4.36.6 Other CSUs called by this CSUSection10.41.1.3 List of global variables accessed and modified | PR100171 | SRAVANTHI MUPPINENI | 14-02-24 |
| 2.3 | The below requirements are modified as per self review H398-LLD-GWY-FNC-145  H398-LLD-GWY-FNC-5785  H398-LLD-GWY-FNC-6233  H398-LLD-GWY-FNC-6234  H398-LLD-GWY-FNC-6070  H398-LLD-GWY-FNC-6112  H398-LLD-GWY-FNC-1892  H398-LLD-GWY-FNC-1893  H398-LLD-GWY-FNC-1894  H398-LLD-GWY-FNC-1895  H398-LLD-GWY-FNC-1896  H398-LLD-GWY-FNC-1897  H398-LLD-GWY-FNC-1898  H398-LLD-GWY-FNC-1899  H398-LLD-GWY-FNC-1900  H398-LLD-GWY-FNC-1901  H398-LLD-GWY-FNC-5814  H398-LLD-GWY-FNC-7426  H398-LLD-GWY-FNC-6485  H398-LLD-GWY-FNC-6541  H398-LLD-GWY-FNC-6542  H398-LLD-GWY-FNC-6545  H398-LLD-GWY-FNC-6546  H398-LLD-GWY-FNC-6619  H398-LLD-GWY-FNC-6620  H398-LLD-GWY-FNC-6639  H398-LLD-GWY-FNC-6695  H398-LLD-GWY-FNC-6860  H398-LLD-GWY-FNC-6862  H398-LLD-GWY-FNC-6876  H398-LLD-GWY-FNC-6877  H398-LLD-GWY-FNC-6890  H398-LLD-GWY-FNC-6892  H398-LLD-GWY-FNC-6893  H398-LLD-GWY-FNC-6904  H398-LLD-GWY-FNC-6915  H398-LLD-GWY-FNC-7052  H398-LLD-GWY-FNC-7067  H398-LLD-GWY-FNC-7068  H398-LLD-GWY-FNC-7078  H398-LLD-GWY-FNC-7093  H398-LLD-GWY-FNC-7094  H398-LLD-GWY-FNC-7106  H398-LLD-GWY-FNC-7107  H398-LLD-GWY-FNC-7121  H398-LLD-GWY-FNC-7237  H398-LLD-GWY-FNC-7239  H398-LLD-GWY-FNC-7402  H398-LLD-GWY-FNC-7404  H398-LLD-GWY-FNC-7410  H398-LLD-GWY-FNC-7412  H398-LLD-GWY-FNC-852  H398-LLD-GWY-FNC-2214  H398-LLD-GWY-FNC-6444  H398-LLD-GWY-FNC-6445  H398-LLD-GWY-FNC-6446  H398-LLD-GWY-FNC-6447  H398-LLD-GWY-FNC-6448  H398-LLD-GWY-FNC-6680  H398-LLD-GWY-FNC-6679  H398-LLD-GWY-FNC-6682  H398-LLD-GWY-FNC-6683  H398-LLD-GWY-FNC-6684  H398-LLD-GWY-FNC-6698  H398-LLD-GWY-FNC-6744  H398-LLD-GWY-FNC-7056  H398-LLD-GWY-FNC-7058  H398-LLD-GWY-FNC-7082  H398-LLD-GWY-FNC-7084  H398-LLD-GWY-FNC-7105  H398-LLD-GWY-FNC-7238  H398-LLD-GWY-FNC-7249  H398-LLD-GWY-FNC-7250  H398-LLD-GWY-FNC-7352  H398-LLD-GWY-FNC-7408  H398-LLD-GWY-FNC-39  H398-LLD-GWY-FNC-50  H398-LLD-GWY-FNC-51  H398-LLD-GWY-FNC-52  H398-LLD-GWY-FNC-53  H398-LLD-GWY-FNC-54  H398-LLD-GWY-FNC-55  H398-LLD-GWY-FNC-56  H398-LLD-GWY-FNC-57  H398-LLD-GWY-FNC-58  H398-LLD-GWY-FNC-59  H398-LLD-GWY-FNC-177  H398-LLD-GWY-FNC-6079  H398-LLD-GWY-FNC-6081  H398-LLD-GWY-FNC-2147  H398-LLD-GWY-FNC-6191  H398-LLD-GWY-FNC-2162  H398-LLD-GWY-FNC-2251  H398-LLD-GWY-FNC-7351  The below requirements got added  H398-LLD-GWY-FNC-7452  H398-LLD-GWY-FNC-7453  H398-LLD-GWY-FNC-7454  H398-LLD-GWY-FNC-7455  H398-LLD-GWY-FNC-7456  H398-LLD-GWY-FNC-7457  H398-LLD-GWY-FNC-7458  H398-LLD-GWY-FNC-7460  H398-LLD-GWY-FNC-7461  H398-LLD-GWY-FNC-7462  H398-LLD-GWY-FNC-7463  H398-LLD-GWY-FNC-7464  H398-LLD-GWY-FNC-7465  H398-LLD-GWY-FNC-7466  H398-LLD-GWY-FNC-7467  H398-LLD-GWY-FNC-7468  H398-LLD-GWY-FNC-7469  H398-LLD-GWY-FNC-7470  The below requirements got deleted  H398-LLD-GWY-FNC-6188  H398-LLD-GWY-FNC-6189  H398-LLD-GWY-FNC-5810  H398-LLD-GWY-FNC-5811  H398-LLD-GWY-FNC-5812  H398-LLD-GWY-FNC-5813  H398-LLD-GWY-FNC-7403  H398-LLD-GWY-FNC-7405  H398-LLD-GWY-FNC-20  H398-LLD-GWY-FNC-21  H398-LLD-GWY-FNC-22  H398-LLD-GWY-FNC-23  H398-LLD-GWY-FNC-24  H398-LLD-GWY-FNC-6022  H398-LLD-GWY-FNC-6031  H398-LLD-GWY-FNC-6040  H398-LLD-GWY-FNC-6041  H398-LLD-GWY-FNC-6042  H398-LLD-GWY-FNC-6051  H398-LLD-GWY-FNC-6060  H398-LLD-GWY-FNC-6061  H398-LLD-GWY-FNC-6222  H398-LLD-GWY-FNC-6190  H398-LLD-GWY-FNC-6246  H398-LLD-GWY-FNC-6247  H398-LLD-GWY-FNC-6082  H398-LLD-GWY-FNC-6083  H398-LLD-GWY-FNC-6249  H398-LLD-GWY-FNC-6250  H398-LLD-GWY-FNC-1765  H398-LLD-GWY-FNC-2031  H398-LLD-GWY-FNC-2032  H398-LLD-GWY-FNC-2033  H398-LLD-GWY-FNC-2034  H398-LLD-GWY-FNC-2035  H398-LLD-GWY-FNC-2036  H398-LLD-GWY-FNC-2037  H398-LLD-GWY-FNC-2038  H398-LLD-GWY-FNC-2039  H398-LLD-GWY-FNC-2040  H398-LLD-GWY-FNC-5817  H398-LLD-GWY-FNC-2041  H398-LLD-GWY-FNC-2042  H398-LLD-GWY-FNC-2051  H398-LLD-GWY-FNC-2052  H398-LLD-GWY-FNC-2053  H398-LLD-GWY-FNC-2054  H398-LLD-GWY-FNC-2055  H398-LLD-GWY-FNC-2056  H398-LLD-GWY-FNC-2057  H398-LLD-GWY-FNC-2058  H398-LLD-GWY-FNC-2059  H398-LLD-GWY-FNC-2060  H398-LLD-GWY-FNC-2061  H398-LLD-GWY-FNC-2062  H398-LLD-GWY-FNC-7332  H398-LLD-GWY-FNC-7333  H398-LLD-GWY-FNC-2071  H398-LLD-GWY-FNC-2073  H398-LLD-GWY-FNC-2074  H398-LLD-GWY-FNC-2075  H398-LLD-GWY-FNC-2076  H398-LLD-GWY-FNC-2077  H398-LLD-GWY-FNC-2078  H398-LLD-GWY-FNC-2079  H398-LLD-GWY-FNC-2080  H398-LLD-GWY-FNC-2081  H398-LLD-GWY-FNC-2082  H398-LLD-GWY-FNC-5826  H398-LLD-GWY-FNC-5827  H398-LLD-GWY-FNC-5828  H398-LLD-GWY-FNC-5829  H398-LLD-GWY-FNC-5830  H398-LLD-GWY-FNC-5831  H398-LLD-GWY-FNC-7334  H398-LLD-GWY-FNC-7335  H398-LLD-GWY-FNC-6092  H398-LLD-GWY-FNC-6093  H398-LLD-GWY-FNC-6094  H398-LLD-GWY-FNC-6095  H398-LLD-GWY-FNC-6096  H398-LLD-GWY-FNC-6097  H398-LLD-GWY-FNC-7336  H398-LLD-GWY-FNC-5840  H398-LLD-GWY-FNC-5841  H398-LLD-GWY-FNC-2114  H398-LLD-GWY-FNC-2115  H398-LLD-GWY-FNC-5842  H398-LLD-GWY-FNC-5844  H398-LLD-GWY-FNC-5845  H398-LLD-GWY-FNC-5846  H398-LLD-GWY-FNC-5847  H398-LLD-GWY-FNC-5989  H398-LLD-GWY-FNC-5849  H398-LLD-GWY-FNC-5858  H398-LLD-GWY-FNC-5859  H398-LLD-GWY-FNC-5860  H398-LLD-GWY-FNC-2148  H398-LLD-GWY-FNC-2223  H398-LLD-GWY-FNC-2241  H398-LLD-GWY-FNC-2242  H398-LLD-GWY-FNC-2252  H398-LLD-GWY-FNC-2253  H398-LLD-GWY-FNC-2254  H398-LLD-GWY-FNC-2255  H398-LLD-GWY-FNC-5951  H398-LLD-GWY-DRQ-5868  H398-LLD-GWY-FNC-2639  H398-LLD-GWY-FNC-2640  H398-LLD-GWY-FNC-2649  H398-LLD-GWY-FNC-2650  H398-LLD-GWY-FNC-2659  H398-LLD-GWY-FNC-2660  H398-LLD-GWY-FNC-2661  H398-LLD-GWY-FNC-2670  H398-LLD-GWY-FNC-5881  H398-LLD-GWY-FNC-5882  H398-LLD-GWY-FNC-5883  H398-LLD-GWY-FNC-5884  H398-LLD-GWY-FNC-5893  H398-LLD-GWY-FNC-5894  H398-LLD-GWY-FNC-2688  H398-LLD-GWY-FNC-2689  H398-LLD-GWY-FNC-6209  H398-LLD-GWY-FNC-2690  H398-LLD-GWY-FNC-2691  H398-LLD-GWY-FNC-2700  H398-LLD-GWY-FNC-2709  H398-LLD-GWY-FNC-2718  H398-LLD-GWY-FNC-2727  H398-LLD-GWY-FNC-2736  H398-LLD-GWY-FNC-2745  H398-LLD-GWY-FNC-2746  H398-LLD-GWY-FNC-6282  H398-LLD-GWY-FNC-6283  H398-LLD-GWY-FNC-6284  H398-LLD-GWY-FNC-6285  H398-LLD-GWY-FNC-6286  H398-LLD-GWY-FNC-2755  H398-LLD-GWY-FNC-2764  H398-LLD-GWY-FNC-2765  H398-LLD-GWY-FNC-2774  H398-LLD-GWY-FNC-6287  H398-LLD-GWY-FNC-6288  H398-LLD-GWY-FNC-2783  H398-LLD-GWY-FNC-2792  H398-LLD-GWY-FNC-2801  H398-LLD-GWY-FNC-2802  H398-LLD-GWY-FNC-6289  H398-LLD-GWY-FNC-6290  H398-LLD-GWY-FNC-6291  H398-LLD-GWY-FNC-6292  H398-LLD-GWY-FNC-2803  H398-LLD-GWY-FNC-6296  H398-LLD-GWY-FNC-6297  H398-LLD-GWY-FNC-6298  H398-LLD-GWY-FNC-2804  H398-LLD-GWY-FNC-2805  H398-LLD-GWY-FNC-2814  H398-LLD-GWY-FNC-2815  H398-LLD-GWY-FNC-2816  H398-LLD-GWY-FNC-2817  H398-LLD-GWY-FNC-2818  H398-LLD-GWY-FNC-2819  H398-LLD-GWY-FNC-2828  H398-LLD-GWY-FNC-2837  H398-LLD-GWY-FNC-2846  H398-LLD-GWY-FNC-2855  H398-LLD-GWY-FNC-2864  H398-LLD-GWY-FNC-2873  H398-LLD-GWY-FNC-2882  H398-LLD-GWY-FNC-2883  H398-LLD-GWY-FNC-2884  H398-LLD-GWY-FNC-2893  H398-LLD-GWY-FNC-6299  H398-LLD-GWY-FNC-6300  H398-LLD-GWY-FNC-6301  H398-LLD-GWY-FNC-6302  H398-LLD-GWY-FNC-5952  H398-LLD-GWY-FNC-6303  H398-LLD-GWY-FNC-6304  H398-LLD-GWY-FNC-6305  H398-LLD-GWY-FNC-5953  H398-LLD-GWY-FNC-6396  H398-LLD-GWY-FNC-6395  H398-LLD-GWY-FNC-2902  H398-LLD-GWY-FNC-6306  H398-LLD-GWY-FNC-6307  H398-LLD-GWY-FNC-6308  H398-LLD-GWY-FNC-6309  H398-LLD-GWY-FNC-6310  H398-LLD-GWY-FNC-6311  H398-LLD-GWY-FNC-6312  H398-LLD-GWY-FNC-5954  H398-LLD-GWY-FNC-6313  H398-LLD-GWY-FNC-6314  H398-LLD-GWY-FNC-6315  H398-LLD-GWY-FNC-6316  H398-LLD-GWY-FNC-6317  H398-LLD-GWY-FNC-6318  H398-LLD-GWY-FNC-5955  H398-LLD-GWY-FNC-6319  H398-LLD-GWY-FNC-6320  H398-LLD-GWY-FNC-6321  H398-LLD-GWY-FNC-5956  H398-LLD-GWY-FNC-2911  H398-LLD-GWY-FNC-6398  H398-LLD-GWY-FNC-6397  H398-LLD-GWY-FNC-2920  H398-LLD-GWY-FNC-6322  H398-LLD-GWY-FNC-6323  H398-LLD-GWY-FNC-6324  H398-LLD-GWY-FNC-6325  H398-LLD-GWY-FNC-6326  H398-LLD-GWY-FNC-6327  H398-LLD-GWY-FNC-6328  H398-LLD-GWY-FNC-5957  H398-LLD-GWY-FNC-6332  H398-LLD-GWY-FNC-6333  H398-LLD-GWY-FNC-6334  H398-LLD-GWY-FNC-6335  H398-LLD-GWY-FNC-5958  H398-LLD-GWY-FNC-6336  H398-LLD-GWY-FNC-6337  H398-LLD-GWY-FNC-6338  H398-LLD-GWY-FNC-5959  H398-LLD-GWY-FNC-2929  H398-LLD-GWY-FNC-2930  H398-LLD-GWY-FNC-2931  H398-LLD-GWY-FNC-6399  H398-LLD-GWY-FNC-6400  H398-LLD-GWY-FNC-6401  H398-LLD-GWY-FNC-6210  H398-LLD-GWY-FNC-2940  H398-LLD-GWY-FNC-2949  H398-LLD-GWY-FNC-2958  H398-LLD-GWY-FNC-2967  H398-LLD-GWY-FNC-2976  H398-LLD-GWY-FNC-2985  H398-LLD-GWY-FNC-2994  H398-LLD-GWY-FNC-3003  H398-LLD-GWY-FNC-3012  H398-LLD-GWY-FNC-6414  H398-LLD-GWY-FNC-6423  H398-LLD-GWY-FNC-3030  H398-LLD-GWY-FNC-6339  H398-LLD-GWY-FNC-6340  H398-LLD-GWY-FNC-6341  H398-LLD-GWY-FNC-3021  H398-LLD-GWY-FNC-3039  H398-LLD-GWY-FNC-3048  H398-LLD-GWY-FNC-3057  H398-LLD-GWY-FNC-3066  H398-LLD-GWY-FNC-3067  H398-LLD-GWY-FNC-3068  H398-LLD-GWY-FNC-3077  H398-LLD-GWY-FNC-3086  H398-LLD-GWY-FNC-6342  H398-LLD-GWY-FNC-6343  H398-LLD-GWY-FNC-6344  H398-LLD-GWY-FNC-3095  H398-LLD-GWY-FNC-3096  H398-LLD-GWY-FNC-3105  H398-LLD-GWY-FNC-6345  H398-LLD-GWY-FNC-6346  H398-LLD-GWY-FNC-6347  H398-LLD-GWY-FNC-3106  H398-LLD-GWY-FNC-3115  H398-LLD-GWY-FNC-3124  H398-LLD-GWY-FNC-6402  H398-LLD-GWY-FNC-6403  H398-LLD-GWY-FNC-3133  H398-LLD-GWY-FNC-3142  H398-LLD-GWY-FNC-3143  H398-LLD-GWY-FNC-3144  H398-LLD-GWY-FNC-3145  H398-LLD-GWY-FNC-3146  H398-LLD-GWY-FNC-3147  H398-LLD-GWY-FNC-3148  H398-LLD-GWY-FNC-3149  H398-LLD-GWY-FNC-3158  H398-LLD-GWY-FNC-6348  H398-LLD-GWY-FNC-6349  H398-LLD-GWY-FNC-6350  H398-LLD-GWY-FNC-6351  H398-LLD-GWY-FNC-6352  H398-LLD-GWY-FNC-3167  H398-LLD-GWY-FNC-3168  H398-LLD-GWY-FNC-3169  H398-LLD-GWY-FNC-6354  H398-LLD-GWY-FNC-6355  H398-LLD-GWY-FNC-3178  H398-LLD-GWY-FNC-3179  H398-LLD-GWY-FNC-3180  H398-LLD-GWY-FNC-3181  H398-LLD-GWY-FNC-3182  H398-LLD-GWY-FNC-3183  H398-LLD-GWY-FNC-3184  H398-LLD-GWY-FNC-3193  H398-LLD-GWY-FNC-6356  H398-LLD-GWY-FNC-6357  H398-LLD-GWY-FNC-6358  H398-LLD-GWY-FNC-6359  H398-LLD-GWY-FNC-3202  H398-LLD-GWY-FNC-3211  H398-LLD-GWY-FNC-3220  H398-LLD-GWY-FNC-2520  H398-LLD-GWY-FNC-2529  H398-LLD-GWY-FNC-2530  H398-LLD-GWY-FNC-5960  H398-LLD-GWY-FNC-5961  H398-LLD-GWY-FNC-5962  H398-LLD-GWY-FNC-5963  H398-LLD-GWY-FNC-5964  H398-LLD-GWY-FNC-5965  H398-LLD-GWY-FNC-5966  H398-LLD-GWY-FNC-5967  H398-LLD-GWY-FNC-5968  H398-LLD-GWY-FNC-5969  H398-LLD-GWY-FNC-5970  H398-LLD-GWY-FNC-5971  H398-LLD-GWY-FNC-5972  H398-LLD-GWY-FNC-5973  H398-LLD-GWY-FNC-5974  H398-LLD-GWY-FNC-5975  H398-LLD-GWY-FNC-5976  H398-LLD-GWY-FNC-5977  H398-LLD-GWY-FNC-5978  H398-LLD-GWY-FNC-5979  H398-LLD-GWY-FNC-2539  H398-LLD-GWY-FNC-6265  H398-LLD-GWY-FNC-6187  H398-LLD-GWY-FNC-2548  H398-LLD-GWY-FNC-2557  H398-LLD-GWY-FNC-6268  H398-LLD-GWY-FNC-2566  H398-LLD-GWY-FNC-2567  H398-LLD-GWY-FNC-2576  H398-LLD-GWY-FNC-2585  H398-LLD-GWY-FNC-2594  H398-LLD-GWY-FNC-2603  H398-LLD-GWY-FNC-2612  H398-LLD-GWY-FNC-2621  H398-LLD-GWY-FNC-2630  H398-LLD-GWY-FNC-2232  H398-LLD-GWY-FNC-2266  H398-LLD-GWY-FNC-2275  H398-LLD-GWY-FNC-2276  H398-LLD-GWY-FNC-2285  H398-LLD-GWY-FNC-6269  H398-LLD-GWY-FNC-6270  H398-LLD-GWY-FNC-6271  H398-LLD-GWY-FNC-2294  H398-LLD-GWY-FNC-2295  H398-LLD-GWY-FNC-2296  H398-LLD-GWY-FNC-2297  H398-LLD-GWY-FNC-2306  H398-LLD-GWY-FNC-2307  H398-LLD-GWY-FNC-2308  H398-LLD-GWY-FNC-2309  H398-LLD-GWY-FNC-2318  H398-LLD-GWY-FNC-2319  H398-LLD-GWY-FNC-2320  H398-LLD-GWY-FNC-2329  H398-LLD-GWY-FNC-2330  H398-LLD-GWY-FNC-2331  H398-LLD-GWY-FNC-2332  H398-LLD-GWY-FNC-2341  H398-LLD-GWY-FNC-2350  H398-LLD-GWY-FNC-2351  H398-LLD-GWY-FNC-2360  H398-LLD-GWY-FNC-6274  H398-LLD-GWY-FNC-6275  H398-LLD-GWY-FNC-2361  H398-LLD-GWY-FNC-6276  H398-LLD-GWY-FNC-6277  H398-LLD-GWY-FNC-2362  H398-LLD-GWY-FNC-2363  H398-LLD-GWY-FNC-2364  H398-LLD-GWY-FNC-6278  H398-LLD-GWY-FNC-2365  H398-LLD-GWY-FNC-2374  H398-LLD-GWY-FNC-2383  H398-LLD-GWY-FNC-2392  H398-LLD-GWY-FNC-2401  H398-LLD-GWY-FNC-2410  H398-LLD-GWY-FNC-2419  H398-LLD-GWY-FNC-2428  H398-LLD-GWY-FNC-2437  H398-LLD-GWY-FNC-2446  H398-LLD-GWY-FNC-2455  H398-LLD-GWY-FNC-2464  H398-LLD-GWY-FNC-2473  H398-LLD-GWY-FNC-2474  H398-LLD-GWY-FNC-2475  H398-LLD-GWY-FNC-2484  H398-LLD-GWY-FNC-6279  H398-LLD-GWY-FNC-6280  H398-LLD-GWY-FNC-6281  H398-LLD-GWY-FNC-2493  H398-LLD-GWY-FNC-6424  H398-LLD-GWY-FNC-6425  H398-LLD-GWY-FNC-6426  H398-LLD-GWY-FNC-2502  H398-LLD-GWY-FNC-5869  H398-LLD-GWY-FNC-5870  H398-LLD-GWY-FNC-5871  H398-LLD-GWY-FNC-5872  H398-LLD-GWY-FNC-3230  H398-LLD-GWY-FNC-6206  H398-LLD-GWY-FNC-6207  H398-LLD-GWY-FNC-5905  H398-LLD-GWY-FNC-3239  H398-LLD-GWY-FNC-3240  H398-LLD-GWY-FNC-3241  H398-LLD-GWY-FNC-3242  H398-LLD-GWY-FNC-3243  H398-LLD-GWY-FNC-3244  H398-LLD-GWY-FNC-3245  H398-LLD-GWY-FNC-3246  H398-LLD-GWY-FNC-3247  H398-LLD-GWY-FNC-3248  H398-LLD-GWY-FNC-3257  H398-LLD-GWY-FNC-3258  H398-LLD-GWY-FNC-5781  H398-LLD-GWY-FNC-3259  H398-LLD-GWY-FNC-3260  H398-LLD-GWY-FNC-3261  H398-LLD-GWY-FNC-3262  H398-LLD-GWY-FNC-3271  H398-LLD-GWY-FNC-3280  H398-LLD-GWY-FNC-3281  H398-LLD-GWY-FNC-3290  H398-LLD-GWY-FNC-3291  H398-LLD-GWY-FNC-3300  H398-LLD-GWY-FNC-3301  H398-LLD-GWY-FNC-3302  H398-LLD-GWY-FNC-3303  H398-LLD-GWY-FNC-3312  H398-LLD-GWY-FNC-6360  H398-LLD-GWY-FNC-6361  H398-LLD-GWY-FNC-6362  H398-LLD-GWY-FNC-5907  H398-LLD-GWY-FNC-3313  H398-LLD-GWY-FNC-3314  H398-LLD-GWY-FNC-3315  H398-LLD-GWY-FNC-6363  H398-LLD-GWY-FNC-6364  H398-LLD-GWY-FNC-3324  H398-LLD-GWY-FNC-6365  H398-LLD-GWY-FNC-6366  H398-LLD-GWY-FNC-6367  H398-LLD-GWY-FNC-6368  H398-LLD-GWY-FNC-6369  H398-LLD-GWY-FNC-5909  H398-LLD-GWY-FNC-3325  H398-LLD-GWY-FNC-3326  H398-LLD-GWY-FNC-3327  H398-LLD-GWY-FNC-3328  H398-LLD-GWY-FNC-3329  H398-LLD-GWY-FNC-3330  H398-LLD-GWY-FNC-5774  H398-LLD-GWY-FNC-5775  H398-LLD-GWY-FNC-5776  H398-LLD-GWY-FNC-6370  H398-LLD-GWY-FNC-6371  H398-LLD-GWY-FNC-3339  H398-LLD-GWY-FNC-3340  H398-LLD-GWY-FNC-3341  H398-LLD-GWY-FNC-3342  H398-LLD-GWY-FNC-3351  H398-LLD-GWY-FNC-5777  H398-LLD-GWY-FNC-3352  H398-LLD-GWY-FNC-3353  H398-LLD-GWY-FNC-3362  H398-LLD-GWY-FNC-3363  H398-LLD-GWY-FNC-3364  H398-LLD-GWY-FNC-3373  H398-LLD-GWY-FNC-3374  H398-LLD-GWY-FNC-3375  H398-LLD-GWY-FNC-3376  H398-LLD-GWY-FNC-3385  H398-LLD-GWY-FNC-3386  H398-LLD-GWY-FNC-3387  H398-LLD-GWY-FNC-3388  H398-LLD-GWY-FNC-3389  H398-LLD-GWY-FNC-3390  H398-LLD-GWY-FNC-3391  H398-LLD-GWY-FNC-3392  H398-LLD-GWY-FNC-3393  H398-LLD-GWY-FNC-3394  H398-LLD-GWY-FNC-3395  H398-LLD-GWY-FNC-3396  H398-LLD-GWY-FNC-3405  H398-LLD-GWY-FNC-3406  H398-LLD-GWY-FNC-3415  H398-LLD-GWY-FNC-3424  H398-LLD-GWY-FNC-3425  H398-LLD-GWY-FNC-3426  H398-LLD-GWY-FNC-3427  H398-LLD-GWY-FNC-3428  H398-LLD-GWY-FNC-3429  H398-LLD-GWY-FNC-3430  H398-LLD-GWY-FNC-3431  H398-LLD-GWY-FNC-3432  H398-LLD-GWY-FNC-3433  H398-LLD-GWY-FNC-3434  H398-LLD-GWY-FNC-3435  H398-LLD-GWY-FNC-3436  H398-LLD-GWY-FNC-3437  H398-LLD-GWY-FNC-3446  H398-LLD-GWY-FNC-3455  H398-LLD-GWY-FNC-3456  H398-LLD-GWY-FNC-5910  H398-LLD-GWY-FNC-3465  H398-LLD-GWY-FNC-3466  H398-LLD-GWY-FNC-6116  H398-LLD-GWY-FNC-3475  H398-LLD-GWY-FNC-3476  H398-LLD-GWY-FNC-3477  H398-LLD-GWY-FNC-3478  H398-LLD-GWY-FNC-3487  H398-LLD-GWY-FNC-3496  H398-LLD-GWY-FNC-3505  H398-LLD-GWY-FNC-3506  H398-LLD-GWY-FNC-3507  H398-LLD-GWY-FNC-3508  H398-LLD-GWY-FNC-3517  H398-LLD-GWY-FNC-3518  H398-LLD-GWY-FNC-3519  H398-LLD-GWY-FNC-3520  H398-LLD-GWY-FNC-3521  H398-LLD-GWY-FNC-3530  H398-LLD-GWY-FNC-3539  H398-LLD-GWY-FNC-6389  H398-LLD-GWY-FNC-6390  H398-LLD-GWY-FNC-6391  H398-LLD-GWY-FNC-6392  H398-LLD-GWY-FNC-6393  H398-LLD-GWY-FNC-6394  H398-LLD-GWY-FNC-3548  H398-LLD-GWY-FNC-3557  H398-LLD-GWY-FNC-3558  H398-LLD-GWY-FNC-3567  H398-LLD-GWY-FNC-5944  H398-LLD-GWY-FNC-5945  H398-LLD-GWY-FNC-3576  H398-LLD-GWY-FNC-5946  H398-LLD-GWY-FNC-3577  H398-LLD-GWY-FNC-3578  H398-LLD-GWY-FNC-3579  H398-LLD-GWY-FNC-3580  H398-LLD-GWY-FNC-3581  H398-LLD-GWY-FNC-3583  H398-LLD-GWY-FNC-3584  H398-LLD-GWY-FNC-3585  H398-LLD-GWY-FNC-3586  H398-LLD-GWY-FNC-5911  H398-LLD-GWY-FNC-3587  H398-LLD-GWY-FNC-3588  H398-LLD-GWY-FNC-5947  H398-LLD-GWY-FNC-5948  H398-LLD-GWY-FNC-5782  H398-LLD-GWY-FNC-5783  H398-LLD-GWY-FNC-5949  H398-LLD-GWY-FNC-3597  H398-LLD-GWY-FNC-3598  H398-LLD-GWY-FNC-3600  H398-LLD-GWY-FNC-3601  H398-LLD-GWY-FNC-3602  H398-LLD-GWY-FNC-3603  H398-LLD-GWY-FNC-5913  H398-LLD-GWY-FNC-3604  H398-LLD-GWY-FNC-5950  H398-LLD-GWY-FNC-5922  H398-LLD-GWY-FNC-5923  H398-LLD-GWY-FNC-5924  H398-LLD-GWY-FNC-5925  H398-LLD-GWY-FNC-5926  H398-LLD-GWY-FNC-5927  H398-LLD-GWY-FNC-5928  H398-LLD-GWY-FNC-5929  H398-LLD-GWY-FNC-3613  H398-LLD-GWY-FNC-3614  H398-LLD-GWY-FNC-3615  H398-LLD-GWY-FNC-3624  H398-LLD-GWY-FNC-3625  H398-LLD-GWY-FNC-3626  H398-LLD-GWY-FNC-3627  H398-LLD-GWY-FNC-3628  H398-LLD-GWY-FNC-3629  H398-LLD-GWY-FNC-3630  H398-LLD-GWY-FNC-3631  H398-LLD-GWY-FNC-3632  H398-LLD-GWY-FNC-3633  H398-LLD-GWY-FNC-3634  H398-LLD-GWY-FNC-3635  H398-LLD-GWY-FNC-3636  H398-LLD-GWY-FNC-3637  H398-LLD-GWY-FNC-3638  H398-LLD-GWY-FNC-3647  H398-LLD-GWY-FNC-5930  H398-LLD-GWY-FNC-6118  H398-LLD-GWY-FNC-3648  H398-LLD-GWY-FNC-5931  H398-LLD-GWY-DRQ-3657  H398-LLD-GWY-FNC-5940  H398-LLD-GWY-FNC-6372  H398-LLD-GWY-FNC-6373  H398-LLD-GWY-FNC-5941  H398-LLD-GWY-FNC-6374  H398-LLD-GWY-FNC-6375  H398-LLD-GWY-FNC-6376  H398-LLD-GWY-FNC-6377  H398-LLD-GWY-FNC-6378  H398-LLD-GWY-FNC-5942  H398-LLD-GWY-FNC-6379  H398-LLD-GWY-FNC-6380  H398-LLD-GWY-FNC-6381  H398-LLD-GWY-FNC-6382  H398-LLD-GWY-FNC-6383  H398-LLD-GWY-FNC-6384  H398-LLD-GWY-FNC-5943  H398-LLD-GWY-FNC-5988  H398-LLD-GWY-FNC-7117  The below sections are updated as per self review  10.1.2.6  10.1.7.3  10.21.23.3  10.24.22.3  10.24.26.3  10.24.37.3  10.41.37.3  10.41.38.3  10.41.39.3  10.41.39.6  10.41.40.3  10.41.40.6  10.41.41.3  10.41.41.36  10.41.59.3  10.41.59.6  10.41.60.3  10.41.60.6  10.41.82.3 | PR100224 | SRAVANTHI MUPPINENI | 05-07-2024 |
| 2.4 | The below requirements are updated as per review comments H398-LLD-GWY-FNC-6445  H398-LLD-GWY-FNC-7453  H398-LLD-GWY-FNC-7456  The below requirements are updated as per self review  H398-LLD-GWY-FNC-2145  H398-LLD-GWY-FNC-5765  H398-LLD-GWY-FNC-442  H398-LLD-GWY-FNC-2146  H398-LLD-GWY-FNC-2214  H398-LLD-GWY-FNC-6495  H398-LLD-GWY-FNC-6620  H398-LLD-GWY-FNC-6684  H398-LLD-GWY-FNC-7108  H398-LLD-GWY-FNC-7198  H398-LLD-GWY-FNC-7120  H398-LLD-GWY-FNC-5762  H398-LLD-GWY-FNC-5785  H398-LLD-GWY-FNC-6619  H398-LLD-GWY-FNC-7469  H398-LLD-GWY-FNC-6641  H398-LLD-GWY-FNC-6838  H398-LLD-GWY-FNC-6839  H398-LLD-GWY-FNC-518  H398-LLD-GWY-FNC-5843  H398-LLD-GWY-FNC-1760  H398-LLD-GWY-FNC-1776  H398-LLD-GWY-FNC-5752  H398-LLD-GWY-FNC-5755  H398-LLD-GWY-FNC-7325  H398-LLD-GWY-FNC-7326  H398-LLD-GWY-FNC-7327  H398-LLD-GWY-FNC-7328  H398-LLD-GWY-FNC-7329  H398-LLD-GWY-FNC-7330  H398-LLD-GWY-FNC-7331  H398-LLD-GWY-FNC-6762  H398-LLD-GWY-FNC-6771  H398-LLD-GWY-FNC-6944  H398-LLD-GWY-FNC-7084  H398-LLD-GWY-FNC-7161  H398-LLD-GWY-FNC-7259  H398-LLD-GWY-FNC-7357  H398-LLD-GWY-FNC-7316  H398-LLD-GWY-FNC-570  H398-LLD-GWY-FNC-7067  H398-LLD-GWY-FNC-7093  H398-LLD-GWY-FNC-7470  H398-LLD-GWY-FNC-881  H398-LLD-GWY-FNC-2097  H398-LLD-GWY-FNC-2100  H398-LLD-GWY-FNC-7413  H398-LLD-GWY-FNC-7444  H398-LLD-GWY-FNC-7445  H398-LLD-GWY-FNC-7068  H398-LLD-GWY-FNC-7452  H398-LLD-GWY-FNC-7454  H398-LLD-GWY-FNC-7410  The below requirements are added  H398-LLD-GWY-FNC-7471  H398-LLD-GWY-FNC-7472  H398-LLD-GWY-FNC-7473  H398-LLD-GWY-FNC-7474  H398-LLD-GWY-FNC-7475  The below requirements are deleted  H398-LLD-GWY-FNC-6080  H398-LLD-GWY-FNC-6248  H398-LLD-GWY-FNC-2193  H398-LLD-GWY-FNC-2194  H398-LLD-GWY-FNC-2195  H398-LLD-GWY-FNC-7268  H398-LLD-GWY-FNC-7393  H398-LLD-GWY-FNC-2192  H398-LLD-GWY-FNC-6660  H398-LLD-GWY-FNC-6661  H398-LLD-GWY-FNC-7079  H398-LLD-GWY-FNC-376  H398-LLD-GWY-FNC-385  H398-LLD-GWY-FNC-1988  H398-LLD-GWY-FNC-1989  H398-LLD-GWY-FNC-1990  H398-LLD-GWY-FNC-1991  H398-LLD-GWY-FNC-1992  H398-LLD-GWY-FNC-1993  H398-LLD-GWY-FNC-7053  H398-LLD-GWY-FNC-882  H398-LLD-GWY-FNC-883 | PR100224 | SRAVANTHI MUPPINENI | 13-08-2024 |
| 2.5 | The below requirements are updated as per review comments  H398-LLD-GWY-FNC-7067  H398-LLD-GWY-FNC-7316  H398-LLD-GWY-FNC-7259  H398-LLD-GWY-FNC-7161  H398-LLD-GWY-FNC-7325  H398-LLD-GWY-FNC-7326  H398-LLD-GWY-FNC-7327  H398-LLD-GWY-FNC-7328  H398-LLD-GWY-FNC-7329  H398-LLD-GWY-FNC-7330  H398-LLD-GWY-FNC-7331  H398-LLD-GWY-FNC-5752  H398-LLD-GWY-FNC-5843  H398-LLD-GWY-FNC-6838  H398-LLD-GWY-FNC-7198  H398-LLD-GWY-FNC-7207  H398-LLD-GWY-FNC-7210  H398-LLD-GWY-FNC-7219  H398-LLD-GWY-FNC-7228  H398-LLD-GWY-FNC-7108  H398-LLD-GWY-FNC-6684  H398-LLD-GWY-FNC-7354 | PR100224 | SRAVANTHI MUPPINENI | 16-08-2024 |
| 2.6 | The below requirements are updated as QA review comments  H398-LLD-GWY-FNC-7161 | PR100224 | SRAVANTHI MUPPINENI | 20-08-2024 |
| 2.7 | The below requirements are updated as per self review  H398-LLD-GWY-FNC-6697  H398-LLD-GWY-FNC-7451  H398-LLD-GWY-FNC-6695  H398-LLD-GWY-FNC-6781  H398-LLD-GWY-FNC-6782  H398-LLD-GWY-FNC-6795  H398-LLD-GWY-FNC-6809  H398-LLD-GWY-FNC-6819  H398-LLD-GWY-FNC-6829  H398-LLD-GWY-FNC-6840  H398-LLD-GWY-FNC-6849  H398-LLD-GWY-FNC-6850  H398-LLD-GWY-FNC-7010  H398-LLD-GWY-FNC-7014  H398-LLD-GWY-FNC-7023  H398-LLD-GWY-FNC-7024  H398-LLD-GWY-FNC-7033  H398-LLD-GWY-FNC-7043  H398-LLD-GWY-FNC-7108  H398-LLD-GWY-FNC-7121 | PR100257 | SRAVANTHI MUPPINENI | 23-08-2024 |
| 2.8 | The below requirements are updated as per self review  H398-LLD-GWY-FNC-7463 | PR100275 | SRAVANTHI MUPPINENI | 28-08-2024 |
| 2.9 | The below requirements are updated as per self review  H398-LLD-GWY-FNC-7452   Section 14 Appendix B: Data Constants | PR100327 | SRAVANTHI MUPPINENI | 10-10-2024 |
| 2.10 | Updated front page company name as  "ALTEN Global Technologies Private Limited" as per QA comment  The below requirements are updated as per QA comment:  H398-LLD-GWY-FNC-25 | PR100327 | SRAVANTHI MUPPINENI | 14-10-2024 |
| 2.11 | The below requirements are added :  H398-LLD-GWY-FNC-7476  H398-LLD-GWY-FNC-7478  H398-LLD-GWY-FNC-7479  H398-LLD-GWY-FNC-7482  H398-LLD-GWY-FNC-7483  H398-LLD-GWY-FNC-7477  H398-LLD-GWY-FNC-7480  H398-LLD-GWY-FNC-7481  H398-LLD-GWY-FNC-7484  H398-LLD-GWY-FNC-7485  H398-LLD-GWY-FNC-7486  H398-LLD-GWY-FNC-7487  The below requirement is deleted:  H398-LLD-GWY-FNC-7404  The below requirements are updated:  H398-LLD-GWY-FNC-6444  H398-LLD-GWY-FNC-6448  H398-LLD-GWY-FNC-6445  H398-LLD-GWY-FNC-6517  H398-LLD-GWY-FNC-6518  H398-LLD-GWY-FNC-6695  H398-LLD-GWY-FNC-6696  H398-LLD-GWY-FNC-7277  H398-LLD-GWY-FNC-7067  H398-LLD-GWY-FNC-7068  H398-LLD-GWY-FNC-7093  H398-LLD-GWY-FNC-7094  H398-LLD-GWY-FNC-7095  H398-LLD-GWY-FNC-7219  H398-LLD-GWY-FNC-2147  H398-LLD-GWY-FNC-316  H398-LLD-GWY-FNC-5752  Section 13 Appendix A : The Constant data table, Gateway MCD software.  Section 14 Appendix B : Data Constants, Gateway application software. | PR100356 | Madhumitha M | 20-06-2025 |
| 2.12 | The below requirements are added:  H398-LLD-GWY-FNC-7488  H398-LLD-GWY-FNC-7489  H398-LLD-GWY-FNC-7490  The below requirement is deleted:  H398-LLD-GWY-FNC-546  The below requirements are Modified:  H398-LLD-GWY-FNC-6115  H398-LLD-GWY-FNC-6444  H398-LLD-GWY-FNC-6445  H398-LLD-GWY-FNC-6448  H398-LLD-GWY-FNC-6809  H398-LLD-GWY-FNC-6818  H398-LLD-GWY-FNC-7056  H398-LLD-GWY-FNC-7067  H398-LLD-GWY-FNC-7082  H398-LLD-GWY-FNC-7093  H398-LLD-GWY-FNC-7480  H398-LLD-GWY-FNC-7238  H398-LLD-GWY-FNC-7239  H398-LLD-GWY-FNC-7237  H398-LLD-GWY-FNC-6990  H398-LLD-GWY-FNC-6999  H398-LLD-GWY-FNC-6683  H398-LLD-GWY-FNC-6682  H398-LLD-GWY-FNC-6676  H398-LLD-GWY-FNC-6678  H398-LLD-GWY-FNC-6677  H398-LLD-GWY-FNC-7469  H398-LLD-GWY-FNC-7470  H398-LLD-GWY-FNC-7132  H398-LLD-GWY-FNC-7130  H398-LLD-GWY-FNC-7131  H398-LLD-GWY-FNC-7142  H398-LLD-GWY-FNC-7161  H398-LLD-GWY-FNC-7171  H398-LLD-GWY-FNC-7180  H398-LLD-GWY-FNC-7189  H398-LLD-GWY-FNC-7068  H398-LLD-GWY-FNC-7151  H398-LLD-GWY-FNC-6744  H398-LLD-GWY-FNC-6819  H398-LLD-GWY-FNC-6679  H398-LLD-GWY-FNC-6680  H398-LLD-GWY-FNC-6681  H398-LLD-GWY-FNC-6684  H398-LLD-GWY-FNC-7300  H398-LLD-GWY-FNC-7259  H398-LLD-GWY-FNC-7406  H398-LLD-GWY-FNC-7108  H398-LLD-GWY-FNC-7402  H398-LLD-GWY-FNC-7478  H398-LLD-GWY-FNC-7479  H398-LLD-GWY-FNC-7487  H398-LLD-GWY-FNC-7482  H398-LLD-GWY-FNC-7481  H398-LLD-GWY-FNC-7483  H398-LLD-GWY-FNC-7094  H398-LLD-GWY-FNC-6619  H398-LLD-GWY-FNC-6620  Section 10.41.56.1 is updated  Section 14 Appendix B: Data Constants, Gateway application software is updated | PR100364 | Madhumitha M | 25-07-2025 |
| 2.13 | The below requirements are Modified:  H398-LLD-GWY-FNC-6696  H398-LLD-GWY-FNC-6698  Section 14 Appendix B: Data Constants, Gateway application software is updated | PR100385 | Afreen P | 10-10-2025 |

# 2 Objective

The document H398-003-011-GWY defines the Software Low Level Requirements for the Computer Software Configuration Item (CSCI) Gateway Module Application Software of Engine Data Acquisition Unit (EDAU) for Airbus Helicopters model AS532U2.

# 3 Scope

The document H398-003-011-GWY specifies the Software Low Level Requirements of the Gateway Module. This document is written according to H398-001-007 and satisfies DO-178B section 11.7 for Software Design Data.

# 4 References

The Table 2 provides the list of documents referred while creating the Software Low Level Requirements of Gateway Module (H398-003-011-GWY).

Table : Reference document

|  |  |  |
| --- | --- | --- |
| **Source** | **Document No.** | **Title** |
| RTCA | DO-178B | Software Considerations in Airborne Systems and Equipment Certification |
| Howell | SYS2160SRS | Airbus Helicopters AS532U2 Engine Instrument System Requirements Specification |
| 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** | H398-001-007 | Software Design Standards |
| ALTEN Global Technologies **Private Limited** | H398-002-001-GWY | Software Requirements Specification for Gateway Module of Airbus Engine Data Acquisition Unit (EDAU) and Configuration Management Unit plus NVM (CMU+) |
| ALTEN Global Technologies **Private Limited** | H398-003-001-GWY | Software Architectural Design for Gateway of Airbus Engine Data Acquisition Unit (EDAU) and Configuration Management Unit plus NVM (CMU+) |
| STMicroelectronics | RM0090  Reference manual | Reference manual for  STM32F405xx/07xx, STM32F415xx/17xx, STM32F42xxx and STM32F43xxx advanced ARM®-based 32-bit MCUs |

# 5 Acronyms and Definitions

The Table 3 provides the list of abbreviation used for the project.

Table : List of Abbreviation

|  |  |
| --- | --- |
| **Acronym** | **Definition** |
| A825 | ARINC 825 |
| ADC | Analog to Digital Converter |
| AHB | Advanced High-Performance Bus |
| ALTEGT | Alternate EGT |
| ALTEGTL | Alternate EGT for Left Engine |
| ALTEGTR | Alternate EGT for Right Engine |
| ALTQFP | Alternate Torque |
| AOG | Aircraft On-Ground |
| APB | Advanced Peripheral Bus |
| ARINC | Aeronautical Radio Incorporated |
| ASCII | American Standard Code for Information Interchange |
| BMS | Balance Monitoring System |
| CAN | Controller Area Network |
| CBIT | Continuous Built In Test |
| CLP | Collective Position |
| CMSIS | Cortex microcontroller software interface standard |
| CMU | Configuration Management Unit |
| CPU | Central Processing Unit |
| CRC | Cyclic Redundancy Check |
| CSC | Computer Software Component |
| CSCI | Computer Software Configuration Item |
| CSU | Computer Software Unit |
| CTCNTC | Compressor Turbine Counts Course |
| DAC | Digital to Analog Convertor |
| DAU | Data Acquisition Unit |
| DIU | Discrete Unit |
| DLC | Data Length Code |
| DLE | Data Link Escape |
| DLU | Data Logger Unit |
| DMA | Direct Memory Access |
| DOC | Document Object Code |
| DSCWD1 | Status Discrete Word 1 |
| DSCWD2 | Status Discrete Word 2 |
| DTOA | Digital TO Analog |
| ECB | Event Control Block |
| ECDU | UASC EFIS Control Display Unit |
| EEC | Electronic Engine Controller |
| EFI | Electronic Flight Instrument |
| EFIS | Electronic Flight Instrument System |
| EGT | Exhaust Gas Temperature |
| EICAS | Engine Indicating and Crew Alerting System |
| EOP | Engine Oil Pressure |
| EOT | Engine Oil Temperature |
| FID | Functional Code Identifier |
| FPCCR | Floating-Point Context Control Register |
| FPSCR | Floating-Point Status Control Register |
| FSMC | Flexible Static Memory Controller |
| GPS | Global Positioning System |
| GPU | Ground Power Unit |
| H/W | Hardware |
| HL | High Level |
| HLR | High Level Requirements |
| HW | Hardware |
| ICD | Interface Control Document |
| ID | Identifier |
| IMCNTC | Impeller Counts Course |
| IPS | Inlet Particle Separator |
| IS | Interface Specification |
| ISR | Interrupt Service Routine |
| LED | Light Emitting Diode |
| LL | Low Level |
| LLR | Low Level Requirements |
| MFD | Multi-Function Display |
| MGTIND (EGT) | Indicated Measured Temperature |
| NACA | National Advisory Committee for Aeronautics |
| NCFUR1 | Non Critical Failure Word 1 |
| NCFUR2 | Non Critical Failure Word 2 |
| NCFUR3 | Non Critical Failure Word 3 |
| NG | Gas Generator Speed |
| NOC | Normal Operation Channel |
| NSC | Node Service Channel |
| NVM | Non-Volatile Memory |
| NVRAM | Non-Volatile Random Access Memory |
| OEI | One Engine Inoperative |
| P2P | Peer To Peer |
| PBIT | Power-On Built In Test |
| PDLPOS | Pedal Position |
| PFD | Primary Flight Display |
| PLA | Power Lever Angle |
| PTCNTC | Power Turbine Counts Course |
| PSR | Program Status Control Register |
| QA | Quality Assurance |
| QIND (QI) | Indicated Torque |
| ROM | Read Only Memory |
| RAM | Random Access Memory |
| RCI | Redundancy Channel Identifier |
| RMS | Root Mean Square |
| RPM | Revolutions Per Minute |
| RTD | Resistance Temperature Detectors |
| RTHETA | Thermal Resistance |
| RTOS | Real Time Operating System |
| RX | Receive |
| S/W | Software |
| SAD | Software Architectural Document |
| SCMGTT | Engine-specific EGT trim value in degrees C |
| SCMP | Software Configuration Management Plan |
| SCN | Top Level Software Configuration Number |
| SCQBI | Torque Bias Trim Value |
| SCQGI | Torque Gain Trim Value |
| SDD | Software Design Document |
| SFC | Service Function Code |
| SID | Server IDentifier |
| SMT | Service Message Type |
| SRAM | Static Random Access Memory |
| SRS | Software Requirement Specifications |
| SSM | Sign/Status Matrix |
| SVN | SubVersioN |
| T1 | Inlet Temperature |
| TX | Transmit |
| TACH | Tachometer |
| TCB | Task Control Block |
| UART | Universal Asynchronous Receiver Transmitter |
| UAS | Universal Avionics Systems |
| VDC | Voltage Direct Current |
| VSCS | Vertical Stabilizer Control System |
| WFPRED | Predicted Fuel Flow |

# 6 Document Control

According to the H398-001-004 (SCMP), this document shall be maintained in AlTEN Software & Software Configuration management System (CMS) after the QA review, under the following path [<http://192.168.1.229/svn/A21HOWBLDAU/ACCORD/SW/Trunk/Documents/Design/LLD>] under change control.

# 7 Responsibilities

1. Software Development Team members specify the Software Low Level Requirements.

2. Reviewers review the Software Low Level Requirements based on the DO-178B check points provided as attributes in ReMa.

3. Once all the comments from the Reviewer are 'Looked Into' and the Software Low Level Requirement status is ‘Closed’ in ReMa, QA changes the QA Review Status Requirement to "QA Approved" and closed.

4. Engineering Manager is responsible for base lining the document to SVN through ReMa, as well as the Exported document, into the path

http://192.168.1.230/svn/A21HOWBLDAU/V2\_H398/ACCORD/SW/Branches/SOI2/Documents/Design/LLD

# 8 Distributions

The Software Low Level Requirements Document of Gateway Module (H398-003-011-GWY) is distributed to Howell Instruments, Inc. through a secured File Transform Protocol server.

# 9 Traceability

The Software Low Level Requirements Document for Gateway Module of Engine Data Acquisition Unit (EDAU)for Airbus Helicopter model AS532U2(H398-003-011-GWY) is derived from the Software High level Requirements present in the Software High Level Requirements Specification Document for Gateway Module of Engine Data Acquisition Unit (EDAU) for Airbus Helicopter model AS532U2(H398-002-001-GWY) and Software Architectural Design for Gateway Module of Engine Data Acquisition Unit (EDAU) for Airbus Helicopter model AS532U2(H398-003-001-GWY). Traceability of H398-003-011-GWY to H398-002-001-GWY and H398-003-001-GWY will be provided in Bi-Directional Traceability Matrix from SLL to SRS/SAD(H398-003-012-GWY).

# 10 Software Low Level Requirements- Gateway Module Application Software

This section specifies the Software Low Level Requirements for Gateway Module Application Software.

Refer Refer section 11 Appendix A: Data Structures of H398-003-001-GWY for Data Structures and Enumerations.

## 10.1 daugwya429

The daugwya429 CSC defines the a429 driver and its labels.

### 10.1.1 A429StatusMiscompare

Low Level Design Details about CSU A429StatusMiscompare will follow in the sub sections.

#### 10.1.1.1 Brief Description

The A429StatusMiscompare function processes the A429 Status of Miss Comparions.

#### 10.1.1.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.1.1.3 List of global variables accessed and modified

Accessed : Mis\_cmp\_fail

Modified : None

#### 10.1.1.4 Parameter list (Input/Output)

Inputs: T\_UINT8 a429\_type - A429 type(BNR,DIS)

T\_DATA\_STAT status\_mis - status miscompare flag

T\_UINT8 label\_i - Transmit labels 270, 350, 371, 372 and 373

Outputs: T\_DATA\_STAT status\_mis - status miscompare flag

#### 10.1.1.5 Return Value

T\_DATA\_STAT – returns the status of variable

#### 10.1.1.6 Other CSUs called by this CSU

None

#### 10.1.1.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to A429StatusMiscompare

##### 10.1.1.7.1 daugwya429-A429StatusMiscompare-LLR-006

Requirement ID: H398-LLD-GWY-FNC-25

The function shall return status\_mis.

### 10.1.2 A429OutTask

Low Level Design Details about CSU A429OutTask will follow in the sub sections.

#### 10.1.2.1 Brief Description

The A429OutTask function processes the ARINC 429 output Task.

#### 10.1.2.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.1.2.3 List of global variables accessed and modified

Accessed : App\_emode, Inita429, Cmdownload\_lp\_cmd\_data, Tx\_a429\_bnr,App\_strap

Modified : None

#### 10.1.2.4 Parameter list (Input/Output)

Inputs: void \*pdata– Not Used

Outputs: void \*pdata– Not Used

#### 10.1.2.5 Return Value

None

#### 10.1.2.6 Other CSUs called by this CSU

OsSemPend

A429StatusMiscompare

TransmitArinc429

CalcArincWordChecksum

#### 10.1.2.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to A429OutTask

##### 10.1.2.7.1 daugwya429-A429OutTask-LLR-001

Requirement ID: H398-LLD-GWY-FNC-34

The function shall set pdata to M\_HW\_NULL.

##### 10.1.2.7.2 daugwya429-A429OutTask-LLR-002

Requirement ID: H398-LLD-GWY-FNC-35

The function shall loop forever and call OsSemPend with parameters (Semaphore a429 output task, M\_DEC\_ZERO and reference of error status).

##### 10.1.2.7.3 daugwya429-A429OutTask-LLR-003

Requirement ID: H398-LLD-GWY-FNC-36

The function shall loop forever and do nothing when

1. ((App\_emode not equal to APP\_NORMAL) OR (NOT of Inita429) OR (NOT of App\_strap)) is TRUE

##### 10.1.2.7.4 daugwya429-A429OutTask-LLR-004

Requirement ID: H398-LLD-GWY-FNC-37

The function shall loop forever and then loop from counter is M\_ZERO until numOf429out\_bnr of Cmdownload\_lp\_cmd\_data and set binary output counter to M\_ONE when

1. ((App\_emode not equal to APP\_NORMAL) OR NOT of Inita429 OR NOT of App\_strap) returns FALSE
2. binary output counter is greater than or equal to (update\_rate of A429out\_bnr of Cmdownload\_lp\_cmd\_data is multiplied with M\_A429OUT\_MSECOND) by M\_TEN

##### 10.1.2.7.5 daugwya429-A429OutTask-LLR-005

Requirement ID: H398-LLD-GWY-FNC-38

The function shall loop forever, loop from counter is M\_ZERO until numOf429out\_bnr of Cmdownload\_lp\_cmd\_data minus 1 and then loop from channel loop is M\_ZERO to M\_A429\_MAX\_TX\_CHAN minus 1 and do nothing

when

a) ((App\_emode not equal to APP\_NORMAL) OR NOT of Inita429 OR NOT of App\_strap) returns FALSE

b) binary output counter is greater than or equal to (update\_rate of A429out\_bnr of Cmdownload\_lp\_cmd\_data is multiplied with M\_A429OUT\_MSECOND) by M\_TEN

c) (((channel of A429out\_bnr of Cmdownload\_lp\_cmd\_data right shifted to channel loop) Bitwise AND M\_LSB\_FIRST\_BIT\_MASK) not equal to M\_ZERO)

d) type of A429out\_bnr of Cmdownload\_lp\_cmd\_data is equal to A429D\_BNR

e) (M\_ZERO is equal to resolution of A429out\_bnr with index loop counter of Cmdownload\_lp\_cmd\_data by calling the macro M\_FP\_COMPARE)

##### 10.1.2.7.6 daugwya429-A429OutTask-LLR-006

Requirement ID: H398-LLD-GWY-FNC-39

The function shall loop forever,loop from counter is M\_ZERO until numOf429out\_bnr of Cmdownload\_lp\_cmd\_data and then loop from channel loop is M\_ZERO to M\_A429\_MAX\_TX\_CHAN and do the following

1. Call A429StatusMiscompare with parameters (A429D\_BNR,counter, status of Tx\_a429\_bnr with index as counter) and set the return value to status binary
2. Call TransmitArinc429 with parameters channel loop, label of A429out\_bnr (with index as counter) of Cmdownload\_lp\_cmd\_data, (division of (data of Tx\_a429\_bnr with index as counter is subtracted with offsetValue of A429out\_bnr(with index as counter) of Cmdownload\_lp\_cmd\_data)

by resolution of A429out\_bnr (with index as counter) of Cmdownload\_lp\_cmd\_data), sig\_bits of A429out\_bnr(with index as counter) of Cmdownload\_lp\_cmd\_data, M\_DEC\_ZERO, id of Tx\_a429\_bnr with index as counter, signBit of A429out\_bnr(with index as counter) of Cmdownload\_lp\_cmd\_data, status binary to variable ARINC word out.

when

1. ((App\_emode not equal to APP\_NORMAL) OR NOT of Inita429 OR NOT of App\_strap) returns FALSE
2. binary output counter is greater than or equal to (update\_rate of A429out\_bnr with index u16i of Cmdownload\_lp\_cmd\_data is multiplied with M\_A429OUT\_MSECOND) by M\_TEN
3. (((channel of A429out\_bnr of Cmdownload\_lp\_cmd\_data right shifted to channel loop) Bitwise AND M\_LSB\_FIRST\_BIT\_MASK) not equal to M\_ZERO)
4. type of A429out\_bnr with index u16i of Cmdownload\_lp\_cmd\_data is equal to A429D\_BNR
5. resolution of A429out\_bnr with index u16i of Cmdownload\_lp\_cmd\_data is not equal to M\_ZERO

##### 10.1.2.7.7 daugwya429-A429OutTask-LLR-007

Requirement ID: H398-LLD-GWY-FNC-40

The function shall loop forever,loop from counter is M\_ZERO until numOf429out\_bnr of Cmdownload\_lp\_cmd\_data and then loop from channel loop is M\_ZERO to M\_A429\_MAX\_TX\_CHAN and do nothing

when

1. ((App\_emode not equal to APP\_NORMAL) OR NOT of Inita429 OR NOT of App\_strap) returns FALSE
2. binary output counter is greater than or equal to (update\_rate of A429out\_bnr of Cmdownload\_lp\_cmd\_data is multiplied with M\_A429OUT\_MSECOND) by M\_TEN
3. (((channel of A429out\_bnr of Cmdownload\_lp\_cmd\_data right shifted to channel loop) Bitwise AND M\_LSB\_FIRST\_BIT\_MASK) not equal to M\_ZERO)
4. type of A429out\_bnr of Cmdownload\_lp\_cmd\_data is not equal to A429D\_BNR

##### 10.1.2.7.8 daugwya429-A429OutTask-LLR-008

Requirement ID: H398-LLD-GWY-FNC-41

The function shall loop forever, loop from counter is M\_ZERO until numOf429out\_bnr of Cmdownload\_lp\_cmd\_data and increment the binary output counter (with counter as index) by one

when

1. ((App\_emode not equal to APP\_NORMAL) OR NOT of Inita429 OR NOT of App\_strap) returns FALSE
2. binary output counter is less than to (update\_rate of A429out\_bnr of Cmdownload\_lp\_cmd\_data is multiplied with M\_A429OUT\_MSECOND) by M\_TEN
3. counter is less than or equal to binary loop

##### 10.1.2.7.9 daugwya429-A429OutTask-LLR-009

Requirement ID: H398-LLD-GWY-FNC-6151

The function shall increment the binary loop by one when

a) ((App\_emode not equal to APP\_NORMAL) OR NOT of Inita429 OR NOT of App\_strap) returns FALSE

b) binary loop is less than numOf429out\_bnr of Cmdownload\_lp\_cmd\_data.

##### 10.1.2.7.10 daugwya429-A429OutTask-LLR-010

Requirement ID: H398-LLD-GWY-FNC-7452

The function shall loop forever,loop from counter is M\_ZERO until numOf429out\_bnr of Cmdownload\_lp\_cmd\_data and then loop from channel loop is M\_ZERO to M\_A429\_MAX\_TX\_CHAN and do the following

1. Call CalcArincWordChecksum with parameters variable ARINC word out and set the return value to a429 checksum 074

2. Set the checksum data package to a429 checksum 074.

3.Update the checksum data package to bitwise OR of checksum data package and (label of A429out\_bnr of index size of count of Cmdownload\_lp\_cmd\_data shifted to left by M\_SHIFT\_BY\_8) .

4. Update the checksum data package to bitwise OR of checksum data package and (id of Tx\_a429\_bnr of index size of count shifted left by M\_SHIFT\_BY\_16).

5. Call TransmitArinc429 function wiith parameters (channel loop, M\_LABEL\_074, checksum data package,  M\_EIGHTEEN, M\_ZERO, M\_HW\_GET\_RCI plus M\_ONE, A429\_DATA, OK).

When

1. label of A429out\_bnr of index size of count of Cmdownload\_lp\_cmd\_data is not equal to M\_LABEL\_031 AND label of A429out\_bnr of index size of count of Cmdownload\_lp\_cmd\_data is not equal to M\_LABEL\_034 AND label of A429out\_bnr of index size of count of Cmdownload\_lp\_cmd\_data is not equal to M\_LABEL\_062 AND label of A429out\_bnr of index size of count of Cmdownload\_lp\_cmd\_data is not equal to M\_LABEL\_063 AND label of A429out\_bnr of index size of count of Cmdownload\_lp\_cmd\_data is not equal to M\_LABEL\_064 AND label of A429out\_bnr of index size of count of Cmdownload\_lp\_cmd\_data is not equal to M\_LABEL\_065
2. ((App\_emode not equal to APP\_NORMAL) OR NOT of Inita429 OR NOT of App\_strap) returns FALSE
3. binary output counter is greater than or equal to (update\_rate  of A429out\_bnr with index count of Cmdownload\_lp\_cmd\_data is multiplied with M\_A429OUT\_MSECOND) by M\_TEN
4. (((channel of A429out\_bnr of Cmdownload\_lp\_cmd\_data right shifted to channel loop) Bitwise AND M\_LSB\_FIRST\_BIT\_MASK) not equal to M\_ZERO)
5. type of A429out\_bnr of Cmdownload\_lp\_cmd\_data is equal to A429D\_BNR
6. resolution of A429out\_bnr with index count of Cmdownload\_lp\_cmd\_data is not equal to M\_ZERO

### 10.1.3 LabelPriority

Low Level Design Details about CSU LabelPriority will follow in the sub sections.

#### 10.1.3.1 Brief Description

The LabelPriority function processes label priority data.

#### 10.1.3.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.1.3.3 List of global variables accessed and modified

Accessed : Rx\_a429, Cmdownload\_lp\_cmd\_data

Modified : Priority

#### 10.1.3.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.1.3.5 Return Value

None

#### 10.1.3.6 Other CSUs called by this CSU

GpioReadInputDataBit

#### 10.1.3.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to LabelPriority.

##### 10.1.3.7.1 daugwya429-LabelPriority-LLR-001

Requirement ID: H398-LLD-GWY-FNC-50

The function shall loop from label loop is zero to numOf429in of Cmdownload\_lp\_cmd\_data and set the Priority for each iteration with DU\_1 when following conditions are satisfied:

1. Return value of function call GpioReadInputDataBit with parameters (M\_GPIOC, M\_GPIOC\_SYS\_SEL) through M\_HW\_GET\_RCI is equal to M\_ZERO
2. status of Rx\_a429 with indices (DU\_1 and label loop) is equal to OK

##### 10.1.3.7.2 daugwya429-LabelPriority-LLR-002

Requirement ID: H398-LLD-GWY-FNC-51

The function shall loop from label loop is zero to numOf429in of Cmdownload\_lp\_cmd\_data and set the Priority for each iteration with DU\_2 when following conditions are satisfied:

1. Return value of function call GpioReadInputDataBit with parameters (M\_GPIOC, M\_GPIOC\_SYS\_SEL) through M\_HW\_GET\_RCI is equal to M\_ZERO
2. status of Rx\_a429 with indices (DU\_1 and label loop) is not equal to OK
3. status of Rx\_a429 with indices (DU\_2 and label loop) is equal to OK

##### 10.1.3.7.3 daugwya429-LabelPriority-LLR-003

Requirement ID: H398-LLD-GWY-FNC-52

The function shall loop from label loop is zero to numOf429in of Cmdownload\_lp\_cmd\_data and set the Priority for each iteration with DU\_3 when following conditions are satisfied:

1. Return value of function call GpioReadInputDataBit with parameters (M\_GPIOC, M\_GPIOC\_SYS\_SEL) through M\_HW\_GET\_RCI is equal to M\_ZERO
2. status of Rx\_a429 with indices (DU\_1 and label loop) is not equal to OK
3. status of Rx\_a429 with indices (DU\_2 and label loop) is not equal to OK
4. status of Rx\_a429 with indices (DU\_3 and label loop) is equal to OK

##### 10.1.3.7.4 daugwya429-LabelPriority-LLR-004

Requirement ID: H398-LLD-GWY-FNC-53

The function shall loop from label loop is zero to numOf429in of Cmdownload\_lp\_cmd\_data and set the Priority for each iteration with DU\_4 when following conditions are satisfied:

1. Return value of function call GpioReadInputDataBit with parameters (M\_GPIOC, M\_GPIOC\_SYS\_SEL) through M\_HW\_GET\_RCI is equal to M\_ZERO
2. status of Rx\_a429 with indices (DU\_1 and label loop) is not equal to OK
3. status of Rx\_a429 with indices (DU\_2 and label loop) is not equal to OK
4. status of Rx\_a429 with indices (DU\_3 and label loop) is not equal to OK
5. status of Rx\_a429 with indices (DU\_4 and label loop) is equal to OK

##### 10.1.3.7.5 daugwya429-LabelPriority-LLR-005

Requirement ID: H398-LLD-GWY-FNC-54

The function shall loop from label loop is zero to numOf429in of Cmdownload\_lp\_cmd\_data and set the Priority for each iteration with DU\_1 when following conditions are satisfied:

1. Return value of function call GpioReadInputDataBit with parameters (M\_GPIOC, M\_GPIOC\_SYS\_SEL) through M\_HW\_GET\_RCI is equal to M\_ZERO
2. status of Rx\_a429 with indices (DU\_1 and label loop) is not equal to OK
3. status of Rx\_a429 with indices (DU\_2 and label loop) is not equal to OK
4. status of Rx\_a429 with indices (DU\_3 and label loop) is not equal to OK
5. status of Rx\_a429 with indices (DU\_4 and label loop) is not equal to OK

##### 10.1.3.7.6 daugwya429-LabelPriority-LLR-006

Requirement ID: H398-LLD-GWY-FNC-55

The function shall loop from label loop is zero to numOf429in of Cmdownload\_lp\_cmd\_data and set the Priority for each iteration with DU\_1 when following conditions are satisfied:

1. Return value of function call GpioReadInputDataBit with parameters (M\_GPIOC, M\_GPIOC\_SYS\_SEL) through M\_HW\_GET\_RCI is not equal to M\_ZERO
2. status of Rx\_a429 with indices (DU\_1 and label loop) is equal to OK

##### 10.1.3.7.7 daugwya429-LabelPriority-LLR-007

Requirement ID: H398-LLD-GWY-FNC-56

The function shall loop from label loop is zero to numOf429in of Cmdownload\_lp\_cmd\_data and set the Priority for each iteration with DU\_2 when following conditions are satisfied:

1. Return value of function call GpioReadInputDataBit with parameters (M\_GPIOC, M\_GPIOC\_SYS\_SEL) through M\_HW\_GET\_RCI is not equal to M\_ZERO
2. status of Rx\_a429 with indices (DU\_1 and label loop) is not equal to OK
3. status of Rx\_a429 with indices (DU\_2 and label loop) is equal to OK

##### 10.1.3.7.8 daugwya429-LabelPriority-LLR-008

Requirement ID: H398-LLD-GWY-FNC-57

The function shall loop from label loop is zero to numOf429in of Cmdownload\_lp\_cmd\_data and set the Priority for each iteration with DU\_3 when following conditions are satisfied:

1. Return value of function call GpioReadInputDataBit with parameters (M\_GPIOC, M\_GPIOC\_SYS\_SEL) through M\_HW\_GET\_RCI is not equal to M\_ZERO
2. status of Rx\_a429 with indices (DU\_1 and label loop) is not equal to OK
3. status of Rx\_a429 with indices (DU\_2 and label loop) is not equal to OK
4. status of Rx\_a429 with indices (DU\_3 and label loop) is equal to OK

##### 10.1.3.7.9 daugwya429-LabelPriority-LLR-009

Requirement ID: H398-LLD-GWY-FNC-58

The function shall loop from label loop is zero to numOf429in of Cmdownload\_lp\_cmd\_data and set the Priority for label loop with DU\_4 when following conditions are satisfied:

1. Return value of function call GpioReadInputDataBit with parameters (M\_GPIOC, M\_GPIOC\_SYS\_SEL) through M\_HW\_GET\_RCI is not equal to M\_ZERO
2. status of Rx\_a429 with indices (DU\_1 and label loop) is not equal to OK
3. status of Rx\_a429 with indices (DU\_2 and label loop) is not equal to OK
4. status of Rx\_a429 with indices (DU\_3 and label loop) is not equal to OK
5. status of Rx\_a429 with indices (DU\_4 and label loop) is equal to OK

##### 10.1.3.7.10 daugwya429-LabelPriority-LLR-010

Requirement ID: H398-LLD-GWY-FNC-59

The function shall loop from label loop is zero to numOf429in of Cmdownload\_lp\_cmd\_data and set the Priority for each iteration with DU\_1 when following conditions are satisfied:

1. Return value of function call GpioReadInputDataBit with parameters (M\_GPIOC, M\_GPIOC\_SYS\_SEL) through M\_HW\_GET\_RCI is not equal to M\_ZERO
2. status of Rx\_a429 with indices (DU\_1 and label loop) is not equal to OK
3. status of Rx\_a429 with indices (DU\_2 and label loop) is not equal to OK
4. status of Rx\_a429 with indices (DU\_3 and label loop) is not equal to OK
5. status of Rx\_a429 with indices (DU\_4 and label loop) is not equal to OK

### 10.1.4 ProcessLabel

Low Level Design Details about CSU ProcessLabel will follow in the sub sections.

#### 10.1.4.1 Brief Description

The ProcessLabel function processes ARINC 429 data from display unit.

#### 10.1.4.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.1.4.3 List of global variables accessed and modified

Accessed : A429\_aas\_chans

Modified : Rx\_a429

#### 10.1.4.4 Parameter list (Input/Output)

Inputs : T\_A429\_CHAN u8\_chan - channel number

T\_UINT8 u8\_label\_index - label index

T\_SINT32 i32\_arinc\_word-arinc data

Outputs : None

#### 10.1.4.5 Return Value

None

#### 10.1.4.6 Other CSUs called by this CSU

InputLabelTx

#### 10.1.4.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to ProcessLabel.

##### 10.1.4.7.1 daugwya429-ProcessLabel-LLR-001

Requirement ID: H398-LLD-GWY-FNC-68

The function shall set ssm temporary to (Bitwise AND of (bit shift of i32\_arinc\_word to right by M\_SHIFT\_BY\_9) and M\_THREE)

##### 10.1.4.7.2 daugwya429-ProcessLabel-LLR-002

Requirement ID: H398-LLD-GWY-FNC-69

The function shall set the following when status of A429\_aas\_chans with indices (u8\_chan , u8\_label\_index) is equal to OK OR status of A429\_aas\_chans with indices (u8\_chan , u8\_label\_index) is equal to CHAN\_ERR:

1. Set id of Rx\_a429 for indices (u8\_chan, u8\_label\_index) with (Bitwise AND of (bit shift of i32\_arinc\_word to right by M\_SHIFT\_BY\_11) and M\_SDI\_BITS)
2. Set data of Rx\_a429 for indices (u8\_chan, u8\_label\_index) with product of (i32\_reading of A429\_aas\_chans for the indices u8\_chan ,u8\_label\_index) and (i32\_scalefactor of A429\_aas\_chans for the indices u8\_chan ,u8\_label\_index)
3. Set status of Rx\_a429 for indices (u8\_chan, u8\_label\_index) with status of A429\_aas\_chans for the indices u8\_chan ,u8\_label\_index

##### 10.1.4.7.3 daugwya429-ProcessLabel-LLR-003

Requirement ID: H398-LLD-GWY-FNC-70

The function shall set status of Rx\_a429 for indices u8\_chan, u8\_label\_index with OK when

1. status of A429\_aas\_chans with indices (u8\_chan , u8\_label\_index) is equal to OK OR status of A429\_aas\_chans with indices (u8\_chan , u8\_label\_index) is equal to CHAN\_ERR
2. ((ssm temporary is equal to M\_THREE) AND (type of A429\_aas\_chans for indices u8\_chan,u8\_label\_index is equal to BNR)) returns TRUE

##### 10.1.4.7.4 daugwya429-ProcessLabel-LLR-004

Requirement ID: H398-LLD-GWY-FNC-71

The function shall set status of Rx\_a429 for indices u8\_chan, u8\_label\_index with OK when

1. status of A429\_aas\_chans with indices (u8\_chan , u8\_label\_index) is equal to OK OR status of A429\_aas\_chans with indices (u8\_chan , u8\_label\_index) is equal to CHAN\_ERR
2. ((ssm temporary is equal to M\_THREE) AND (type of A429\_aas\_chans for indices u8\_chan,u8\_label\_index is equal to BNR)) returns FALSE
3. ((ssm temporary is equal to M\_ZERO) AND (type of A429\_aas\_chans for indices u8\_chan,u8\_label\_index is not equal to BNR)) returns TRUE

##### 10.1.4.7.5 daugwya429-ProcessLabel-LLR-005

Requirement ID: H398-LLD-GWY-FNC-72

The function shall set status of Rx\_a429 for indices u8\_chan, u8\_label\_index with FUNC\_TEST when

1. status of A429\_aas\_chans with indices (u8\_chan , u8\_label\_index) is equal to OK OR status of A429\_aas\_chans with indices (u8\_chan , u8\_label\_index) is equal to CHAN\_ERR
2. ((ssm temporary is equal to M\_THREE) AND (type of A429\_aas\_chans for indices u8\_chan,u8\_label\_index is equal to BNR)) returns FALSE
3. ((ssm temporary is equal to M\_ZERO) AND (type of A429\_aas\_chans for indices u8\_chan,u8\_label\_index is not equal to BNR)) returns FALSE
4. ssm temporary is equal to M\_TWO

##### 10.1.4.7.6 daugwya429-ProcessLabel-LLR-006

Requirement ID: H398-LLD-GWY-FNC-73

The function shall set status of Rx\_a429 for indices u8\_chan, u8\_label\_index with CHAN\_ERR when

1. status of A429\_aas\_chans with indices (u8\_chan , u8\_label\_index) is equal to OK OR status of A429\_aas\_chans with indices (u8\_chan , u8\_label\_index) is equal to CHAN\_ERR
2. ((ssm temporary is equal to three) AND (type of A429\_aas\_chans for indices u8\_chan,u8\_label\_index is equal to BNR)) returns FALSE
3. ((ssm temporary is equal to zero) AND (type of A429\_aas\_chans for indices u8\_chan,u8\_label\_index is not equal to BNR)) returns FALSE
4. ssm temporary is not equal to two

##### 10.1.4.7.7 daugwya429-ProcessLabel-LLR-007

Requirement ID: H398-LLD-GWY-FNC-74

The function shall do the following when status of A429\_aas\_chans with indices (u8\_chan, u8\_label\_index) is equal to OK OR status of A429\_aas\_chans with indices (u8\_chan, u8\_label\_index) is equal to CHAN\_ERR:

1. Call InputLabelTx with parameters u8\_chan, u8\_label\_index and i32\_arinc\_word

##### 10.1.4.7.8 daugwya429-ProcessLabel-LLR-008

Requirement ID: H398-LLD-GWY-FNC-75

The function shall do the following when (status of A429\_aas\_chans with indices (u8\_chan, u8\_label\_index) is equal to OK OR status of A429\_aas\_chans with indices (u8\_chan, u8\_label\_index) is equal to CHAN\_ERR) returns FALSE and status of A429\_aas\_chans with indices (u8\_chan, u8\_label\_index) is MISSING:

1. set data of Rx\_a429 for indices u8\_chan, u8\_label\_index to M\_ZERO
2. set status of Rx\_a429 for indices u8\_chan, u8\_label\_index to MISSING

##### 10.1.4.7.9 daugwya429-ProcessLabel-LLR-009

Requirement ID: H398-LLD-GWY-FNC-76

The function shall do nothing when

1. (status of A429\_aas\_chans with indices (u8\_chan, u8\_label\_index) is equal to OK OR status of A429\_aas\_chans with indices (u8\_chan, u8\_label\_index) is equal to CHAN\_ERR) returns FALSE
2. status of A429\_aas\_chans with indices (u8\_chan, u8\_label\_index) is not equal to MISSING

### 10.1.5 ProcessArincWord

Low Level Design Details about CSU ProcessArincWord will follow in the sub sections.

#### 10.1.5.1 Brief Description

The ProcessArincWord function extracts the ARINC word except significant bit.

#### 10.1.5.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.1.5.3 List of global variables accessed and modified

Accessed : A429\_mode, Appdata\_appdata,A429\_aas\_chans

Modified : A429\_aas\_chans

#### 10.1.5.4 Parameter list (Input/Output)

Inputs : T\_UINT8 u8\_rx\_chan - Channel number,

T\_SINT32 i32\_arinc\_word -32-bit ARINC word

Outputs : None

#### 10.1.5.5 Return Value

None

#### 10.1.5.6 Other CSUs called by this CSU

ProcessLabel

#### 10.1.5.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to ProcessArincWord.

##### 10.1.5.7.1 daugwya429-ProcessArincWord-LLR-001

Requirement ID: H398-LLD-GWY-FNC-85

The function shall set test data with (bit shift of i32\_arinc\_word to right by (subtraction of M\_THIRTYONE with Test sign bits)) when

1. func\_state of a429chan (with index u8\_rx\_chan) of Appdata\_appdata is equal to ENABLED
2. A429\_mode is equal to A429\_SELFTEST

##### 10.1.5.7.2 daugwya429-ProcessArincWord-LLR-026

Requirement ID: H398-LLD-GWY-FNC-6183

The function shall call RterrError function with parameters A429\_LOOPBACK\_ERR,u8\_rx\_chan, reference of function pointer RterrForever and Set A429\_loop\_back\_fail\_flag to TRUE when following conditions are satisfied otherwise set Arinc\_loop\_back\_status with index u8\_rx\_chan as TRUE.

a) func\_state of a429chan (with index u8\_rx\_chan) of Appdata\_appdata is equal to ENABLED

b) A429\_mode is equal to A429\_SELFTEST

c) (test data is not equal to M\_ARINC\_TEST\_DATA) AND (test data is not equal to M\_ARINC\_TEST\_DATA\_INVRT) OR ((i32\_arinc\_word Bitwise AND with M\_EXTRCT\_32BIT\_LSB) is not equal to M\_ARINC\_LABEL\_INVERT\_0270) AND ((i32\_arinc\_word Bitwise AND with M\_EXTRCT\_32BIT\_LSB) is not equal to M\_ARINC\_LABEL\_0270).

##### 10.1.5.7.3 daugwya429-ProcessArincWord-LLR-002

Requirement ID: H398-LLD-GWY-FNC-86

The function shall loop from loop index is M\_ZERO to M\_A429\_MAX\_LABEL and do the following when func\_state of a429chan (with index u8\_rx\_chan) of Appdata\_appdata is equal to ENABLED and A429\_mode is not equal to A429\_SELFTEST

1. Set label for all the indices of (u8\_rx\_chan,loop index) with (i32\_arinc\_word Bitwise AND M\_EXTRCT\_32BIT\_LSB)
2. Set sdi for all the indices of (u8\_rx\_chan,loop index) with ((bit shift of i32\_arinc\_word to right by M\_SHIFT\_BY\_11) Bitwise AND M\_SDI\_BITS)
3. Set parity for all the indices of (u8\_rx\_chan,loop index) with (bit shift of (i32\_arinc\_word Bitwise AND M\_EXTRCT\_PARITY) to right by M\_SHIFT\_BY\_8)

##### 10.1.5.7.4 daugwya429-ProcessArincWord-LLR-004

Requirement ID: H398-LLD-GWY-FNC-88

The function shall loop from loop index is M\_ZERO to M\_A429\_MAX\_LABEL minus 1 and do the following

1. Set no\_lbl\_cnt of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) with the product of u16\_update\_rate of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) and M\_SIX and it is divided by M\_TEN .
2. Set lbl\_miss of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) with TRUE
3. Set status of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) with MISSING
4. Call ProcessLabel with parameters (u8\_rx\_chan, loop index, i32\_arinc\_word)

when

1. func\_state of a429chan (with index u8\_rx\_chan) of Appdata\_appdata is equal to ENABLED
2. A429\_mode is equal to A429\_SELFTEST
3. i32\_arinc\_word is not equal to M\_ZERO
4. no\_lbl\_cnt of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) is greater than product of u16\_update\_rate of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) and M\_SIX and it is divided by M\_TEN .

##### 10.1.5.7.5 daugwya429-ProcessArincWord-LLR-005

Requirement ID: H398-LLD-GWY-FNC-89

The function shall loop from loop index is M\_ZERO to M\_A429\_MAX\_LABEL and do nothing

when

1. func\_state of a429chan (with index u8\_rx\_chan) of Appdata\_appdata is equal to ENABLED
2. A429\_mode is not equal to A429\_SELFTEST
3. i32\_arinc\_word is equal to M\_ZERO
4. no\_lbl\_cnt of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) is not equal to product of u16\_update\_rate of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) and M\_SIX

##### 10.1.5.7.6 daugwya429-ProcessArincWord-LLR-006

Requirement ID: H398-LLD-GWY-FNC-90

The function shall loop from loop index is M\_ZERO to M\_A429\_MAX\_LABEL and do the following

1. set normal operation with M\_A429\_NORMAL\_OP\_BNR
2. set normal operation result with (i32\_arinc\_word Bitwise AND normal operation)

when

1. func\_state of a429chan (with index u8\_rx\_chan) of Appdata\_appdata is equal to ENABLED
2. A429\_mode is not equal to A429\_SELFTEST
3. i32\_arinc\_word is not equal to M\_ZERO
4. ((u8\_label of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) is equal to label) AND ((u8\_sdi of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) is equal to sdi) OR (u8\_sdi of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) is equal to SDI\_ALL))) returns TRUE
5. type of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) is equal to BNR

##### 10.1.5.7.7 daugwya429-ProcessArincWord-LLR-007

Requirement ID: H398-LLD-GWY-FNC-91

The function shall loop from loop index is M\_ZERO to M\_A429\_MAX\_LABEL minus 1 and do the following

1. set normal operation with M\_A429\_NORMAL\_OP\_DIS.
2. set normal operation result with (i32\_arinc\_word Bitwise AND M\_A429\_NORMAL\_OP)

when

1. func\_state of a429chan (with index u8\_rx\_chan) of Appdata\_appdata is equal to ENABLED
2. A429\_mode is not equal to A429\_SELFTEST
3. i32\_arinc\_word is not equal to M\_ZERO
4. ((u8\_label of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) is equal to label) AND ((u8\_sdi of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) is equal to sdi) OR (u8\_sdi of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) is equal to SDI\_ALL))) returns TRUE
5. type of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) is equal to DIS

##### 10.1.5.7.8 daugwya429-ProcessArincWord-LLR-008

Requirement ID: H398-LLD-GWY-FNC-92

The function shall loop from loop index is M\_ZERO to M\_A429\_MAX\_LABEL and do nothing

when

1. func\_state of a429chan (with index u8\_rx\_chan) of Appdata\_appdata is equal to ENABLED
2. A429\_mode is not equal to A429\_SELFTEST
3. i32\_arinc\_word is not equal to M\_ZERO
4. ((u8\_label of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) is equal to label) AND ((u8\_sdi of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) is equal to sdi) OR (u8\_sdi of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) is equal to SDI\_ALL))) returns TRUE
5. type of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) is other than BNR and DIS

##### 10.1.5.7.9 daugwya429-ProcessArincWord-LLR-010

Requirement ID: H398-LLD-GWY-FNC-94

The function shall loop from loop index is M\_ZERO to M\_A429\_MAX\_LABEL do the following

1. Set no\_lbl\_cnt of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) with the result of (u16\_update\_rate of A429\_aas\_chans with indices(u8\_rx\_chan,loop index) multiplied by M\_THREE) divided by M\_TEN.
2. Set lbl\_miss of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) with TRUE
3. Set status of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) with MISSING
4. Call ProcessLabel with parameters (u8\_rx\_chan, loop index, i32\_arinc\_word)

when

1. func\_state of a429chan (with index u8\_rx\_chan) of Appdata\_appdata is equal to ENABLED
2. A429\_mode is not equal to A429\_SELFTEST
3. i32\_arinc\_word is not equal to M\_ZERO
4. ((u8\_label of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) is equal to label) AND ((u8\_sdi of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) is equal to sdi) OR (u8\_sdi of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) is equal to SDI\_ALL))) returns TRUE
5. ((normal operation result is not equal to normal op) OR (parity for the indices (u8\_rx\_chan, loop index) is TRUE)) returns TRUE
6. (no\_lbl\_cnt of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) is greater than (product of u16\_update\_rate of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) and M\_THREE)) is divided by M\_TEN, returns TRUE

##### 10.1.5.7.10 daugwya429-ProcessArincWord-LLR-012

Requirement ID: H398-LLD-GWY-FNC-96

The function shall loop from loop index is M\_ZERO to M\_A429\_MAX\_LABEL and do the following

1. Set lbl\_miss of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) with TRUE
2. Set status of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) with CHAN\_ERR

when

1. func\_state of a429chan (with index u8\_rx\_chan) of Appdata\_appdata is equal to ENABLED
2. A429\_mode is not equal to A429\_SELFTEST
3. i32\_arinc\_word is not equal to M\_ZERO
4. ((u8\_label of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) is equal to label) AND ((u8\_sdi of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) is equal to sdi) OR (u8\_sdi of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) is equal to SDI\_ALL))) returns TRUE
5. ((normal operation result is not equal to normal op) OR (parity for the indices (u8\_rx\_chan, loop index) is TRUE)) returns TRUE
6. (no\_lbl\_cnt of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) is equal to (product of u16\_update\_rate of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) and M\_THREE)) returns FALSE

##### 10.1.5.7.11 daugwya429-ProcessArincWord-LLR-013

Requirement ID: H398-LLD-GWY-FNC-97

The function shall loop from loop index is M\_ZERO to M\_A429\_MAX\_LABEL and do the following

1. Set i32\_reading of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) with (bit shift of i32\_arinc\_word to right by (subtraction of M\_32BIT\_ARINCWORD with u8\_sigbits of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index)))

when

1. func\_state of a429chan (with index u8\_rx\_chan) of Appdata\_appdata is equal to ENABLED
2. A429\_mode is not equal to A429\_SELFTEST
3. i32\_arinc\_word is not equal to M\_ZERO
4. ((u8\_label of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) is equal to label) AND ((u8\_sdi of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) is equal to sdi) OR (u8\_sdi of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) is equal to SDI\_ALL))) returns TRUE
5. ((normal operation result is not equal to normal op) OR (parity for the indices (u8\_rx\_chan, loop index) is TRUE)) returns TRUE
6. (no\_lbl\_cnt of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) is equal to (product of u16\_update\_rate of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) and M\_THREE)) returns FALSE
7. type of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) is not equal to DIS
8. sign\_bit of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) is equal to SIGN

##### 10.1.5.7.12 daugwya429-ProcessArincWord-LLR-014

Requirement ID: H398-LLD-GWY-FNC-98

The function shall loop from loop index is M\_ZERO to M\_A429\_MAX\_LABEL and do the following

1. Set i32\_reading of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) with (bit shift of i32\_arinc\_word to right by (subtraction of M\_32BIT\_ARINCWORD with u8\_sigbits of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index)))

when

1. func\_state of a429chan (with index u8\_rx\_chan) of Appdata\_appdata is equal to ENABLED
2. A429\_mode is not equal to A429\_SELFTEST
3. i32\_arinc\_word is not equal to M\_ZERO
4. ((u8\_label of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) is equal to label) AND ((u8\_sdi of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) is equal to sdi) OR (u8\_sdi of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) is equal to SDI\_ALL))) returns TRUE
5. ((normal operation result is not equal to normal op) OR (parity for the indices (u8\_rx\_chan, loop index) is TRUE)) returns TRUE
6. (no\_lbl\_cnt of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) is equal to (product of u16\_update\_rate of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) and M\_THREE)) returns FALSE
7. type of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) is not equal to DIS
8. sign\_bit of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) is not equal to SIGN

##### 10.1.5.7.13 daugwya429-ProcessArincWord-LLR-015

Requirement ID: H398-LLD-GWY-FNC-99

The function shall loop from loop index is M\_ZERO to M\_A429\_MAX\_LABEL and do the following

1. Call ProcessLabel with parameters (u8\_rx\_chan, loop index, i32\_arinc\_word)

when

1. func\_state of a429chan (with index u8\_rx\_chan) of Appdata\_appdata is equal to ENABLED
2. A429\_mode is not equal to A429\_SELFTEST
3. i32\_arinc\_word is not equal to M\_ZERO
4. ((u8\_label of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) is equal to label) AND ((u8\_sdi of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) is equal to sdi) OR (u8\_sdi of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) is equal to SDI\_ALL))) returns TRUE
5. ((normal operation result is not equal to normal op) OR (parity for the indices (u8\_rx\_chan, loop index) is TRUE)) returns TRUE
6. (no\_lbl\_cnt of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) is equal to (product of u16\_update\_rate of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) and M\_THREE)) returns FALSE
7. type of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) is not equal to DIS

##### 10.1.5.7.14 daugwya429-ProcessArincWord-LLR-016

Requirement ID: H398-LLD-GWY-FNC-100

The function shall loop from loop index is M\_ZERO to M\_A429\_MAX\_LABEL and do the following

1. Set i32\_reading of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) with ((bit shift of i32\_arinc\_word to right by (subtraction of M\_32BIT\_ARINCWORD\_DIS with u8\_sigbits of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index))))
2. Call ProcessLabel with parameters (u8\_rx\_chan, loop index, i32\_arinc\_word)

when

1. func\_state of a429chan (with index u8\_rx\_chan) of Appdata\_appdata is equal to ENABLED
2. A429\_mode is not equal to A429\_SELFTEST
3. i32\_arinc\_word is not equal to M\_ZERO
4. ((u8\_label of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) is equal to label) AND ((u8\_sdi of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) is equal to sdi) OR (u8\_sdi of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) is equal to SDI\_ALL))) returns TRUE
5. ((normal operation result is not equal to normal op) OR (parity for the indices (u8\_rx\_chan, loop index) is TRUE)) returns TRUE
6. (no\_lbl\_cnt of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) is equal to (product of u16\_update\_rate of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) and M\_THREE)) returns FALSE
7. type of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) is equal to DIS

##### 10.1.5.7.15 daugwya429-ProcessArincWord-LLR-017

Requirement ID: H398-LLD-GWY-FNC-101

The function shall loop from loop index is M\_ZERO to M\_A429\_MAX\_LABEL do the following

1. Set no\_lbl\_cnt of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) with M\_ZERO
2. Set lbl\_miss of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) with FALSE
3. Set status of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) with OK

when

1. func\_state of a429chan (with index u8\_rx\_chan) of Appdata\_appdata is equal to ENABLED
2. A429\_mode is not equal to A429\_SELFTEST
3. i32\_arinc\_word is not equal to M\_ZERO
4. ((u8\_label of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) is equal to label) AND ((u8\_sdi of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) is equal to sdi) OR (u8\_sdi of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) is equal to SDI\_ALL))) returns TRUE
5. ((normal operation result is not equal to normal op) OR (parity for the indices (u8\_rx\_chan, loop index) is TRUE)) returns FALSE
6. type of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) is not equal to DIS

##### 10.1.5.7.16 daugwya429-ProcessArincWord-LLR-018

Requirement ID: H398-LLD-GWY-FNC-102

The function shall loop from loop index is M\_ZERO to M\_A429\_MAX\_LABEL do the following

1. Set i32\_reading of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) with (bit shift of i32\_arinc\_word to right by (subtraction of M\_32BIT\_ARINCWORD with u8\_sigbits of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index)))

when

1. func\_state of a429chan (with index u8\_rx\_chan) of Appdata\_appdata is equal to ENABLED
2. A429\_mode is not equal to A429\_SELFTEST
3. i32\_arinc\_word is not equal to M\_ZERO
4. ((u8\_label of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) is equal to label) AND ((u8\_sdi of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) is equal to sdi) OR (u8\_sdi of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) is equal to SDI\_ALL))) returns TRUE
5. ((normal operation result is not equal to normal op) OR (parity for the indices (u8\_rx\_chan, loop index) is TRUE)) returns FALSE
6. type of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) is not equal to DIS
7. sign\_bit of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) is equal to SIGN

##### 10.1.5.7.17 daugwya429-ProcessArincWord-LLR-019

Requirement ID: H398-LLD-GWY-FNC-103

The function shall loop from loop index is M\_ZERO to M\_A429\_MAX\_LABEL do the following

1. Set i32\_reading of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) with (bit shift of i32\_arinc\_word to right by (subtraction of M\_32BIT\_ARINCWORD with u8\_sigbits of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index)))

when

1. func\_state of a429chan (with index u8\_rx\_chan) of Appdata\_appdata is equal to ENABLED
2. A429\_mode is not equal to A429\_SELFTEST
3. i32\_arinc\_word is not equal to M\_ZERO
4. ((u8\_label of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) is equal to label) AND ((u8\_sdi of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) is equal to sdi) OR (u8\_sdi of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) is equal to SDI\_ALL))) returns TRUE
5. ((normal operation result is not equal to normal op) OR (parity for the indices (u8\_rx\_chan, loop index) is TRUE)) returns FALSE
6. type of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) is not equal to DIS
7. sign\_bit of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) is not equal to SIGN

##### 10.1.5.7.18 daugwya429-ProcessArincWord-LLR-020

Requirement ID: H398-LLD-GWY-FNC-104

The function shall loop from loop index is M\_ZERO to M\_A429\_MAX\_LABEL do the following

1. Call ProcessLabel with parameters (u8\_rx\_chan, loop index, i32\_arinc\_word)

when

1. func\_state of a429chan (with index u8\_rx\_chan) of Appdata\_appdata is equal to ENABLED
2. A429\_mode is not equal to A429\_SELFTEST
3. i32\_arinc\_word is not equal to M\_ZERO
4. ((u8\_label of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) is equal to label) AND ((u8\_sdi of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) is equal to sdi) OR (u8\_sdi of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) is equal to SDI\_ALL))) returns TRUE
5. ((normal operation result is not equal to normal op) OR (parity for the indices (u8\_rx\_chan, loop index) is TRUE)) returns FALSE
6. type of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) is not equal to DIS

##### 10.1.5.7.19 daugwya429-ProcessArincWord-LLR-021

Requirement ID: H398-LLD-GWY-FNC-105

The function shall loop from loop index is M\_ZERO to M\_A429\_MAX\_LABEL do the following

1. Set no\_lbl\_cnt of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) with M\_ZERO
2. Set lbl\_miss of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) with FALSE
3. Set status of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) with OK
4. Set i32\_reading of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) with ((bit shift of i32\_arinc\_word to right by (subtraction of M\_32BIT\_ARINCWORD\_DIS with u8\_sigbits of A429\_aas\_chans for the indices (u8\_rx\_chan, u8\_idx))))
5. Call ProcessLabel with parameters (u8\_rx\_chan, loop index, i32\_arinc\_word)

when

1. func\_state of a429chan (with index u8\_rx\_chan) of Appdata\_appdata is equal to ENABLED
2. A429\_mode is not equal to A429\_SELFTEST
3. i32\_arinc\_word is not equal to M\_ZERO
4. ((u8\_label of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) is equal to label) AND ((u8\_sdi of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) is equal to sdi) OR (u8\_sdi of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) is equal to SDI\_ALL))) returns TRUE
5. ((normal operation result is not equal to normal op) OR (parity for the indices (u8\_rx\_chan, loop index) is TRUE)) returns FALSE
6. type of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) is equal to DIS

##### 10.1.5.7.20 daugwya429-ProcessArincWord-LLR-023

Requirement ID: H398-LLD-GWY-FNC-107

The function shall loop from loop index is M\_ZERO to M\_A429\_MAX\_LABEL minus 1 do the following

1. Set no\_lbl\_cnt of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) to the (product of u16\_update\_rate of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) and M\_SIX)) is divided by M\_TEN.
2. Set lbl\_miss of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) with TRUE

when

1. func\_state of a429chan (with index u8\_rx\_chan) of Appdata\_appdata is equal to ENABLED
2. A429\_mode is not equal to A429\_SELFTEST
3. i32\_arinc\_word is not equal to M\_ZERO
4. ((u8\_label of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) is equal to label) AND ((u8\_sdi of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) is equal to sdi) OR (u8\_sdi of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) is equal to SDI\_ALL))) returns FALSE
5. (no\_lbl\_cnt of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) is greater than (product of u16\_update\_rate of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) and M\_SIX)) divided by M\_TEN returns TRUE

##### 10.1.5.7.21 daugwya429-ProcessArincWord-LLR-024

Requirement ID: H398-LLD-GWY-FNC-108

The function shall loop from loop index is M\_ZERO to M\_A429\_MAX\_LABEL do nothing

when

1. func\_state of a429chan (with index u8\_rx\_chan) of Appdata\_appdata is equal to ENABLED
2. A429\_mode is not equal to A429\_SELFTEST
3. i32\_arinc\_word is not equal to M\_ZERO
4. ((u8\_label of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) is equal to label) AND ((u8\_sdi of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) is equal to sdi) OR (u8\_sdi of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) is equal to SDI\_ALL))) returns FALSE
5. (no\_lbl\_cnt of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) is equal to product of u16\_update\_rate of A429\_aas\_chans for the indices (u8\_rx\_chan, loop index) and M\_SIX) returns FALSE

##### 10.1.5.7.22 daugwya429-ProcessArincWord-LLR-025

Requirement ID: H398-LLD-GWY-FNC-110

The function shall do nothing when func\_state of a429chan (with index u8\_rx\_chan) of Appdata\_appdata is not equal to ENABLED

### 10.1.6 ReceiveArinc429

Low Level Design Details about CSU ReceiveArinc429 will follow in the sub sections.

#### 10.1.6.1 Brief Description

The ReceiveArinc429 function reads the content of Receiver 1 and Receiver 2 of all the four ARINC 429 chips.

#### 10.1.6.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.1.6.3 List of global variables accessed and modified

Accessed :None

Modified : None

#### 10.1.6.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.1.6.5 Return Value

None

#### 10.1.6.6 Other CSUs called by this CSU

ProcessArincWord

#### 10.1.6.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to ReceiveArinc429.

##### 10.1.6.7.1 daugwya429-ReceiveArinc429-LLR-008

Requirement ID: H398-LLD-GWY-FNC-6149

The function shall loop from loop index is M\_ZERO to M\_A429\_IC\_COUNT minus 1 and also loop from label index M\_ZERO to M\_A429\_MAX\_LABEL minus 1 and do the following

1. Increment the no\_lbl\_cnt of A429\_aas\_chans for the indices (product of loop index and M\_TWO, label index) by one.
2. Increment the no\_lbl\_cnt of A429\_aas\_chans for the indices ((product of loop index\* M\_TWO) plus M\_ONE, label index) by one.

##### 10.1.6.7.2 daugwya429-ReceiveArinc429-LLR-001

Requirement ID: H398-LLD-GWY-FNC-119

The function shall loop from loop index is M\_ZERO to M\_A429\_IC\_COUNT and set address pointer with (addition of M\_A429\_BASE\_ADDR and product of (loop index and M\_A429\_BLOCK\_SIZE))

##### 10.1.6.7.3 daugwya429-ReceiveArinc429-LLR-002

Requirement ID: H398-LLD-GWY-FNC-120

The function shall loop from loop index is M\_ZERO to M\_A429\_IC\_COUNTand then loop until (status Bitwise AND with (M\_A429\_STATUS\_RX1\_MASK Bitwise OR M\_A429\_STATUS\_RX2\_MASK)) and set the status with content of (addition of address pointer and M\_A429\_STATUS\_REG\_RD)

##### 10.1.6.7.4 daugwya429-ReceiveArinc429-LLR-003

Requirement ID: H398-LLD-GWY-FNC-121

The function shall loop from loop index is M\_ZERO to M\_A429\_IC\_COUNTand then loop until (status Bitwise AND with (M\_A429\_STATUS\_RX1\_MASK Bitwise OR M\_A429\_STATUS\_RX2\_MASK)) and do the following:

1. Set 32 bit arinc word with content of (addition of address pointer and M\_A429\_RX1\_FIFO\_LSB)
2. Update 32 bit arinc word by performing Bitwise OR of 32 bit arinc word with (content of bit shift of (addition of address pointer and M\_A429\_RX1\_FIFO\_MSB) to left by M\_SHIFT\_BY\_16)
3. Call ProcessArincWord with parameters ((product of loop index and M\_TWO),32 bit arinc word)

When,

Bitwise AND of (status with M\_A429\_STATUS\_RX1\_MASK) is not equal to M\_ZERO

##### 10.1.6.7.5 daugwya429-ReceiveArinc429-LLR-004

Requirement ID: H398-LLD-GWY-FNC-122

The function shall loop from loop index is M\_ZERO to M\_A429\_IC\_COUNTand then loop until (status Bitwise AND with (M\_A429\_STATUS\_RX1\_MASK Bitwise OR M\_A429\_STATUS\_RX2\_MASK)) and Call ProcessArincWord with parameters ((product of loop index and M\_TWO), M\_ZERO) when,

Bitwise AND of (status with M\_A429\_STATUS\_RX1\_MASK) is equal to M\_ZERO

##### 10.1.6.7.6 daugwya429-ReceiveArinc429-LLR-005

Requirement ID: H398-LLD-GWY-FNC-123

The function shall loop from loop index is M\_ZERO to M\_A429\_IC\_COUNTand then loop until (status Bitwise AND with (M\_A429\_STATUS\_RX1\_MASK Bitwise OR M\_A429\_STATUS\_RX2\_MASK)) and do the following:

1. Set 32 bit arinc word with content of (addition of address pointer and M\_A429\_RX2\_FIFO\_LSB)
2. Update 32 bit arinc word by performing Bitwise OR of 32 bit arinc word with (content of bit shift of (addition of address pointer and M\_A429\_RX2\_FIFO\_MSB) to left by M\_SHIFT\_BY\_16)
3. Call ProcessArincWord with parameters ((addition of (product of loop index and M\_TWO) and M\_ONE),32 bit arinc word)

When,

Bitwise AND of (status with M\_A429\_STATUS\_RX2\_MASK) is not equal to M\_ZERO

##### 10.1.6.7.7 daugwya429-ReceiveArinc429-LLR-006

Requirement ID: H398-LLD-GWY-FNC-124

The function shall loop from loop index is M\_ZERO to M\_A429\_IC\_COUNTand then loop until (status Bitwise AND with (M\_A429\_STATUS\_RX1\_MASK Bitwise OR M\_A429\_STATUS\_RX2\_MASK)) and Call ProcessArincWord with parameters (addition of (product of loop index and M\_TWO) and M\_ONE, M\_ZERO) when,

Bitwise AND of (status with M\_A429\_STATUS\_RX2\_MASK) is equal to M\_ZERO

##### 10.1.6.7.8 daugwya429-ReceiveArinc429-LLR-007

Requirement ID: H398-LLD-GWY-FNC-125

The function shall loop from loop index is M\_ZERO to M\_A429\_IC\_COUNTand then loop until (status Bitwise AND with (M\_A429\_STATUS\_RX1\_MASK Bitwise OR M\_A429\_STATUS\_RX2\_MASK)) and set status with content of (addition of address pointer and M\_A429\_STATUS\_REG\_RD)

### 10.1.7 TransmitArinc429

Low Level Design Details about CSU TransmitArinc429 will follow in the sub sections.

#### 10.1.7.1 Brief Description

The TransmitArinc429 function loads the 32-bit ARINC word to transmit FIFO.

#### 10.1.7.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.1.7.3 List of global variables accessed and modified

Accessed               : None

Modified                 :  I32\_arinc\_checksum

#### 10.1.7.4 Parameter list (Input/Output)

Inputs : T\_UBYTE u8\_chan -channel number

T\_UINT8 u8\_label -ARINC label

T\_SINT32 i32\_data -ARINC data

T\_UINT8 u8\_sigbits -Number of Significant bits

T\_UINT16 id -identifier

T\_SINT32 a429\_sdi -ARINC Source destination index

T\_BOOL sign - Sign bit

T\_DATA\_STAT status - status of label data

Outputs : T\_SINT32 i32\_data -ARINC data

T\_UINT8 u8\_sigbits -Number of Significant bits

#### 10.1.7.5 Return Value

T\_SINT32 - Return the status of transmit FIFO,

return FALSE if transmit FIFO is full,

return TRUE if data is loaded to transmit FIFO successfully.

#### 10.1.7.6 Other CSUs called by this CSU

HwAbs32

#### 10.1.7.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TransmitArinc429.

##### 10.1.7.7.1 daugwya429-TransmitArinc429-LLR-001

Requirement ID: H398-LLD-GWY-FNC-134

The function shall set chip address to (addition of M\_A429\_BASE\_ADDR with product of (u8\_chan and M\_A429\_BLOCK\_SIZE))

##### 10.1.7.7.2 daugwya429-TransmitArinc429-LLR-002

Requirement ID: H398-LLD-GWY-FNC-135

The function shall do the following when u8\_sigbits is greater than M\_TWENTY:

1. Set i32\_data with Bitwise OR of (bit shift of i32\_data to left by TWO) and a429\_sdi
2. Set move bits with Bitwise AND of i32\_data and M\_MOVE\_BITS\_MASK\_VAL.
3. Set move bits with bit shift of move bits to right by M\_TWELVE

##### 10.1.7.7.3 daugwya429-TransmitArinc429-LLR-003

Requirement ID: H398-LLD-GWY-FNC-136

The function shall do the following when u8\_sigbits is not greater than M\_TWENTY:

-set i32\_data to (i32\_data left shifted by the return value of HwAbs32 with parameters (difference of (M\_TWENTY and u8\_sigbits))) Bitwise OR with (id left shifted by M\_TWO) Bitwise OR with a429\_sdi

-set u8\_sigbits to sum of (u8\_sigbits and (difference of M\_TWENTY and u8\_sigbits))

##### 10.1.7.7.4 daugwya429-TransmitArinc429-LLR-004

Requirement ID: H398-LLD-GWY-FNC-137

The function shall return FALSE when ((Bitwise AND of content of (addition of chip address and M\_A429\_STATUS\_REG\_RD) with M\_A429\_STATUS\_TX\_FULL\_MASK) is not equal to M\_DEC\_ZERO)

##### 10.1.7.7.5 daugwya429-TransmitArinc429-LLR-005

Requirement ID: H398-LLD-GWY-FNC-138

The function shall do the following when sign equal to TRUE AND status equal to OK

1. Set 32 bit arinc word with (M\_EXTRACT\_SSM Bitwise OR u8\_label)
2. Update 32 bit arinc word with Bitwise OR of 32 bit arinc word and (bit shift of i32\_data to left by (subtraction of M\_THIRTYONE with u8\_sigbits))

##### 10.1.7.7.6 daugwya429-TransmitArinc429-LLR-006

Requirement ID: H398-LLD-GWY-FNC-139

The function shall do the following

1. Set 32 bit arinc word with Bitwise OR of move bits and u8\_label
2. Update 32 bit arinc word with Bitwise OR of 32 bit arinc word and (bit shift of i32\_data to left by M\_ELEVEN)

when ,

1. (sign equal to TRUE AND status equal to OK) is FALSE
2. u8\_sigbits is greater than M\_TWENTY AND status is equal to OK

##### 10.1.7.7.7 daugwya429-TransmitArinc429-LLR-007

Requirement ID: H398-LLD-GWY-FNC-140

The function shall do the following

1. Update 32 bit arinc word with Bitwise OR of (M\_ARINC\_WORD\_CHAN\_ERR\_MASK and u8\_label).
2. Update 32 bit arinc word with Bitwise OR of (32 bit arinc word and (i32\_data LEFT SHIFT by (M\_THIRTYONE minus u8\_sigbits))

when ,

1. status is equal to either CHAN\_ERR or RANGE\_ERR
2. (sign equal to TRUE AND status equal to OK) is FALSE
3. (u8\_sigbits is greater than twenty AND status is OK) is FALSE

##### 10.1.7.7.8 daugwya429-TransmitArinc429-LLR-008

Requirement ID: H398-LLD-GWY-FNC-141

The function shall do the following

1. Set 32 bit arinc word with Bitwise OR of M\_ARINC\_WORD\_MISSING\_MASK and u8\_label
2. Update 32 bit arinc word with Bitwise OR of 32 bit arinc word and (bit shift of i32\_data to left by (subtraction of M\_THIRTYONE with u8\_sigbits))

when ,

1. (sign equal to TRUE AND status equal to OK) is FALSE
2. (u8\_sigbits is greater than twenty AND status is OK) is FALSE
3. status is equal to MISSING

##### 10.1.7.7.9 daugwya429-TransmitArinc429-LLR-009

Requirement ID: H398-LLD-GWY-FNC-142

The function shall do the following

1. Set 32 bit arinc word with Bitwise OR of M\_ARINC\_WORD\_FUNC\_TEST\_MASK and u8\_label
2. Update 32 bit arinc word with Bitwise OR of 32 bit arinc word and (bit shift of i32\_data to left by (subtraction of M\_THIRTYONE with u8\_sigbits))

when ,

1. (sign equal to TRUE AND status equal to OK) is FALSE
2. (u8\_sigbits is greater than twenty AND status is OK) is FALSE
3. status is equal to FUNC\_TEST

##### 10.1.7.7.10 daugwya429-TransmitArinc429-LLR-010

Requirement ID: H398-LLD-GWY-FNC-143

The function shall do the following

1. Set 32 bit arinc word with Bitwise OR of M\_A429\_NORMAL\_OP and u8\_label
2. Update 32 bit arinc word with Bitwise OR of 32 bit arinc word and (bit shift of i32\_data to left by (subtraction of M\_THIRTYONE with u8\_sigbits))

when ,

1. (sign equal to TRUE AND status equal to OK) is FALSE
2. (u8\_sigbits is greater than twenty AND status is OK) is FALSE
3. status is other than CHAN\_ERR, MISSING and FUNC\_TEST
4. sign is equal to FALSE AND status is equal to OK

##### 10.1.7.7.11 daugwya429-TransmitArinc429-LLR-011

Requirement ID: H398-LLD-GWY-FNC-144

The function shall set (addition of chip address and M\_A429\_TX\_FIFO\_LSB) with (32 bit arinc word Bitwise AND M\_0X0000FFFF)

##### 10.1.7.7.12 daugwya429-TransmitArinc429-LLR-012

Requirement ID: H398-LLD-GWY-FNC-145

The function shall do the following:

1. Set (addition of chip address and M\_A429\_TX\_FIFO\_MSB) with ((32 bit arinc word right shifted by M\_SHIFT\_BY\_16 )Bitwise AND M\_0X0000FFFF)
2. Return variable 32 bit arinc word.

### 10.1.8 A429Task

Low Level Design Details about CSU A429Task will follow in the sub sections.

#### 10.1.8.1 Brief Description

The A429Task function receives ARINC words and performs CBIT.

#### 10.1.8.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.1.8.3 List of global variables accessed and modified

Accessed : A429\_mode, A429\_chan\_id, Inita429, Cmdownload\_lp\_cmd\_data, Tx\_a429\_ddw,App\_emode

Modified : None

#### 10.1.8.4 Parameter list (Input/Output)

Inputs : void \*pdata – data pointer

Outputs : None

#### 10.1.8.5 Return Value

None

#### 10.1.8.6 Other CSUs called by this CSU

OsSemPend

ReceiveArinc429

LabelPriority

A429StatusMiscompare

TransmitArinc429

CalcArincWordChecksum

#### 10.1.8.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to A429Task.

##### 10.1.8.7.1 daugwya429-A429Task-LLR-001

Requirement ID: H398-LLD-GWY-FNC-154

The function shall set data pointer with M\_HW\_NULL

##### 10.1.8.7.2 daugwya429-A429Task-LLR-002

Requirement ID: H398-LLD-GWY-FNC-155

The function shall loop forever and call OsSemPend with parameters (Semaphore a429 task, M\_ZERO, reference of error status)

##### 10.1.8.7.3 daugwya429-A429Task-LLR-003

Requirement ID: H398-LLD-GWY-FNC-156

The function shall loop forever, loop from loop index is M\_ZERO to M\_A429\_IC\_COUNT and do the following

1. Set IC address with (addition of M\_A429\_BASE\_ADDR and (product of loop index and M\_A429\_BLOCK\_SIZE)
2. Set value at location (addition of IC address and M\_A429\_MASTER\_RESET) with M\_MS\_LOW
3. Set value at location (addition of IC address and M\_A429\_MASTER\_RESET) with M\_MS\_HIGH
4. Set value at location (addition of IC address and M\_A429\_CNTRL\_REG\_WR) with M\_A429\_CNTRL\_SELF\_TEST\_EN

When,

1. A429\_mode is A429\_SELFTEST
2. self-test state is TRUE

##### 10.1.8.7.4 daugwya429-A429Task-LLR-004

Requirement ID: H398-LLD-GWY-FNC-157

The function shall loop forever, loop from loop index is M\_ZERO to M\_A429\_IC\_COUNT and do the following

1. Set self-test state to FALSE
2. Set First normal state to TRUE

When,

1. A429\_mode is A429\_SELFTEST
2. self-test state is TRUE

##### 10.1.8.7.5 daugwya429-A429Task-LLR-005

Requirement ID: H398-LLD-GWY-FNC-158

The function shall loop forever and do nothing

When,

1. A429\_mode is A429\_SELFTEST
2. self-test state is not TRUE

##### 10.1.8.7.6 daugwya429-A429Task-LLR-006

Requirement ID: H398-LLD-GWY-FNC-159

The function shall loop forever and do the following

1. increment the local counter by one
2. Set Test sign bits to M\_TEST\_DATA\_SIG19

When,

1. A429\_mode is equal to A429\_SELFTEST
2. local counter is equal to M\_ZERO

##### 10.1.8.7.7 daugwya429-A429Task-LLR-007

Requirement ID: H398-LLD-GWY-FNC-160

The function shall loop from M\_ZERO to M\_A429\_MAX\_RX\_CHAN minus 1 and perform the following when A429\_mode is equal to A429\_SELFTEST, local counter is equal to M\_ZERO

a) call TransmitArinc429 function with parameters (loop index, M\_ARINC\_LABEL\_0270, M\_ARINC\_TEST\_DATA, M\_TEST\_DATA\_SIG19, M\_ZERO, M\_ZERO, FALSE, OK) when (sys\_connected of a429chan with index as loop index of Appdata\_appdata is equals to CONNECTED) AND (func\_state of a429chan with index as loop index of Appdata\_appdata is equals to ENABLED) otherwise do nothing.

##### 10.1.8.7.8 daugwya429-A429Task-LLR-015

Requirement ID: H398-LLD-GWY-FNC-168

The function shall Call ReceiveArinc429 to read contents of ARINC 429 receive FIFOs

When,

1. A429\_mode is equal to A429\_SELFTEST
2. local counter is not equal to M\_ZERO

##### 10.1.8.7.9 daugwya429-A429Task-LLR-029

Requirement ID: H398-LLD-GWY-FNC-6184

The function shall loop from M\_ZERO to M\_A429\_MAX\_RX\_CHAN minus 1 and calls RterrError function with parameters (A429\_LOOPBACK\_ERR, u8 idx, reference of function pointer RterrForever) and Set A429\_loop\_back\_fail\_flag to TRUE when

a) A429\_mode is equal to A429\_SELFTEST

b) local counter is not equal to M\_ZERO

c) (sys\_connected of a429chan with index as loop index of Appdata\_appdata is equals to CONNECTED) AND (func\_state of a429chan with index as loop index of Appdata\_appdata is equals to ENABLED).

d) Arinc\_loop\_back\_status with index as loop index is equals to FALSE.

##### 10.1.8.7.10 daugwya429-A429Task-LLR-030

Requirement ID: H398-LLD-GWY-FNC-6185

The function shall Set A429 mode to A429\_NORMAL and Reset the local counter to M\_ZERO When,

a) A429\_mode is equal to A429\_SELFTEST

b) local counter is not equal to M\_ZERO

##### 10.1.8.7.11 daugwya429-A429Task-LLR-016

Requirement ID: H398-LLD-GWY-FNC-169

The function shall loop forever, loop from loop index is M\_ZERO to M\_A429\_IC\_COUNT and do the following

1. Set IC address with (addition of M\_A429\_BASE\_ADDR and (product of loop index and M\_A429\_BLOCK\_SIZE))
2. Set value at the location (addition of IC address and M\_A429\_MASTER\_RESET) with M\_MS\_LOW
3. Set value at the location (addition of IC address and M\_A429\_MASTER\_RESET) with M\_MS\_HIGH
4. Set value at the location (addition of IC address and M\_A429\_CNTRL\_REG\_WR) with ( M\_A429\_CNTRL\_PARITY\_EN Bitwise OR M\_A429\_CNTRL\_SELF\_TEST\_DIS)

When,

1. A429\_mode is not equal to A429\_SELFTEST
2. ((App\_emode is equal to APP\_NORMAL ) AND Inita429) returns TRUE
3. First normal state is equal to TRUE

##### 10.1.8.7.12 daugwya429-A429Task-LLR-017

Requirement ID: H398-LLD-GWY-FNC-170

The function shall loop forever, loop from loop index is M\_ZERO to M\_A429\_IC\_COUNT and do the following

1. Set self-test state to TRUE
2. Set First normal state to FALSE  
   When,
3. A429\_mode is not equal to A429\_SELFTEST
4. (App\_emode is equal to APP\_NORMAL ) AND Inita429 returns TRUE

First normal state is equal to TRUE

##### 10.1.8.7.13 daugwya429-A429Task-LLR-018

Requirement ID: H398-LLD-GWY-FNC-171

The function shall loop forever and do the following

1. Call ReceiveArinc429 to read contents of ARINC 429
2. Call LabelPriority

When,

1. A429\_mode is not equal to A429\_SELFTEST
2. ((App\_emode is equal to APP\_NORMAL ) AND Inita429) returns TRUE
3. First normal state is not equal to TRUE

##### 10.1.8.7.14 daugwya429-A429Task-LLR-019

Requirement ID: H398-LLD-GWY-FNC-172

The function shall loop forever, loop from index is M\_ZERO to (numOf429out\_ddw of Cmdownload\_lp\_cmd\_data AND App\_strap)minus one and set output counter ddw to M\_ONE.

When,

1. A429\_mode is not equal to A429\_SELFTEST
2. ((App\_emode is equal to APP\_NORMAL ) AND Inita429) returns TRUE
3. First normal state is not equal to TRUE
4. (output counter ddw with index (index) is greater than or equal to division of (update\_rate of A429out\_ddw with index(index) of Cmdownload\_lp\_cmd\_data is multiplied with M\_A429\_MSECOND) by M\_TEN) returns TRUE

##### 10.1.8.7.15 daugwya429-A429Task-LLR-020

Requirement ID: H398-LLD-GWY-FNC-173

The function shall loop forever, loop from index is (M\_ZERO to numOf429out\_ddw of Cmdownload\_lp\_cmd\_dataAND App\_strap)minus one, loop from channel loop is zero to M\_A429\_MAX\_TX\_CHAN and do the following

1. Call A429StatusMiscompare with parameters A429D\_DIS, index, status of Tx\_a429\_ddw with index (index)
2. Set status with return value of A429StatusMiscompare

When,

1. A429\_mode is not equal to A429\_SELFTEST
2. ((App\_emode is equal to APP\_NORMAL ) AND Inita429) returns TRUE
3. First normal state is not equal to TRUE
4. (output counter ddw with index (index) is greater than or equal to division of (update\_rate of A429out\_ddw with index(index) of Cmdownload\_lp\_cmd\_data is multiplied with M\_A429\_MSECOND) by ten) returns TRUE
5. (((channel of A429out\_ddw(with index value as index) of Cmdownload\_lp\_cmd\_data right shifted by channel loop) Bitwise AND M\_LSB\_FIRST\_BIT\_MASK) not equal to M\_ZERO) returns TRUE
6. (type of A429out\_ddw (with index value as index) of Cmdownload\_lp\_cmd\_data not equal to A429D\_BNR) returns TRUE

##### 10.1.8.7.16 daugwya429-A429Task-LLR-021

Requirement ID: H398-LLD-GWY-FNC-174

The function shall loop forever, loop from index is (M\_ZERO to numOf429out\_ddw of Cmdownload\_lp\_cmd\_data AND App\_strap)minus one, loop from channel loop is M\_ZERO to M\_A429\_MAX\_TX\_CHAN and set status with CHAN\_ERR

When,

1. A429\_mode is not equal to A429\_SELFTEST
2. ((App\_emode is equal to APP\_NORMAL ) AND Inita429) returns TRUE
3. First normal state is not equal to TRUE
4. (output counter ddw with index (index) is greater than or equal to division of (update\_rate of A429out\_ddw with index(index) of Cmdownload\_lp\_cmd\_data is multiplied with M\_A429\_MSECOND) by ten) returns TRUE
5. (((channel of A429out\_ddw(with index value as index) of Cmdownload\_lp\_cmd\_data right shifted by channel loop) Bitwise AND M\_LSB\_FIRST\_BIT\_MASK) not equal to M\_ZERO) returns TRUE
6. (type of A429out\_ddw (with index value as index) of Cmdownload\_lp\_cmd\_data not equal to A429D\_BNR) returns TRUE
7. status is equal to OK

##### 10.1.8.7.17 daugwya429-A429Task-LLR-022

Requirement ID: H398-LLD-GWY-FNC-175

The function shall loop forever, (loop from index is M\_ZERO to numOf429out\_ddw of Cmdownload\_lp\_cmd\_data AND App\_strap)minus one, loop from channel loop is M\_ZERO to M\_A429\_MAX\_TX\_CHAN minus one and set status with OK

When,

1. A429\_mode is not equal to A429\_SELFTEST
2. ((App\_emode is equal to APP\_NORMAL ) AND Inita429) returns TRUE
3. First normal state is not equal to TRUE
4. (output counter ddw with index (index) is greater than or equal to division of (update\_rate of A429out\_ddw with index(index) of Cmdownload\_lp\_cmd\_data is multiplied with M\_A429\_MSECOND) by ten) returns TRUE
5. (((channel of A429out\_ddw(with index value as index) of Cmdownload\_lp\_cmd\_data right shifted by channel loop) Bitwise AND M\_LSB\_FIRST\_BIT\_MASK) not equal to M\_ZERO) returns TRUE
6. (type of A429out\_ddw (with index value as index) of Cmdownload\_lp\_cmd\_data not equal to A429D\_BNR) returns TRUE
7. status is equal to CHAN\_ERR

##### 10.1.8.7.18 daugwya429-A429Task-LLR-023

Requirement ID: H398-LLD-GWY-FNC-176

The function shall loop forever, loop counter from M\_ZERO to (numOf429out\_ddw of Cmdownload\_lp\_cmd\_data AND App\_strap)minus 1, loop from channel loop is M\_ZERO to M\_A429\_MAX\_TX\_CHAN minus 1 and do nothing

When,

a) A429\_mode is not equal to A429\_SELFTEST

b) ((App\_emode is equal to APP\_NORMAL ) AND Inita429) returns TRUE

c) First normal state is not equal to TRUE

d) (output counter ddw with index (index) is greater than or equal to division of (update\_rate of A429out\_ddw with index(index) of Cmdownload\_lp\_cmd\_data is multiplied with M\_A429\_MSECOND) by ten) returns TRUE

e) (((channel of A429out\_ddw(with index value as index) of Cmdownload\_lp\_cmd\_data right shifted by channel loop) Bitwise AND M\_LSB\_FIRST\_BIT\_MASK) not equal to M\_ZERO) returns TRUE

f) (type of A429out\_ddw (with index value as index) of Cmdownload\_lp\_cmd\_data not equal to A429D\_BNR) returns TRUE

g) (M\_ZERO is equal to resolution of A429out\_bnr with index as loop counter of Cmdownload\_lp\_cmd\_data by calling the macro M\_FP\_COMPARE)

##### 10.1.8.7.19 daugwya429-A429Task-LLR-024

Requirement ID: H398-LLD-GWY-FNC-177

The function shall loop forever, loop from index is M\_ZERO to (numOf429out\_ddw of Cmdownload\_lp\_cmd\_data AND App\_strap)minus one, loop from channel loop is M\_ZERO to M\_A429\_MAX\_TX\_CHAN minus one and do the following

1. Call TransmitArinc429 with parameters channel loop, label of A429out\_ddw (with index as index) of Cmdownload\_lp\_cmd\_data, ((data of Tx\_a429\_ddw (with index as index) subtracted with offsetValue of A429out\_ddw (with index as index) of Cmdownload\_lp\_cmd\_data) divided by resolution of A429out\_ddw (with index as index) of Cmdownload\_lp\_cmd\_data), sig\_bits of A429out\_ddw (with index as index) of Cmdownload\_lp\_cmd\_data, M\_DEC\_ZERO, id of Tx\_a429\_ddw (with index as index), signBit of A429out\_ddw (with index as index) of Cmdownload\_lp\_cmd\_data and status and set to variable ARINC word out

When,

1. A429\_mode is not equal to A429\_SELFTEST
2. ((App\_emode is equal to APP\_NORMAL ) AND Inita429) returns TRUE
3. First normal state is not equal to TRUE
4. (output counter ddw with index (index) is greater than or equal to division of (update\_rate of A429out\_ddw with index(index) of Cmdownload\_lp\_cmd\_data is multiplied with M\_A429\_MSECOND) by M\_TEN) returns TRUE
5. (((channel of A429out\_ddw(with index value as index) of Cmdownload\_lp\_cmd\_data right shifted by channel loop) Bitwise AND M\_LSB\_FIRST\_BIT\_MASK) not equal to M\_ZERO) returns TRUE
6. (type of A429out\_ddw (with index value as index) of Cmdownload\_lp\_cmd\_data not equal to A429D\_BNR) returns TRUE
7. resolution of A429out\_bnr (with index as index) of Cmdownload\_lp\_cmd\_data is not equal to M\_ZERO

##### 10.1.8.7.20 daugwya429-A429Task-LLR-033

Requirement ID: H398-LLD-GWY-FNC-7454

The function shall loop forever, loop from index is M\_ZERO to (numOf429out\_ddw of Cmdownload\_lp\_cmd\_data AND App\_strap)minus one, loop from channel loop is M\_ZERO to M\_A429\_MAX\_TX\_CHAN minus one and do the following

1. Call CalcArincWordChecksum with parameters variable ARINC word out and set the return value to a429 checksum 074.

2. Set the checksum data package to a429 checksum 074.

3.Update the checksum data package to bitwise OR of checksum data package and (label of A429out\_ddw of index size of count of Cmdownload\_lp\_cmd\_data shifted to left by M\_SHIFT\_BY\_8).

4. Update the checksum data package to bitwise OR of checksum data package and (id of Tx\_a429\_ddw of index size of count shifted left by M\_SHIFT\_BY\_16).

5. Call TransmitArinc429 function wiith parameters (channel loop, M\_LABEL\_074, checksum data package,  M\_EIGHTEEN, M\_ZERO, M\_HW\_GET\_RCI plus M\_ONE, A429\_DATA, OK).

When

1. label of A429out\_ddw of index size of count of Cmdownload\_lp\_cmd\_data is not equal to M\_LABEL\_035 AND label of A429out\_ddw of index size of count of Cmdownload\_lp\_cmd\_data is not equal to M\_LABEL\_036
2. A429\_mode is not equal to A429\_SELFTEST
3. ((App\_emode is equal to APP\_NORMAL ) AND Inita429) returns TRUE
4. First normal state is not equal to TRUE
5. (output counter ddw with index (index) is greater than or equal to (update\_rate of A429out\_ddw with index(index) of Cmdownload\_lp\_cmd\_data is multiplied with M\_A429\_MSECOND) by M\_TEN) returns TRUE
6. (((channel of A429out\_ddw(with index value as index) of Cmdownload\_lp\_cmd\_data right shifted by channel loop) Bitwise AND M\_LSB\_FIRST\_BIT\_MASK) not equal to M\_ZERO) returns TRUE
7. (type of A429out\_ddw (with index value as index) of Cmdownload\_lp\_cmd\_data not equal to A429D\_BNR) returns TRUE
8. resolution of A429out\_bnr (with index as index) of Cmdownload\_lp\_cmd\_data returns TRUE

##### 10.1.8.7.21 daugwya429-A429Task-LLR-025

Requirement ID: H398-LLD-GWY-FNC-178

The function shall loop forever, loop from index is M\_ZERO to (numOf429out\_ddw of Cmdownload\_lp\_cmd\_data AND App\_strap)minus one, loop from channel loop is M\_ZERO to M\_A429\_MAX\_TX\_CHAN minus one and do nothing

When,

1. A429\_mode is not equal to A429\_SELFTEST
2. ((App\_emode is equal to APP\_NORMAL ) AND Inita429) returns TRUE
3. First normal state is not equal to TRUE
4. (output counter ddw with index (index) is greater than or equal to division of (update\_rate of A429out\_ddw with index(index) of Cmdownload\_lp\_cmd\_data is multiplied with M\_A429\_MSECOND) by ten) returns TRUE
5. (((channel of A429out\_ddw(with index value as index) of Cmdownload\_lp\_cmd\_data right shifted by channel loop) Bitwise AND M\_LSB\_FIRST\_BIT\_MASK) not equal to M\_ZERO) returns TRUE
6. (type of A429out\_ddw (with index value as index) of Cmdownload\_lp\_cmd\_data not equal to A429D\_BNR) returns FALSE

##### 10.1.8.7.22 daugwya429-A429Task-LLR-026

Requirement ID: H398-LLD-GWY-FNC-179

The function shall loop forever, loop from index is M\_ZERO to (numOf429out\_ddw of Cmdownload\_lp\_cmd\_data AND App\_strap)minus one and increment output counter ddw (with index value as index) by one

When,

1. A429\_mode is not equal to A429\_SELFTEST
2. ((App\_emode is equal to APP\_NORMAL ) AND Inita429) returns TRUE
3. First normal state is not equal to TRUE
4. (output counter ddw with index (index) is greater than or equal to division of (update\_rate of A429out\_ddw with index(index) of Cmdownload\_lp\_cmd\_data is multiplied with M\_A429\_MSECOND) by ten) returns FALSE
5. Counter is less than or equal to loop ddw lables

##### 10.1.8.7.23 daugwya429-A429Task-LLR-028

Requirement ID: H398-LLD-GWY-FNC-6150

The function shall increment loop\_ddw by one till Max DDW labels

when

1. A429\_mode is not equal to A429\_SELFTEST
2. ((App\_emode is equal to APP\_NORMAL ) AND Inita429) returns TRUE

c) First normal state is not equal to TRUE

d) loop\_ddw less than numOf429out\_ddw of Cmdownload\_lp\_cmd\_data AND App\_strap.

##### 10.1.8.7.24 daugwya429-A429Task-LLR-027

Requirement ID: H398-LLD-GWY-FNC-180

The function shall loop forever and do nothing

When,

1. A429\_mode is not equal to A429\_SELFTEST
2. ((App\_emode is equal to APP\_NORMAL ) AND Inita429) returns FALSE

### 10.1.9 InitChannels

Low Level Design Details about CSU InitChannels will follow in the sub sections.

#### 10.1.9.1 Brief Description

The InitChannels function configures all four Arinc chips.

#### 10.1.9.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.1.9.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.1.9.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.1.9.5 Return Value

None

#### 10.1.9.6 Other CSUs called by this CSU

None

#### 10.1.9.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to InitChannels.

##### 10.1.9.7.1 daugwya429-InitChannels-LLR-001

Requirement ID: H398-LLD-GWY-FNC-189

The function shall loop from loop index M\_ZERO to M\_A429\_IC\_COUNT and set the following

1. Set IC address with (addition of M\_A429\_BASE\_ADDR with (product of loop index and M\_A429\_BLOCK\_SIZE))
2. Set value at location (addition of IC address and M\_A429\_MASTER\_RESET) with M\_MS\_LOW
3. Set value at location (addition of IC address and M\_A429\_MASTER\_RESET) with M\_MS\_HIGH
4. Set value at location (addition of IC address and M\_A429\_CNTRL\_REG\_WR) with Bitwise OR of ( M\_A429\_CNTRL\_PARITY\_EN and M\_A429\_CNTRL\_SELF\_TEST\_DIS)

### 10.1.10 Inita429Parameters

Low Level Design Details about CSU Inita429Parameters will follow in the sub sections.

#### 10.1.10.1 Brief Description

The Inita429Parameters function initializes all Arinc parameters.

#### 10.1.10.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.1.10.3 List of global variables accessed and modified

Accessed : Cmdownload\_lp\_cmd\_data,

Modified : A429\_aas\_chans, Rx\_a429

#### 10.1.10.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.1.10.5 Return Value

None

#### 10.1.10.6 Other CSUs called by this CSU

None

#### 10.1.10.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to Inita429Parameters.

##### 10.1.10.7.1 daugwya429-Inita429Parameters-LLR-001

Requirement ID: H398-LLD-GWY-FNC-198

The function shall loop from label value is M\_DEC\_ZERO to numOf429in of Cmdownload\_lp\_cmd\_data, loop from channel value is ZERO to M\_A429\_MAX\_RX\_CHAN and do the following

1. Set u8\_label of A429\_aas\_chans for indices (channel value,label value) with label of A429in (with index is label value) of Cmdownload\_lp\_cmd\_data
2. Set u8\_sdi of A429\_aas\_chans for indices (channel value,label value) with sdi of A429in (with index is label value) of Cmdownload\_lp\_cmd\_data
3. Set u16\_update\_rate of A429\_aas\_chans for indices (channel value,label value) with update\_rate of A429in (with index is label value) of Cmdownload\_lp\_cmd\_data
4. Set u8\_sigbits of A429\_aas\_chans for indices (channel value,label value) with sig\_bits of A429in (with index is label value) of Cmdownload\_lp\_cmd\_data
5. Set sign\_bit of A429\_aas\_chans for indices (channel value,label value) with signBit of A429in (with index is label value) of Cmdownload\_lp\_cmd\_data
6. Set i32\_scalefactor of A429\_aas\_chans for indices (channel value,label value) with resolution of A429in (with index is label value) of Cmdownload\_lp\_cmd\_data
7. Set default\_value of A429\_aas\_chans for indices (channel value,label value) with default\_value of A429in (with index is label value) of Cmdownload\_lp\_cmd\_data
8. Set i32\_reading of A429\_aas\_chans for indices (channel value,label value) with M\_DEC\_ZERO
9. Set type of A429\_aas\_chans for indices (channel value,label value) with type of A429in (with index is label value) of Cmdownload\_lp\_cmd\_data

When,

(((channel of A429in (with index as label value) of Cmdownload\_lp\_cmd\_data right shifted to channel value) Bitwise AND M\_LSB\_FIRST\_BIT\_MASK) not equal to M\_ZERO) returns TRUE

##### 10.1.10.7.2 daugwya429-Inita429Parameters-LLR-002

Requirement ID: H398-LLD-GWY-FNC-199

The function shall loop from label value is M\_DEC\_ZERO to numOf429in of Cmdownload\_lp\_cmd\_data, loop from channel value is zero to M\_A429\_MAX\_RX\_CHAN and do nothing When,

(((channel of A429in (with index as label value) of Cmdownload\_lp\_cmd\_data right shifted to channel value) Bitwise AND 0x01) not equal to zero) returns FALSE

##### 10.1.10.7.3 daugwya429-Inita429Parameters-LLR-003

Requirement ID: H398-LLD-GWY-FNC-200

The function shall loop from label value is M\_DEC\_ZERO to numOf429in of Cmdownload\_lp\_cmd\_data, loop from channel value is zero to M\_A429\_MAX\_RX\_CHAN and do the following

1. Set status of A429\_aas\_chans for indices (channel value, label value) with MISSING
2. Set status of Rx\_a429 for indices (channel value, label value) with MISSING

### 10.1.11 InitTimmingA429

Low Level Design Details about CSU InitTimmingA429 will follow in the sub sections.

#### 10.1.11.1 Brief Description

The InitTimmingA429 function configures the FSMC and enables the bank 1 of FSMC.

#### 10.1.11.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.1.11.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.1.11.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.1.11.5 Return Value

None

#### 10.1.11.6 Other CSUs called by this CSU

FsmcNorSramInit

FsmcNorSramCmd

#### 10.1.11.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to InitTimmingA429.

##### 10.1.11.7.1 daugwya429-InitTimmingA429-LLR-001

Requirement ID: H398-LLD-GWY-FNC-209

The function shall configure FSMC Configuration for external A429 ICs

23.8 nS from NE1 low to OE or WE low as follows

1. Set the fields of timing sram init structure as below

-fsmc\_address\_setuptime to M\_FOUR

-fsmc\_address\_holdtime to M\_ZERO

-fsmc\_data\_setuptime to M\_TEN

-fsmc\_bus\_turnaround\_duration to M\_FIFTEEN

-fsmc\_clk\_division to M\_ZERO

-fsmc\_data\_latency to M\_ZERO

-fsmc\_access\_mode to M\_FSMC\_ACCESSMODE\_A

b) Set the fields of sram init structure as below

- fsmc\_readwrite\_timing\_struct to reference of timing sram init structure

- fsmc\_write\_timing\_struct to reference of timing sram init structure

##### 10.1.11.7.2 daugwya429-InitTimmingA429-LLR-002

Requirement ID: H398-LLD-GWY-FNC-210

The function shall set fields of sram init structure as follows

-fsmc\_bank to M\_FSMC\_BANK1\_NORSRAM2

-fsmc\_data\_address\_mux to M\_FSMC\_DATAADDRESSMUX\_DISABLE

-fsmc\_memory\_type to M\_FSMC\_MEMORYTYPE\_PSRAM

-fsmc\_memory\_datawidth to M\_FSMC\_MEMORYDATAWIDTH\_16B

-fsmc\_burst\_accessmode to M\_FSMC\_BURSTACCESSMODE\_DISABLE

-fsmc\_asynchronous\_wait to M\_FSMC\_ASYNCHRONOUSWAIT\_DISABLE

-fsmc\_waitsignal\_polarity to M\_FSMC\_WAITSIGNALPOLARITY\_LOW

-fsmc\_wrap\_mode to M\_FSMC\_WRAPMODE\_DISABLE

-fsmc\_waitsignal\_active to M\_FSMC\_WAIT\_SIG\_ACTIVE\_BEF\_WAIT

-fsmc\_write\_operation to M\_FSMC\_WRITEOPERATION\_ENABLE

-fsmc\_waitsignal to M\_FSMC\_WAITSIGNAL\_DISABLE

-fsmc\_extended\_mode to M\_FSMC\_EXTENDEDMODE\_ENABLE

-fsmc\_write\_burst to M\_FSMC\_WRITEBURST\_DISABLE

##### 10.1.11.7.3 daugwya429-InitTimmingA429-LLR-003

Requirement ID: H398-LLD-GWY-FNC-211

The function shall Initialize NORSRAM bank in FSMC by calling 'FsmcNorSramInit' with parameter address of sram init structure.

##### 10.1.11.7.4 daugwya429-InitTimmingA429-LLR-004

Requirement ID: H398-LLD-GWY-FNC-212

The function shall enable first block of bank 1 in FSMC by calling 'FsmcNorSramCmd' with parameters M\_FSMC\_BANK1\_NORSRAM2, ENABLED.

### 10.1.12 A429Init

Low Level Design Details about CSU A429Init will follow in the sub sections.

#### 10.1.12.1 Brief Description

The A429Init function initializes FSMC, A429 chips, task signaling parameters and creates the ARINC 429 task.

#### 10.1.12.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.1.12.3 List of global variables accessed and modified

Accessed : A429\_task\_stk, A429\_out\_task\_stk

Modified : A429\_task\_stk, A429\_out\_task\_stk

#### 10.1.12.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.1.12.5 Return Value

None

#### 10.1.12.6 Other CSUs called by this CSU

InitTimmingA429

InitChannels

TbaseTaskSignaling

RterrError

RterrForever

OsSemCreate

OsTaskCreate

A429OutTask

A429Task

#### 10.1.12.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to A429Init.

##### 10.1.12.7.1 daugwya429-A429Init-LLR-001

Requirement ID: H398-LLD-GWY-FNC-221

The function shall Initialize FSMC timming for ARINC 429 interface by calling function InitTimmingA429

##### 10.1.12.7.2 daugwya429-A429Init-LLR-002

Requirement ID: H398-LLD-GWY-FNC-222

The function shall Initialize transmit and receive channels by calling InitChannels

##### 10.1.12.7.3 daugwya429-A429Init-LLR-003

Requirement ID: H398-LLD-GWY-FNC-223

The function shall Install task signaling parameters by calling the function TbaseTaskSignaling with parameters

a) M\_A429\_TASK\_TICKS

b) Semaphore a429 task value set by return value of function call OsSemCreate with parameter M\_ZERO

##### 10.1.12.7.4 daugwya429-A429Init-LLR-004

Requirement ID: H398-LLD-GWY-FNC-224

The function shall Initialize the stack for run time checks by looping from M\_ZERO to M\_A429\_TASK\_STK\_SIZE and set A429\_task\_stk for all the iteration with M\_CBIT\_TASK\_STK\_VAL

##### 10.1.12.7.5 daugwya429-A429Init-LLR-005

Requirement ID: H398-LLD-GWY-FNC-225

The function shall do the following

1. create the ARINC 429 task by calling function OsTaskCreate with parameters reference of function pointer A429Task, M\_HW\_NULL, address of A429\_task\_stk of size M\_A429\_TASK\_STK\_SIZE and M\_A429\_TASK\_PRIO
2. Call function RterrError with parameters BIT\_ERR, M\_THREE, reference of function pointer RterrForever when return value of OsTaskCreate is not equal to M\_OS\_NO\_ERR, otherwise do nothing

##### 10.1.12.7.6 daugwya429-A429Init-LLR-006

Requirement ID: H398-LLD-GWY-FNC-226

The function shall do the following

1. initialize task signaling parameters by calling the function TbaseTaskSignaling with parameters

- M\_A429\_TASK\_TICKS

- Sem\_a429\_out\_task value set by return value of function call OsSemCreate with parameter M\_ZERO

b) loop from M\_ZERO to M\_A429\_TASK\_STK\_SIZE and set A429\_out\_task\_stk for all the iteration with M\_CBIT\_TASK\_STK\_VAL

##### 10.1.12.7.7 daugwya429-A429Init-LLR-007

Requirement ID: H398-LLD-GWY-FNC-227

The function shall do the following

1. create the ARINC 429 task by calling function OsTaskCreate with parameters reference of function pointer A429OutTask, M\_HW\_NULL, address of A429\_out\_task\_stk of size M\_A429\_OUT\_TASK\_STK\_SIZE and M\_A429\_OUT\_TASK\_PRIO
2. Call function RterrError with parameters BIT\_ERR, M\_THREE, reference of function pointer RterrForever when return value of OsTaskCreate is not equal to M\_OS\_NO\_ERR, otherwise do nothing

### 10.1.13 CalcArincWordChecksum

Low Level Design Details about CSU CalcArincWordChecksum will follow in the sub sections.

#### 10.1.13.1 Brief Description

The CalcArincWordChecksum function calculates the ARINC TX label Checksum.

#### 10.1.13.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.1.13.3 List of global variables accessed and modified

Accessed               : None

Modified                 :  None

#### 10.1.13.4 Parameter list (Input/Output)

Inputs    : T\_SINT32 arinc\_word - to calculate checksum

Outputs : None

#### 10.1.13.5 Return Value

T\_UINT8 - Return the calculated checksum.

#### 10.1.13.6 Other CSUs called by this CSU

None

#### 10.1.13.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to CalcArincWordChecksum.

##### 10.1.13.7.1 daugwya429-CalcArincWordChecksum-LLR-001

Requirement ID: H398-LLD-GWY-FNC-7453

The function shall do the following

1. Set form arinc word to Bitwise AND of (arinc\_word Bitwise AND M\_HEX\_BYTE)
2. Update form arinc word by performing Bitwise OR of form arinc word with ((arinc\_word right shifted by M\_SHIFT\_BY\_11) Bitwise AND with M\_HEX\_THREE) left shifted by M\_SHIFT\_BY\_8.
3. Update form arinc word by performing Bitwise OR of form arinc word with ((arinc\_word right shifted by M\_SHIFT\_BY\_13) Bitwise AND with M\_0X7FFFF) left shifted by M\_SHIFT\_BY\_10.
4. Update form arinc word by performing Bitwise OR of form arinc word with ((arinc\_word right shifted by M\_SHIFT\_BY\_9) Bitwise AND with M\_HEX\_THREE) left shifted by M\_SHIFT\_29.

5. Set Checksum to ((form arinc word right shifted by M\_SHIFT\_28) Bitwise AND with M\_HEX\_SEVEN.

1. Update Checksum by performing addition of Checksum and ((form arinc word right shifted by M\_SHIFT\_24) ) Bitwise AND with M\_HEX\_SEVEN.
2. Update Checksum by performing addition of Checksum and ((form arinc word right shifted by M\_SHIFT\_20) Bitwise AND with M\_HEX\_SEVEN).
3. Update Checksum by performing addition of Checksum and ((form arinc word right shifted by M\_SHIFT\_BY\_16) Bitwise AND with M\_HEX\_SEVEN).
4. Update Checksum by performing addition of Checksum and ((form arinc word right shifted by M\_SHIFT\_12) Bitwise AND with M\_HEX\_SEVEN).
5. Update checksum by performing addition of Checksum and ((form arinc word right shifted by M\_SHIFT\_BY\_8) Bitwise AND with M\_HEX\_SEVEN).
6. Update checksum by performing addition of Checksum and ((form arinc word right shifted by M\_SHIFT\_BY\_4) Bitwise AND with M\_HEX\_SEVEN).
7. Update Checksum by performing addition of Checksum and ((form arinc word right shifted by M\_SHIFT\_BY\_0) Bitwise AND with M\_HEX\_SEVEN).
8. Set Checksum to addition of Negation of Checksum and M\_ONE.
9. Return Checksum.

## 10.2 daugwya825comm

The daugwya825comm CSC contains implementation of A825 communications portion.

### 10.2.1 A8251CommTask

Low Level Design Details about CSU A8251CommTask will follow in the sub sections.

#### 10.2.1.1 Brief Description

The A8251CommTask is signaled from the ARINC 825 receiver routine to process the received message. The task will only handle messages that it is designated to handle. All others will be discarded.

#### 10.2.1.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.2.1.3 List of global variables accessed and modified

Accessed : Sem\_a825\_task

Modified : None

#### 10.2.1.4 Parameter list (Input/Output)

Inputs : void \*pData - data pointer

Outputs : void \*pData - data pointer

#### 10.2.1.5 Return Value

None

#### 10.2.1.6 Other CSUs called by this CSU

OsSemPend

WdogKickWatchDog

A825GetMessage

APPu32NOCEx

AppU32Nsc

#### 10.2.1.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to A8251CommTask

##### 10.2.1.7.1 daugwya825comm-A8251CommTask-LLR-001

Requirement ID: H398-LLD-GWY-FNC-237

The function shall set pData with addition of pData and M\_ZERO

##### 10.2.1.7.2 daugwya825comm-A8251CommTask-LLR-002

Requirement ID: H398-LLD-GWY-FNC-238

The function shall loop forever and do the following:

1. Call OsSemPend with parameters (Sem\_a825\_task,M\_ZERO, reference of error status) to wait for a message to arrive
2. Call WdogKickWatchDog to reload the watchdog counter.

##### 10.2.1.7.3 daugwya825comm-A8251CommTask-LLR-003

Requirement ID: H398-LLD-GWY-FNC-239

The function shall loop forever and do the following:

1. Call A825GetMessage with parameters EEC and reference of message structure.
2. Do nothing when return value of the function A825GetMessage is not equal to A825\_RX\_QUEUE\_EMPTY.

##### 10.2.1.7.4 daugwya825comm-A8251CommTask-LLR-004

Requirement ID: H398-LLD-GWY-FNC-240

The function shall loop forever and do the following:

a) Call A825GetMessage with parameters NOC and reference of message structure.

b) Set function APPu32NOCEx with parameter as reference of message structure when,

- return value of the function A825GetMessage is not equal to A825\_RX\_QUEUE\_EMPTY

- sfid of bid of sid of message structure is equal to M\_HOWELL\_DAU\_FID

c) Do nothing when,

- return value of the function A825GetMessage is not equal to A825\_RX\_QUEUE\_EMPTY

- sfid of bid of sid of message structure is not equal to M\_HOWELL\_DAU\_FID

##### 10.2.1.7.5 daugwya825comm-A8251CommTask-LLR-005

Requirement ID: H398-LLD-GWY-FNC-241

The function shall loop forever and do the following:

a) Call A825GetMessage with parameters NSC and reference of message structure.

b) Set function AppU32Nsc with parameter as reference of message structure when,

- return value of the function A825GetMessage is not equal to A825\_RX\_QUEUE\_EMPTY

- sid of pid of sid of message structure is less than or equal to CONFIG\_SID

c) Do nothing when,

- return value of the function A825GetMessage is not equal to A825\_RX\_QUEUE\_EMPTY

- sid of pid of sid of message structure is greater than CONFIG\_SID

##### 10.2.1.7.6 daugwya825comm-A8251CommTask-LLR-006

Requirement ID: H398-LLD-GWY-FNC-242

The function shall loop forever and do the following:

1. Call A825GetMessage with parameters TMC and reference of message structure.
2. Do nothing when,

- return value of the function A825GetMessage is not equal to A825\_RX\_QUEUE\_EMPTY

### 10.2.2 A825CommInit

Low Level Design Details about CSU A825CommInit will follow in the sub sections.

#### 10.2.2.1 Brief Description

The A825CommInit initializes stack and creates Task to implement A825 communication .

#### 10.2.2.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.2.2.3 List of global variables accessed and modified

Accessed : Sem\_a825\_task, A825comm\_task\_stk

Modified : A825comm\_task\_stk

#### 10.2.2.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.2.2.5 Return Value

None

#### 10.2.2.6 Other CSUs called by this CSU

TbaseTaskSignaling

OsSemCreate

OsTaskCreate

A8251CommTask

RterrError

RterrForever

#### 10.2.2.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to A825CommInit

##### 10.2.2.7.1 daugwya825comm-A825CommInit-LLR-001

Requirement ID: H398-LLD-GWY-FNC-251

The function shall do the following:

1. Calculate Sem\_a825\_task by calling OsSemCreate with parameter M\_ZERO
2. Call TbaseTaskSignaling with parameters M\_A825\_TASK\_TICKS and Sem\_a825\_task

##### 10.2.2.7.2 daugwya825comm-A825CommInit-LLR-002

Requirement ID: H398-LLD-GWY-FNC-252

The function shall loop from M\_ZERO until M\_A825\_TASK\_STK\_SIZE and set A825comm\_task\_stk for all iterations with M\_CBIT\_TASK\_STK\_VAL.

##### 10.2.2.7.3 daugwya825comm-A825CommInit-LLR-003

Requirement ID: H398-LLD-GWY-FNC-253

The function shall do the following:

1. Call OsTaskCreate with parameters (reference of function pointer A8251CommTask, M\_HW\_NULL, reference of A825comm\_task\_stk with index M\_A825\_TASK\_STK\_SIZE, M\_A825\_TASK\_PRIO)
2. Call RterrError with parameters (BIT\_ERR,M\_THREE, reference of function pointer RterrForever) when return value of the function OsTaskCreate is not equal to M\_OS\_NO\_ERR
3. Do nothing when return value of the function OsTaskCreate is equal to M\_OS\_NO\_ERR

## 10.3 daugwya825comm2

The daugwya825comm2 CSC contains implementation of A825 communication portion.

### 10.3.1 A8251CommTask2

Low Level Design Details about CSU A8251CommTask2 will follow in the sub sections.

#### 10.3.1.1 Brief Description

The A8251CommTask2 is signaled from the ARINC 8252 receiver routine to process the received message. The task will only handle messages that it is desinged to handle. All others will be discarded.

#### 10.3.1.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.3.1.3 List of global variables accessed and modified

Accessed : Sem\_a825\_task2

Modified : None

#### 10.3.1.4 Parameter list (Input/Output)

Inputs : void \*pData– pointer data.

Outputs : void \*pData- pointer data.

#### 10.3.1.5 Return Value

None.

#### 10.3.1.6 Other CSUs called by this CSU

OsSemPend

WdogKickWatchDog

A825GetMessage2

AppU32Noc

AppU32Nsc

#### 10.3.1.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to A8251CommTask2

##### 10.3.1.7.1 daugwya825comm2-A8251CommTask2-LLR-001

Requirement ID: H398-LLD-GWY-FNC-263

The function shall perform as follows:

1. Set pData with addition of pData and M\_ZERO

2. loop infinitely and perform the following operations within loop

a. Pend on Semaphore by calling OsSemPend with the following parameters (Sem\_a825\_task2, pend timeout values as M\_ZERO, Address of error code)

b. Reload watchdog counter by calling function 'WdogKickWatchDog'.

##### 10.3.1.7.2 daugwya825comm2-A8251CommTask2-LLR-002

Requirement ID: H398-LLD-GWY-FNC-264

The function shall loop infinitely and perform the following operations within loop

when the return value of A825GetMessage2 function with parameters ( EEC and reference to A825 message structure) is other than A825\_RX\_QUEUE\_EMPTY

1. Do nothing.

##### 10.3.1.7.3 daugwya825comm2-A8251CommTask2-LLR-003

Requirement ID: H398-LLD-GWY-FNC-265

The function shall loop infinitely and perform the following operations within loop

when the return value of A825GetMessage2 function with parameters ( NOC and reference to A825 message structure) is other than A825\_RX\_QUEUE\_EMPTY

a) set function AppU32Noc with parameter as (reference of A825 message structure) when sfid of bid of sid of A825 message structure is M\_HOWELL\_DAU\_FID. Otherwise

do nothing.

##### 10.3.1.7.4 daugwya825comm2-A8251CommTask2-LLR-004

Requirement ID: H398-LLD-GWY-FNC-266

The function shall loop infinitely and perform the following operations within loop

when the return value of A825GetMessage2 function with parameters ( NSC and reference to A825 message structure) is other than A825\_RX\_QUEUE\_EMPTY

a) Set function AppU32Nsc with parameter as (reference of A825 message structure) when sid of pid of sid of A825 message structure is less than or equal to CONFIG\_SID. Otherwise do nothing.

##### 10.3.1.7.5 daugwya825comm2-A8251CommTask2-LLR-005

Requirement ID: H398-LLD-GWY-FNC-267

The function shall loop infinitely and perform the following operations within loop

when the return value of A825GetMessage2 function with parameters ( TMC and reference to A825 message structure) is other than A825\_RX\_QUEUE\_EMPTY

a) do nothing.

### 10.3.2 A825CommInit2

Low Level Design Details about CSU A825CommInit2 will follow in the sub sections.

#### 10.3.2.1 Brief Description

The A825CommInit2 function initializes stack and creates Task to implement A8252 communication.

#### 10.3.2.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-12-GWY).

#### 10.3.2.3 List of global variables accessed and modified

Accessed : Sem\_a825\_task2, A825comm\_task\_stk2

Modified : A825comm\_task\_stk2

#### 10.3.2.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.3.2.5 Return Value

None.

#### 10.3.2.6 Other CSUs called by this CSU

TbaseTaskSignaling

OsSemCreate

OsTaskCreate

RterrError

A8251CommTask2

RterrForever

#### 10.3.2.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to A825CommInit2.

##### 10.3.2.7.1 daugwya825comm2-A825CommInit2-LLR-001

Requirement ID: H398-LLD-GWY-FNC-276

The function shall install task signalling parameters by calling 'TbaseTaskSignaling' with the parameters

a) Task Ticks M\_A8252\_TASK\_TICKS and

b) return value of function OsSemCreate with parameter M\_ZERO is set to Sem\_a825\_task2

##### 10.3.2.7.2 daugwya825comm2-A825CommInit2-LLR-002

Requirement ID: H398-LLD-GWY-FNC-277

The function shall loop from M\_ZERO to M\_A8252\_TASK\_STK\_SIZE and set A825comm\_task\_stk2 for all iterations with M\_CBIT\_TASK\_STK\_VAL.

##### 10.3.2.7.3 daugwya825comm2-A825CommInit2-LLR-003

Requirement ID: H398-LLD-GWY-FNC-278

The function shall do the following:

a)Call OsTaskCreate with parameters (return of function A8251CommTask2, M\_HW\_NULL, reference of A825comm\_task\_stk2 with index M\_A8252\_TASK\_STK\_SIZE, M\_A8252\_TASK\_PRIO)

b)Call RterrError with parameters (BIT\_ERR,M\_THREE, RterrForever) when return value of the function OsTaskCreate is not equal to M\_OS\_NO\_ERR

c) Do nothing when return value of the function OsTaskCreate is equal to M\_OS\_NO\_ERR

## 10.4 daugwyapp

The daugwyapp CSC defines the application task code. It defines the indicators gathering, processing, and distribution of parameters.

### 10.4.1 DoMirrorCheck

Low Level Design Details about CSU DoMirrorcheck will follow in the sub sections.

#### 10.4.1.1 Brief Description

The DoMirrorCheck function checks if data in memory for Configuration is good.

#### 10.4.1.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.4.1.3 List of global variables accessed and modified

Accessed : None

Modified : Cm\_data

#### 10.4.1.4 Parameter list (Input/Output)

Inputs : T\_UINT32 \* start\_addr - starting address of memory location for calculation of CRC.

Outputs : None

#### 10.4.1.5 Return Value

None

#### 10.4.1.6 Other CSUs called by this CSU

CrcResetDr

CrcCalcBlockCrc

#### 10.4.1.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to DoMirrorCheck.

##### 10.4.1.7.1 daugwyapp-DoMirrorCheck-LLR-001

Requirement ID: H398-LLD-GWY-FNC-288

The function shall reset the CRC control register by calling function ‘CrcResetDr’.

##### 10.4.1.7.2 daugwyapp-DoMirrorCheck-LLR-005

Requirement ID: H398-LLD-GWY-FNC-292

The function shall calculate check sum by calling ‘CrcCalcBlockCrc’ with parameters M\_MEMMAP\_CMU\_ADDR and M\_MEMMAP\_CMU\_CRC\_CNT when start\_addr is equal to M\_MEMMAP\_CMU\_ADDR.

##### 10.4.1.7.3 daugwyapp-DoMirrorCheck-LLR-006

Requirement ID: H398-LLD-GWY-FNC-293

The function shall set cmu\_ok of Cm\_data to FALSE when calculated checksum is not equal to actual CRC at the address M\_MEMMAP\_CMU\_CRC\_ADDR and start\_addr is equal to M\_MEMMAP\_CMU\_ADDR.

##### 10.4.1.7.4 daugwyapp-DoMirrorCheck-LLR-007

Requirement ID: H398-LLD-GWY-FNC-294

The function shall set cmu\_ok of Cm\_data to TRUE when calculated checksum is equal to actual CRC at the address M\_MEMMAP\_CMU\_CRC\_ADDR and start\_addr is equal to M\_MEMMAP\_CMU\_ADDR.

##### 10.4.1.7.5 daugwyapp-DoMirrorCheck-LLR-009

Requirement ID: H398-LLD-GWY-FNC-296

The function shall do nothing when start\_addr is other than M\_MEMMAP\_MIRROR\_DEF\_START, M\_MEMMAP\_CMU\_ADDR, M\_MEMMAP\_MIRROR\_ADDR.

### 10.4.2 AppInit

Low Level Design Details about CSU AppInit will follow in the sub sections.

#### 10.4.2.1 Brief Description

The AppInit function installs application task semaphore into timebase and creates the application OS task.

#### 10.4.2.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.4.2.3 List of global variables accessed and modified

Accessed : L\_app\_task\_stk

Modified : L\_app\_task\_stk

#### 10.4.2.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.4.2.5 Return Value

None

#### 10.4.2.6 Other CSUs called by this CSU

TbaseTaskSignaling

OsTaskCreate

OsSemCreate

AppTask

RterrError

RterrForever

#### 10.4.2.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to AppInit.

##### 10.4.2.7.1 daugwyapp-AppInit-LLR-001

Requirement ID: H398-LLD-GWY-FNC-305

The function shall install task signaling parameters for the application task by calling 'TbaseTaskSignaling' with parameters

a) M\_APP\_TASK\_TICKS as task ticks

b) Return value of function ‘OsSemCreate’ with parameter M\_ZERO.

##### 10.4.2.7.2 daugwyapp-AppInit-LLR-002

Requirement ID: H398-LLD-GWY-FNC-306

The function shall initialize L\_app\_task\_stk with M\_CBIT\_TASK\_STK\_VAL by looping through M\_ZERO to M\_APP\_TASK\_STK\_SIZE.

##### 10.4.2.7.3 daugwyapp-AppInit-LLR-003

Requirement ID: H398-LLD-GWY-FNC-307

The function shall create the application Task by calling 'OsTaskCreate' with parameters

a) pointer to function ‘AppTask’,

b) M\_HW\_NULL as task entry point,

c) Reference to top of stack of application task stack (L\_app\_task\_stk with index M\_APP\_TASK\_STK\_SIZE)

d) M\_APP\_TASK\_PRIO as task priority

and call the function ‘RterrError’ with parameters (BIT\_ERR,M\_THREE and reference to function RterrForever) when the function ‘OsTaskCreate’ returns other than M\_OS\_NO\_ERR, otherwise do nothing.

### 10.4.3 AppU32Noc

Low Level Design Details about CSU AppU32Noc will follow in the sub sections.

#### 10.4.3.1 Brief Description

The function AppU32Noc responds to the broadcast message on Normal Operation Channel. It transmits received ARINC825 message to the buffer for processing.

#### 10.4.3.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.4.3.3 List of global variables accessed and modified

Accessed : Xram\_data\_buffer

App\_discreteapp

App\_discreteappcrc

App\_discreteconfig

App\_discreteconfigcrc

App\_analogapp

App\_analogappcrc

App\_analogconfig

App\_analogconfigcrc

Dauenginedic

Logic\_currenttime

Logic\_recdlu

App\_discreteboot

App\_discretebootcrc

App\_discretebootconfig

App\_discreteswl

App\_discreteswlcrc

App\_analogboot

App\_analogbootcrc

App\_analogbootconfig

App\_analogbootconfigcrc

App\_analogswl

App\_analogswlcrc

App\_analogcalib

App\_analogcalibcrc

Modified : Xram\_data\_buffer

App\_discreteapp

App\_discreteconfig

App\_analogapp

App\_analogconfig

App\_discreteboot

App\_discretebootconfig

App\_discreteswl

App\_analogboot

App\_analogbootconfig

App\_analogswl

App\_analogcalib

#### 10.4.3.4 Parameter list (Input/Output)

Inputs : T\_A825\_MSG \*sMess - Pointer to the A825 message

Outputs : None

#### 10.4.3.5 Return Value

T\_UINT32 - returns 0 if success

returns 1 on failure

#### 10.4.3.6 Other CSUs called by this CSU

HwCopy

GpioReadInputDataBit

#### 10.4.3.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to AppU32Noc.

##### 10.4.3.7.1 daugwyapp-AppU32Noc-LLR-001

Requirement ID: H398-LLD-GWY-FNC-316

The function shall perform as follows:

1. Set Discrete data counter to M\_ZERO.

2. Perform as follows when ((doc of bid of sid of sMess is greater than or equal to DISCI) AND (doc of bid of sid of sMess is lesser than CHIP\_SELECT) returns TRUE:

1. Loop through M\_ZERO to M\_EIGHT(1st loop counter) and set 2nd loop counter to ((doc of bid of sid of sMess subtracted by DISCI) multiplied by M\_SIXTY\_FOUR) and perform the following

* Loop through M\_ZERO to M\_EIGHT (3rd loop counter) and increment 2nd loop counter and set temp disci with index as 2nd loop counter of App\_A825 of Xram\_data\_buffer to (u8\_payload with index as 1st loop counter of sMess bit shifted to right by 3rd loop counter bitwise AND with M\_ONE)

b) Set Discrete\_dat\_received to TRUE if 2nd loop counter is greater than or equal to M\_NINETY\_FIVE.

##### 10.4.3.7.2 daugwyapp-AppU32Noc-LLR-002

Requirement ID: H398-LLD-GWY-FNC-317

The function shall perform as follows:

1. Set Discrete data counter to M\_ZERO.

2. Perform as follows when ((doc of bid of sid of sMess is greater than or equal to CHIP\_SELECT) AND (doc of bid of sid of sMess is lesser than DISCO) returns TRUE:

1. Loop through M\_ZERO to M\_EIGHT(1st loop counter) and set 2nd loop counter to ((doc of bid of sid of sMess subtracted by CHIP\_SELECT) multiplied by M\_SIXTY\_FOUR) and perform the following

* Loop through M\_ZERO to M\_EIGHT (3rd loop counter) and increment 2nd loop counter and set chip\_detect with index as 2nd loop counter of App\_A825 of Xram\_data\_buffer to (u8\_payload with index as 1st loop counter of sMess bit shifted to right by 3rd loop counter bitwise AND with M\_ONE).

##### 10.4.3.7.3 daugwyapp-AppU32Noc-LLR-003

Requirement ID: H398-LLD-GWY-FNC-318

The function shall perform as follows:

1. Set Discrete data counter to M\_ZERO.

2. Perform as follows when ((doc of bid of sid of sMess is greater than or equal to DISCO) AND (doc of bid of sid of sMess is lesser than DISCO\_STAT) returns TRUE:

1. Loop through M\_ZERO to M\_EIGHT(1st loop counter), set 2nd loop counter to ((doc of bid of sid of sMess subtracted by DISCO) multiplied by M\_THIRTY\_TWO), Loop through M\_ZERO to M\_EIGHT (3rd loop counter), increment 2nd loop counter and perform the following:
2. Set disco with index as 2nd loop counter of App\_A825 of Xram\_data\_buffer to (u8\_payload with index as 1st loop counter of sMess bit shifted to right by 3rd loop counter bitwise AND with M\_THREE)
3. Increment 3rd loop counter by one.

##### 10.4.3.7.4 daugwyapp-AppU32Noc-LLR-004

Requirement ID: H398-LLD-GWY-FNC-320

The function shall perform as follows:

1. Set Analog data counter to M\_ZERO.

2. Perform as follows when ((doc of bid of sid of sMess is greater than or equal to ALOG) AND (doc of bid of sid of sMess is lesser than ALOG\_STATS) returns TRUE:

1. Loop through M\_ZERO to M\_EIGHT(1st loop counter) and set 2nd loop counter to ((doc of bid of sid of sMess subtracted by ALOG) multiplied by M\_FOUR), increment 2nd loop counter and perform the following

* Do the following when 2nd loop counter is less than or equal to M\_SEVENTY\_FIVE:

1. Call ‘HwCopy’ with parameters reference to analog with index as 2nd loop counter of App\_A825 of Xram\_data\_buffer, reference to u8\_payload with index as 1st loop counter of sMess and M\_TWO.
2. Set count of analmiss with index as 2nd loop counter of App\_A825 of Xram\_data\_buffer to M\_ZERO.
3. Set state of analmiss with index as 2nd loop counter App\_A825 of Xram\_data\_buffer to TRUE.

* Increment 1st loop counter by one.

b) Set Analog\_dat\_received to TRUE.

##### 10.4.3.7.5 daugwyapp-AppU32Noc-LLR-005

Requirement ID: H398-LLD-GWY-FNC-321

The function shall do the following

. 1) Set Analog data counter to M\_ZERO.

2) when ((doc of bid of sid of sMess is greater than or equal to ALOG\_STATS) AND (doc of bid of sid of sMess is lesser than RS422\_MESSAGES) returns TRUE:

a) Loop through M\_ZERO to M\_EIGHT(1st loop counter) and set 2nd loop counter to ((doc of bid of sid of sMess subtracted by ALOG\_STATS) multiplied by M\_THIRTY\_TWO)

⦁ Loop through M\_ZERO to M\_EIGHT (3rd loop counter) and increment 2nd loop counter.

⦁ Do the following when 2nd loop counter is less than or equal to M\_SEVENTY\_FIVE:

i. Set analStat with index as 2nd loop counter of App\_A825 of Xram\_data\_buffer to u8\_payload of sMess with index as 1st loop counter bit shifted to right by 3rd loop counter bitwise AND with M\_THREE.

⦁ Increment 3rd loop counter.

##### 10.4.3.7.6 daugwyapp-AppU32Noc-LLR-006

Requirement ID: H398-LLD-GWY-FNC-322

The function shall perform as follows:

- Set discrete\_output\_fail of Mis\_cmp\_fail to (u8\_payload with index as M\_ZERO of sMess AND with M\_LSB\_FIRST\_BIT\_MASK) when (doc of bid of sid of sMess is equal to DISC\_28V\_OUT\_FAIL)

##### 10.4.3.7.7 daugwyapp-AppU32Noc-LLR-011

Requirement ID: H398-LLD-GWY-FNC-328

The function shall do nothing when ((doc of bid of sid of sMess is greater than or equal to OTHER\_MESSAGES) AND (doc of bid of sid of sMess is lesser than ACQUIRE\_CRC\_PN) returns TRUE.

##### 10.4.3.7.8 daugwyapp-AppU32Noc-LLR-012

Requirement ID: H398-LLD-GWY-FNC-329

The function shall do nothing when ((doc of bid of sid of sMess is greater than or equal to ACQUIRE\_CRC\_PN) AND (doc of bid of sid of sMess is lesser than PN\_MESSAGES) returns TRUE.

##### 10.4.3.7.9 daugwyapp-AppU32Noc-LLR-013

Requirement ID: H398-LLD-GWY-FNC-330

The function shall do the following when ((doc of bid of sid of sMess is greater than or equal to PN\_MESSAGES) AND (doc of bid of sid of sMess is lesser than MAX\_EDOC) returns TRUE:

* Call HwCopy with parameters (App\_discreteboot, u8\_payload of sMess, M\_EIGHT) when doc of bid of sid of sMess is M\_DOC\_DISCRETE\_BOOTLOADER\_VAL1.
* Call HwCopy with parameters (reference of App\_discreteboot with index M\_EIGHT, u8\_payload of sMess, M\_SEVEN) and set App\_discreteboot of index M\_DEC\_FIFTEEN to M\_ZERO when doc of bid of sid of sMess is M\_DOC\_DISCRETE\_BOOTLOADER\_VAL2.
* Call HwCopy with parameters (reference of App\_discretebootcrc, u8\_payload of sMess, M\_FOUR) when doc of bid of sid of sMess is M\_DOC\_DISCRETE\_BOOTLOADER\_VAL3.

##### 10.4.3.7.10 daugwyapp-AppU32Noc-LLR-018

Requirement ID: H398-LLD-GWY-FNC-6223

The function shall do the following when ((doc of bid of sid of sMess is greater than or equal to PN\_MESSAGES) AND (doc of bid of sid of sMess is lesser than MAX\_EDOC) returns TRUE:

* doc of bid of sid of sMess is M\_DOC\_DISCRETE\_DRI\_APP\_VAL1.

Call ‘HwCopy’ with parameters App\_discreteapp, u8\_payload of sMess and M\_EIGHT.

* Do the following when doc of bid of sid of sMess is M\_DOC\_DISCRETE\_DRI\_APP\_VAL2.

Call ‘HwCopy’ with parameters reference to App\_discreteapp with index M\_EIGHT, u8\_payload of sMess and M\_SEVEN.

Set App\_discreteapp with index M\_DEC\_FIFTEEN to M\_ZERO.

* Do the following when doc of bid of sid of sMess is M\_DOC\_DISCRETE\_DRI\_APP\_VAL3.

Call ‘HwCopy’ with parameters reference to App\_discreteappcrc, u8\_payload of sMess and M\_FOUR

##### 10.4.3.7.11 daugwyapp-AppU32Noc-LLR-019

Requirement ID: H398-LLD-GWY-FNC-6224

The function shall do the following when ((doc of bid of sid of sMess is greater than or equal to PN\_MESSAGES) AND (doc of bid of sid of sMess is lesser than MAX\_EDOC) returns TRUE:

* Do the following when doc of bid of sid of sMess is M\_DOC\_DISCRETE\_DRI\_CFG\_VAL1.

Call ‘HwCopy’ with parameters App\_discreteconfig, u8\_payload of sMess and M\_EIGHT.

* Do the following when doc of bid of sid of sMess is M\_DOC\_DISCRETE\_DRI\_CFG\_VAL2.

Call ‘HwCopy’ with parameters reference to App\_discreteconfig with index M\_EIGHT, u8\_payload of sMess and M\_SEVEN.

Set App\_discreteconfig with index M\_DEC\_FIFTEEN to M\_ZERO.

* Do the following when doc of bid of sid of sMess is M\_DOC\_DISCRETE\_DRI\_CFG\_VAL3.

Call ‘HwCopy’ with parameters reference to App\_discreteconfigcrc, u8\_payload of sMess and M\_FOUR.

##### 10.4.3.7.12 daugwyapp-AppU32Noc-LLR-020

Requirement ID: H398-LLD-GWY-FNC-6225

The function shall do the following when ((doc of bid of sid of sMess is greater than or equal to PN\_MESSAGES) AND (doc of bid of sid of sMess is lesser than MAX\_EDOC) returns TRUE:

* Do the following when doc of bid of sid of sMess is M\_DOC\_DISCRETE\_BOOT\_CFG\_VAL1.

Call HwCopy with parameters (App\_discretebootconfig, u8\_payload of sMess, M\_EIGHT).

* Do the following when doc of bid of sid of sMess is M\_DOC\_DISCRETE\_BOOT\_CFG\_VAL2.

Call ‘HwCopy’ with parameters reference to App\_discretebootconfig with index M\_EIGHT, u8\_payload of sMess and M\_SEVEN.

set App\_discretebootconfig of index M\_DEC\_FIFTEEN to M\_ZERO

* Call HwCopy with parameters (reference of App\_discretebootconfigcrc, u8\_payload of sMess, M\_FOUR) when doc of bid of sid of sMess is M\_DOC\_DISCRETE\_BOOT\_CFG\_VAL3.

##### 10.4.3.7.13 daugwyapp-AppU32Noc-LLR-021

Requirement ID: H398-LLD-GWY-FNC-6226

The function shall do the following when ((doc of bid of sid of sMess is greater than or equal to PN\_MESSAGES) AND (doc of bid of sid of sMess is lesser than MAX\_EDOC) returns TRUE:

* Call HwCopy with parameters (App\_discreteswl, u8\_payload of sMess, M\_EIGHT) when doc of bid of sid of sMess is M\_DOC\_DISCRETE\_SOFT\_LOAD\_VAL1.
* Call HwCopy with parameters (reference of App\_discreteswl with index M\_EIGHT, u8\_payload of sMess, M\_SEVEN) and set App\_discreteswl of index M\_DEC\_FIFTEEN to M\_ZERO when doc of bid of sid of sMess is M\_DOC\_DISCRETE\_SOFT\_LOAD\_VAL2.
* Call HwCopy with parameters (reference of App\_discreteswlcrc, u8\_payload of sMess, M\_FOUR) when doc of bid of sid of sMess is M\_DOC\_DISCRETE\_SOFT\_LOAD\_VAL3.

##### 10.4.3.7.14 daugwyapp-AppU32Noc-LLR-022

Requirement ID: H398-LLD-GWY-FNC-6227

The function shall do the following when ((doc of bid of sid of sMess is greater than or equal to PN\_MESSAGES) AND (doc of bid of sid of sMess is lesser than MAX\_EDOC) returns TRUE:

* Call HwCopy with parameters (App\_analogboot, u8\_payload of sMess, M\_EIGHT) when doc of bid of sid of sMess is M\_DOC\_ANALOG\_BOOTLOADER\_VAL1.
* Call HwCopy with parameters (reference of App\_analogboot with index M\_EIGHT, u8\_payload of sMess, M\_SEVEN) and set App\_analogboot of index M\_DEC\_FIFTEEN to M\_ZERO when doc of bid of sid of sMess is M\_DOC\_ANALOG\_BOOTLOADER\_VAL2.
* Call HwCopy with parameters (reference of App\_analogbootcrc, u8\_payload of sMess, M\_FOUR) when doc of bid of sid of sMess is M\_DOC\_ANALOG\_BOOTLOADER\_VAL3.

##### 10.4.3.7.15 daugwyapp-AppU32Noc-LLR-023

Requirement ID: H398-LLD-GWY-FNC-6228

The function shall do the following when ((doc of bid of sid of sMess is greater than or equal to PN\_MESSAGES) AND (doc of bid of sid of sMess is lesser than MAX\_EDOC) returns TRUE:

* Do the following when doc of bid of sid of sMess is M\_DOC\_ANALOG\_DRV\_APP\_VAL1.

Call ‘HwCopy’ with parameters App\_analogapp, u8\_payload of sMess and M\_EIGHT.

* Do the following when doc of bid of sid of sMess is M\_DOC\_ANALOG\_DRV\_APP\_VAL2.

Call ‘HwCopy’ with parameters reference to App\_analogapp with index M\_EIGHT, u8\_payload of sMess and M\_SEVEN.

Set App\_analogapp with index M\_DEC\_FIFTEEN to M\_ZERO.

* Do the following when doc of bid of sid of sMess is M\_DOC\_ANALOG\_DRV\_APP\_VAL3.

Call ‘HwCopy’ with parameters reference to App\_analogappcrc, u8\_payload of sMess and M\_FOUR.

##### 10.4.3.7.16 daugwyapp-AppU32Noc-LLR-024

Requirement ID: H398-LLD-GWY-FNC-6229

The function shall do the following when ((doc of bid of sid of sMess is greater than or equal to PN\_MESSAGES) AND (doc of bid of sid of sMess is lesser than MAX\_EDOC) returns TRUE:

⦁ Do the following when doc of bid of sid of sMess is M\_DOC\_ANALOG\_DRV\_CFG\_VAL1.

Call ‘HwCopy’ with parameters App\_analogconfig, u8\_payload of sMess and M\_EIGHT.

⦁ Do the following when doc of bid of sid of sMess is M\_DOC\_ANALOG\_DRV\_CFG\_VAL2.

Call ‘HwCopy’ with parameters reference to App\_analogconfig with index M\_EIGHT, u8\_payload of sMess and M\_SEVEN.

Set App\_analogconfig with index M\_DEC\_FIFTEEN to M\_ZERO.

⦁ Do the following when doc of bid of sid of sMess is M\_DOC\_ANALOG\_DRV\_CFG\_VAL3.

Call ‘HwCopy’ with parameters reference to App\_analogconfigcrc, u8\_payload of sMess and M\_FOUR.

##### 10.4.3.7.17 daugwyapp-AppU32Noc-LLR-025

Requirement ID: H398-LLD-GWY-FNC-6230

The function shall do the following when ((doc of bid of sid of sMess is greater than or equal to PN\_MESSAGES) AND (doc of bid of sid of sMess is lesser than MAX\_EDOC) returns TRUE:

* Call HwCopy with parameters (App\_analogswl, u8\_payload of sMess, M\_EIGHT) when doc of bid of sid of sMess is M\_DOC\_ANALOG\_SOFT\_LOAD\_VAL1.
* Call HwCopy with parameters (reference of App\_analogswl with index M\_EIGHT, u8\_payload of sMess, M\_SEVEN) and set App\_analogswl of index M\_DEC\_FIFTEEN to M\_ZERO when doc of bid of sid of sMess is M\_DOC\_ANALOG\_SOFT\_LOAD\_VAL2 .
* Call HwCopy with parameters (reference of App\_analogswlcrc, u8\_payload of sMess, M\_FOUR) when doc of bid of sid of sMess is M\_DOC\_ANALOG\_SOFT\_LOAD\_VAL3.

##### 10.4.3.7.18 daugwyapp-AppU32Noc-LLR-026

Requirement ID: H398-LLD-GWY-FNC-6231

The function shall do the following when ((doc of bid of sid of sMess is greater than or equal to PN\_MESSAGES) AND (doc of bid of sid of sMess is lesser than MAX\_EDOC) returns TRUE:

* Call HwCopy with parameters (App\_analogcalib, u8\_payload of sMess, M\_EIGHT) when doc of bid of sid of sMess is M\_DOC\_ANALOG\_SECOND\_APP\_VAL1.
* Call HwCopy with parameters (reference of App\_analogcalib with index M\_EIGHT, u8\_payload of sMess, M\_SEVEN) and set App\_analogcalib of index M\_DEC\_FIFTEEN to M\_ZERO when doc of bid of sid of sMess is M\_DOC\_ANALOG\_SECOND\_APP\_VAL2.
* Call HwCopy with parameters (reference of App\_analogcalibcrc, u8\_payload of sMess, M\_FOUR) when doc of bid of sid of sMess is M\_DOC\_ANALOG\_SECOND\_APP\_VAL3.

##### 10.4.3.7.19 daugwyapp-AppU32Noc-LLR-027

Requirement ID: H398-LLD-GWY-FNC-6232

The function shall do the following when ((doc of bid of sid of sMess is greater than or equal to PN\_MESSAGES) AND (doc of bid of sid of sMess is lesser than MAX\_EDOC) returns TRUE:

* Do nothing when doc of bid of sid of sMess is other than

M\_DOC\_DISCRETE\_BOOTLOADER\_VAL1,M\_DOC\_DISCRETE\_BOOTLOADER\_VAL2,M\_DOC\_DISCRETE\_BOOTLOADER\_VAL3,M\_DOC\_DISCRETE\_DRI\_APP\_VAL1,M\_DOC\_DISCRETE\_DRI\_APP\_VAL2,M\_DOC\_DISCRETE\_DRI\_APP\_VAL3,M\_DOC\_DISCRETE\_DRI\_CFG\_VAL1,M\_DOC\_DISCRETE\_DRI\_CFG\_VAL2,M\_DOC\_DISCRETE\_DRI\_CFG\_VAL3,M\_DOC\_DISCRETE\_BOOT\_CFG\_VAL1,M\_DOC\_DISCRETE\_BOOT\_CFG\_VAL2,M\_DOC\_DISCRETE\_BOOT\_CFG\_VAL3,M\_DOC\_DISCRETE\_SOFT\_LOAD\_VAL1,M\_DOC\_DISCRETE\_SOFT\_LOAD\_VAL2,M\_DOC\_DISCRETE\_SOFT\_LOAD\_VAL3, M\_DOC\_ANALOG\_BOOTLOADER\_VAL1,M\_DOC\_ANALOG\_BOOTLOADER\_VAL2,M\_DOC\_ANALOG\_BOOTLOADER\_VAL3,M\_DOC\_ANALOG\_DRV\_APP\_VAL1,M\_DOC\_ANALOG\_DRV\_APP\_VAL2,M\_DOC\_ANALOG\_DRV\_APP\_VAL3,M\_DOC\_ANALOG\_DRV\_CFG\_VAL1,M\_DOC\_ANALOG\_DRV\_CFG\_VAL2,M\_DOC\_ANALOG\_DRV\_CFG\_VAL3,M\_DOC\_ANALOG\_BOOT\_CFG\_VAL1,M\_DOC\_ANALOG\_BOOT\_CFG\_VAL2,M\_DOC\_ANALOG\_BOOT\_CFG\_VAL3,M\_DOC\_ANALOG\_SOFT\_LOAD\_VAL1,M\_DOC\_ANALOG\_SOFT\_LOAD\_VAL2,M\_DOC\_ANALOG\_SOFT\_LOAD\_VAL3,M\_DOC\_ANALOG\_SECOND\_APP\_VAL1,M\_DOC\_ANALOG\_SECOND\_APP\_VAL2 and M\_DOC\_ANALOG\_SECOND\_APP\_VAL3.

##### 10.4.3.7.20 daugwyapp-AppU32Noc-LLR-017

Requirement ID: H398-LLD-GWY-FNC-334

The function shall return M\_ZERO.

### 10.4.4 AppU32Nsc

Low Level Design Details about CSU AppU32Nsc will follow in the sub sections.

#### 10.4.4.1 Brief Description

The function AppU32Nsc responds to the Peer to Peer message on Node Service Channel and transmits received ARINC825 message to the buffer for processing.

#### 10.4.4.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.4.4.3 List of global variables accessed and modified

Accessed : Cm\_mode

Download\_size

Xram\_cmu\_buffer

Cm\_data

Modified : Cm\_mode

Download\_size

Cm\_init

Cm\_data

Bit\_mode

App\_transmitoff

#### 10.4.4.4 Parameter list (Input/Output)

Inputs : T\_A825\_MSG \*sMess - Pointer to the ARINC825 message.

Outputs : None

#### 10.4.4.5 Return Value

T\_UINT32 -returns 0 if success

returns 1 on failure

#### 10.4.4.6 Other CSUs called by this CSU

HwCopy

OsTimeDly

SendLog

TransmitMessage

ClearLog

#### 10.4.4.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to AppU32Nsc

##### 10.4.4.7.1 daugwyapp-AppU32Nsc-LLR-001

Requirement ID: H398-LLD-GWY-FNC-343

The function shall do the following:

1. Set identifier to sid of pid of sid of sMess
2. Set Download\_size to (u8\_payload of sMess with index M\_FOUR added to (u8\_payload of sMess with index M\_FIVE bit shifted to left by M\_SHIFT\_BY\_8) added to (u8\_payload of sMess with index M\_SIX bit shifted to left by M\_SIXTEEN) added to (u8\_payload of sMess with index M\_SEVEN bit shifted to left by M\_SHIFT\_24)) and Cm\_mode to CM\_TRACK when identifier is equal to CONFIG\_SID AND Cm\_mode is equal to CM\_SETUP.

##### 10.4.4.7.2 daugwyapp-AppU32Nsc-LLR-002

Requirement ID: H398-LLD-GWY-FNC-344

The function shall do the following when identifier is equal to CONFIG\_SID AND Cm\_mode is equal to CM\_WAIT:

1. Do the following when Download\_size is equal to (u8\_payload of sMess with index M\_FOUR added to (u8\_payload of sMess with index M\_FIVE bit shifted to left by M\_SHIFT\_BY\_8) added to (u8\_payload of sMess with index M\_SIX bit shifted to left by M\_SIXTEEN) added to (u8\_payload of sMess with index M\_SEVEN bit shifted to left by M\_SHIFT\_24)):

* Set Cm\_mode to CM\_DATA\_DOWN.
* Set cmu\_ok of Cm\_data to TRUE.

b) Otherwise do the following:

* Set Cm\_mode to CM\_DATAACK.
* Set cmu\_ok of Cm\_data to FALSE.

##### 10.4.4.7.3 daugwyapp-AppU32Nsc-LLR-003

Requirement ID: H398-LLD-GWY-FNC-345

The function shall do the following when identifier is equal to CONFIG\_SID AND Cm\_mode is equal to CM\_DATA\_DOWN:

1. Call ‘HwCopy’ with parameters reference to Xram\_cmu\_buffer with index as count variable, u8\_payload of sMess and M\_EIGHT.
2. Set count variable to count variable added to M\_EIGHT.
3. Set Cm\_mode to CM\_DATAACK when Count variable is greater than or equal to Download\_size.

##### 10.4.4.7.4 daugwyapp-AppU32Nsc-LLR-004

Requirement ID: H398-LLD-GWY-FNC-346

The function shall do the following when identifier is other than CONFIG\_SID OR Cm\_mode is not equal to CM\_DATA\_DOWN, CM\_WAIT and CM\_SETUP:

1. Call ‘HwCopy’ with parameters reference to command variable, u8\_payload of sMess and M\_TWO.

##### 10.4.4.7.5 daugwyapp-AppU32Nsc-LLR-005

Requirement ID: H398-LLD-GWY-FNC-347

The function shall do the following when identifier is other than CONFIG\_SID OR Cm\_mode is not equal to CM\_DATA\_DOWN, CM\_WAIT and CM\_SETUP:

1. Set aircr of M\_SCB to ((M\_HEX\_5FA bit shifted to left by M\_SCB\_AIRCR\_VECTKEY\_POS) bitwise OR with (aircr of M\_SCB bitwise AND with M\_SCB\_AIRCR\_PRIGROUP\_MSK) bitwise OR with M\_SCB\_AIRCR\_SYSRESETREQ\_MSK) when command variable is NSC\_RESET and identifier is GATEWAY\_SID.

##### 10.4.4.7.6 daugwyapp-AppU32Nsc-LLR-006

Requirement ID: H398-LLD-GWY-FNC-348

The function shall do the following when identifier is other than CONFIG\_SID OR Cm\_mode is not equal to CM\_DATA\_DOWN, CM\_WAIT and CM\_SETUPand command variable is NSC\_GET\_CM\_RCI:

1. Set Cm\_init to TRUE.
2. Set cm\_operational of Cm\_data to TRUE.
3. Set cm\_rci of Cm\_data to u8\_payload with index M\_TWO of sMess.
4. Set cm\_mode of Cm\_data to u8\_payload with index M\_THREE of sMess.
5. Set cm\_data\_valid of Cm\_data to u8\_payload with index M\_FOUR of sMess.
6. Set bridge\_usable of Cm\_data to TRUE when (NORMAL is equal to cm\_mode of Cm\_data) returns TRUE.
7. Set bridge\_usable of Cm\_data to FALSE when (NORMAL is equal to cm\_mode of Cm\_data) returns FALSE.

h) Set Bit\_mode to RCI\_BIT\_SET.

##### 10.4.4.7.7 daugwyapp-AppU32Nsc-LLR-007

Requirement ID: H398-LLD-GWY-FNC-349

The function shall do the following when identifier is other than CONFIG\_SID OR Cm\_mode is not equal to CM\_DATA\_DOWN, CM\_WAIT and CM\_SETUP and command variable is NSC\_SETETI:

1. Call ‘HwCopy’ with parameters reference to seconds of Xram\_data\_buffer, reference to u8\_payload of sMess and M\_FOUR.
2. Set Smt of pid of sid of sMess to M\_ZERO.
3. Set u8\_paysize of sMess to M\_SIX.
4. Set temporary variable to seconds of Xram\_data\_buffer.
5. Call ‘HwCopy’ with parameters reference to u8\_payload with index M\_TWO of sMess, reference to temporary variable and M\_FOUR.
6. Call ‘A825Xmit’ with parameter sMess.

##### 10.4.4.7.8 daugwyapp-AppU32Nsc-LLR-008

Requirement ID: H398-LLD-GWY-FNC-350

The function shall do the following when identifier is other than CONFIG\_SID OR Cm\_mode is not equal to CM\_DATA\_DOWN, CM\_WAIT and CM\_SETUP and command variable is NSC\_GETETI:

1. Set Smt of pid of sid of sMess to M\_ZERO.
2. Set u8\_paysize of sMess to M\_SIX.
3. Set temporary variable to seconds of Xram\_data\_buffer.
4. Call ‘HwCopy’ with parameters reference to u8\_payload with index M\_TWO of sMess, reference to temporary variable and M\_FOUR.
5. Call ‘A825Xmit’ with parameter sMess.

##### 10.4.4.7.9 daugwyapp-AppU32Nsc-LLR-009

Requirement ID: H398-LLD-GWY-FNC-351

The function shall do the following when identifier is other than CONFIG\_SID OR Cm\_mode is not equal to CM\_DATA\_DOWN, CM\_WAIT and CM\_SETUP and command variable is NSC\_CLEARLOG:

1. Set Smt of pid of sid of sMess to M\_ZERO.
2. Set u8\_paysize of sMess to M\_THREE.
3. Set u8\_payload with index M\_TWO of sMess to return value of ‘ClearLog’.
4. Call ‘A825Xmit’ with parameter sMess.

##### 10.4.4.7.10 daugwyapp-AppU32Nsc-LLR-010

Requirement ID: H398-LLD-GWY-FNC-352

The function shall do the following when identifier is other than CONFIG\_SID OR Cm\_mode is not equal to CM\_DATA\_DOWN, CM\_WAIT and CM\_SETUP and command variable is NSC\_GETLOG:

1. Set Smt of pid of sid of sMess to M\_ZERO.
2. Set u8\_paysize of sMess to M\_FOUR.
3. Call ‘HwCopy’ with parameters reference to u8\_payload with index M\_TWO of sMess, reference to numofFaults of Xram\_data\_buffer and M\_TWO.
4. Call ‘A825Xmit’ with parameter sMess.
5. Call ‘OsTimeDly’ with parameter M\_FIFTY.
6. Call ‘SendLog’ to send log data.

##### 10.4.4.7.11 daugwyapp-AppU32Nsc-LLR-014

Requirement ID: H398-LLD-GWY-FNC-356

The function shall do the following when identifier is other than CONFIG\_SID OR Cm\_mode is not equal to CM\_DATA\_DOWN, CM\_WAIT and CM\_SETUP.

Do nothing when command variable is other than NSC\_RESET, NSC\_GET\_CM\_RCI, NSC\_SETETI, NSC\_GETETI, NSC\_CLEARLOG, NSC\_GETLOG.

##### 10.4.4.7.12 daugwyapp-AppU32Nsc-LLR-015

Requirement ID: H398-LLD-GWY-FNC-357

The function shall return M\_ZERO.

### 10.4.5 ClearLog

Low Level Design Details about CSU ClearLog will follow in the sub sections.

#### 10.4.5.1 Brief Description

The function ClearLog clears the all Fault log conditions.

#### 10.4.5.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.4.5.3 List of global variables accessed and modified

Accessed : None

Modified : Xram\_data\_buffer

#### 10.4.5.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.4.5.5 Return Value

T\_BOOL - Retruns TRUE

#### 10.4.5.6 Other CSUs called by this CSU

None

#### 10.4.5.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to ClearLog

##### 10.4.5.7.1 daugwyapp-ClearLog-LLR-001

Requirement ID: H398-LLD-GWY-FNC-366

The function shall do the following by looping through M\_ZERO to MAX\_ERRORS (first loop counter) and M\_ZERO to M\_TWENTY(second loop counter):

1. Set error\_code of FAULTS with indices as first loop counter and second loop counter of Xram\_data\_buffer to M\_ZERO.
2. Set sub\_id of FAULTS with indices as first loop counter and second loop counter of Xram\_data\_buffer to M\_ZERO.
3. Set error\_count of FAULTS with indices as first loop counter and second loop counter of Xram\_data\_buffer to M\_ZERO.

##### 10.4.5.7.2 daugwyapp-ClearLog-LLR-002

Requirement ID: H398-LLD-GWY-FNC-367

The function shall

1. Set numofFaults of Xram\_data\_buffer to M\_ZERO.
2. return TRUE.

### 10.4.6 WragleFilter

Low Level Design Details about CSU WragleFilter will follow in the sub sections.

#### 10.4.6.1 Brief Description

This function WragleFilter applies the exponential filter.

#### 10.4.6.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.4.6.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.4.6.4 Parameter list (Input/Output)

Inputs : T\_UINT8 u8\_chan- XADC channel number

T\_SINT32 value - value for which filtering has to applied

T\_UINT32 power - power value

T\_UINT32 noise - noise value

Outputs : None

#### 10.4.6.5 Return Value

T\_SINT32 - returns filtered value

#### 10.4.6.6 Other CSUs called by this CSU

None

#### 10.4.6.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to WragleFilter

##### 10.4.6.7.1 daugwyapp-WragleFilter-LLR-001

Requirement ID: H398-LLD-GWY-FNC-6106

The function shall set significant filter bits to M\_DEC\_ZERO and delta value to difference of value and average value with index u8\_chan right shifted by power

When the following conditions are satisfied :

a. noise is not equal to M\_DEC\_ZERO

b. power is not equal to M\_DEC\_ZERO.

otherwise does nothing.

##### 10.4.6.7.2 daugwyapp-WragleFilter-LLR-002

Requirement ID: H398-LLD-GWY-FNC-6107

The function shall does the following when the following conditions are satisfied:

a. noise is not equal to M\_DEC\_ZERO

b. power is not equal to M\_DEC\_ZERO.

otherwise does nothing.

Set unsigned delta value to delta value when delta value is greater than M\_DEC\_ZERO

Else set unsigned delta value to difference between M\_DEC\_ZERO and delta value.

##### 10.4.6.7.3 daugwyapp-WragleFilter-LLR-003

Requirement ID: H398-LLD-GWY-FNC-6108

The function shall does the following when noise is not equal to M\_DEC\_ZERO and power is not equal to M\_DEC\_ZERO otherwise do nothing.

1. set new power to unsigned delta value right shifted by noise,

2. decrement significant filter bits by one and set new power to new power right shift by M\_ONE till new power is not equal to M\_DEC\_ZERO

3. set significant filter bits to power when significant filter bits value is greater than power minus M\_ONE.otherwise does nothing.

##### 10.4.6.7.4 daugwyapp-WragleFilter-LLR-004

Requirement ID: H398-LLD-GWY-FNC-6109

The function shall does the following

1. set average value with index u8\_chan to sum of average value with index u8\_chan and (delta value left shifted by significant filter bits)

2. set analog reading to average value with index u8\_chan right shifted by power

When the following conditions are satisfied :

a. noise is not equal to M\_DEC\_ZERO

b. power is not equal to M\_DEC\_ZERO.

otherwise does nothing.

##### 10.4.6.7.5 daugwyapp-WragleFilter-LLR-005

Requirement ID: H398-LLD-GWY-FNC-6110

The function shall do the following

(i)set average value with index u8\_chan added with difference between analog reading and result of average value with index u8\_chan right shifted by power.

(ii)set Analog reading to average value with index u8\_chan right shifted by power.

When the following conditions are satisfied (for (i) and (ii)) :

a. power is not equal to M\_DEC\_ZERO.

b. noise is equal to M\_DEC\_ZERO

otherwise does nothing

##### 10.4.6.7.6 daugwyapp-WragleFilter-LLR-006

Requirement ID: H398-LLD-GWY-FNC-6119

The function shall return filtered value.

### 10.4.9 SendLog

Low Level Design Details about CSU SendLog will follow in the sub sections.

#### 10.4.9.1 Brief Description

The SendLog function sends Log data.

#### 10.4.9.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.4.9.3 List of global variables accessed and modified

Accessed : Xram\_data\_buffer

Modified : None

#### 10.4.9.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.4.9.5 Return Value

T\_BOOL - Retruns TRUE

#### 10.4.9.6 Other CSUs called by this CSU

OsTimeDly

HwCopy

A825Xmit

GpioReadInputDataBit

#### 10.4.9.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to SendLog.

##### 10.4.9.7.1 daugwyapp-SendLog-LLR-001

Requirement ID: H398-LLD-GWY-FNC-394

The function shall set the A825 message fields as mentioned below:

- lcc to NOC

- sfid of bid of sid to M\_HOWELL\_DAU\_FID

- rsd of bid of sid to M\_ZERO

- lcl of bid of sid to M\_ONE

- pvt of bid of sid to M\_ONE

- doc of bid of sid to M\_DATA\_OBJECT\_CODE\_VALUE1

- payload size u8\_paysize to M\_EIGHT.

- rci of bid of sid to ((return value of ‘GpioReadInputDataBit’ (M\_HW\_GET\_RCI) with parameters (M\_GPIOC and M\_GPIOC\_SYS\_SEL)) bitwise AND with M\_THREE).

##### 10.4.9.7.2 daugwyapp-SendLog-LLR-002

Requirement ID: H398-LLD-GWY-FNC-395

The function shall do the following by looping through M\_ZERO to MAX\_ERRORS minus 1(first loop counter) and M\_ZERO to M\_TWENTY minus 1(second loop counter):

Do the following when error\_count of FAULTS with indices as first loop counter and second loop counter of Xram\_data\_buffer is greater than M\_ZERO:

1. Set message payload u8\_payload with index M\_ZERO to error\_code of FAULTS with indices as first loop counter and second loop counter of Xram\_data\_buffer
2. Set Message payload u8\_payload with index M\_ONE to sub\_id of FAULTS with indices as first loop counter and second loop counter of Xram\_data\_buffer
3. Set temporary variable to error\_count of FAULTS with indices as first loop counter and second loop counter of Xram\_data\_buffer.
4. Call ‘HwCopy’ with parameters reference to message payload u8\_payload with index as M\_TWO, reference to temporary variable and M\_TWO.
5. Call 'A825Xmit' with parameter reference to message.
6. Call ‘OsTimeDly’ with parameter M\_TIMEDLY\_FIVE\_HUNDRED.

##### 10.4.9.7.3 daugwyapp-SendLog-LLR-003

Requirement ID: H398-LLD-GWY-FNC-396

The function shall return TRUE.

### 10.4.10 AppSet825P2PMessage

Low Level Design Details about CSU AppSet825P2PMessage will follow in the sub sections.

#### 10.4.10.1 Brief Description

The AppSet825P2PMessage function sends a Peer 2 Peer Message via Arinc 825.

#### 10.4.10.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.4.10.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.4.10.4 Parameter list (Input/Output)

Inputs : T\_ESID who Server ID, where to send message

T\_UINT8 rci RCI/DAU

T\_UINT8 size size of buffer

T\_UINT8 \*data buffer/ data to be sent

Outputs : None

#### 10.4.10.5 Return Value

None

#### 10.4.10.6 Other CSUs called by this CSU

A825Xmit2

HwCopy

#### 10.4.10.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to AppSet825P2PMessage.

##### 10.4.10.7.1 daugwyapp-AppSet825P2PMessage-LLR-001

Requirement ID: H398-LLD-GWY-FNC-405

The function shall

a) Set the A825 message fields as mentioned below:

- lcc to NSC

- cfid of pid of sid to M\_HOWELL\_DAU\_FID

- smt of pid of sid to M\_ONE

- lcl of pid of sid to M\_ONE

- pvt of pid of sid to M\_ONE

- sfid of pid of sid to M\_HOWELL\_DAU\_FID

- sid of pid of sid to input parameter 'who'

- rci of pid of sid to input parameter 'rci' bitwise AND with M\_THREE.

- payload size u8\_paysize to input parameter 'size'

b) Copy the contents of the message 'data' into the message payload by calling ‘HwCopy’ with parameters message payload u8\_payload, data and size.

c) send an arinc 825 peer to peer message by calling the function ‘A825Xmit2’ with reference to the A825 message to be sent as parameter.

### 10.4.11 AppPow

Low Level Design Details about CSU AppPow will follow in the sub sections.

#### 10.4.11.1 Brief Description

This function multiplies the value by the power of mul

#### 10.4.11.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.4.11.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 10.4.11.4 Parameter list (Input/Output)

Inputs: value,mul

Outputs: mul

#### 10.4.11.5 Return Value

T\_FLOAT32 - returns the computed value

#### 10.4.11.6 Other CSUs called by this CSU

None

#### 10.4.11.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to AppPow.

##### 10.4.11.7.1 daugwyapp-AppPow-LLR-001

Requirement ID: H398-LLD-GWY-DRQ-6128

The function shall do the following when M\_ZERO is equal to mul:

Set the return value to M\_FP\_MINUS\_ONE when value less than M\_ZERO, otherwise set it to M\_FP\_ONE

otherwise does nothing.

##### 10.4.11.7.2 daugwyapp-AppPow-LLR-002

Requirement ID: H398-LLD-GWY-DRQ-6129

The function shall do the following when mul is less than M\_ZERO:

a) Set the mul to product of (mul with M\_MINUS\_ONE)

b) Set the new value to new value divide by value and loops till item less than mul plus M\_ONE

##### 10.4.11.7.3 daugwyapp-AppPow-LLR-003

Requirement ID: H398-LLD-GWY-DRQ-6130

The function shall do the following when mul is greater than or equal to M\_ZERO:

Set the new value to new value multiply by value and loop through M\_ZERO to till item less than mul minus M\_ONE

##### 10.4.11.7.4 daugwyapp-AppPow-LLR-004

Requirement ID: H398-LLD-GWY-DRQ-6131

Return new value.

### 10.4.12 AppTask

Low Level Design Details about CSU AppTask will follow in the sub sections.

#### 10.4.12.1 Brief Description

The AppTask function is the Application TASK.

It is used by the OS to create the thread/task for this module.

#### 10.4.12.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.4.12.3 List of global variables accessed and modified

Accessed : Cm\_data

App\_emode

App\_switching

Inita429

Crossexcitationtx

App\_powerup

Inita429

App\_strap

Initparam

Two\_sequence\_done,Xram\_data\_buffer

Modified : Cm\_data

App\_emode

Inita429

Crossexcitationtx

Inita429

#### 10.4.12.4 Parameter list (Input/Output)

Inputs : void \*pdata - pointer to task parameters

Outputs : void \*pdata - pointer to task parameters

#### 10.4.12.5 Return Value

None

#### 10.4.12.6 Other CSUs called by this CSU

DoInit

OsSemPend

WdogKickWatchDog

DoBit

DoNormal

Inita429Parameters

ScaleReadings

RxLabelsToCMU

Limmon

GpioResetBits

GpioSetBits

DoError

DoCmDownload

NonLabelParametersToCMU

DoExcite

HwMemset

TransmitMessage

#### 10.4.12.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to AppTask.

##### 10.4.12.7.1 daugwyapp-AppTask-LLR-001

Requirement ID: H398-LLD-GWY-FNC-423

The function shall

1. Set pdata to pdata added to M\_ZERO.
2. Call function ‘DoInit’ to initialize the task control parameters and error signals to default values.

##### 10.4.12.7.2 daugwyapp-AppTask-LLR-002

Requirement ID: H398-LLD-GWY-FNC-424

The function shall loop forever and do the following

a) Pend on semaphore by calling ‘OsSemPend’ with the following parameters

- Allocated semaphore for the task sem app task.

- pend timeout value as M\_ZERO.

- Address of local error code.

b) Call WdogKickWatchDog to reload watchdog counter.

##### 10.4.12.7.3 daugwyapp-AppTask-LLR-003

Requirement ID: H398-LLD-GWY-FNC-425

The function shall loop forever and do the following when App\_emode is APP\_BIT

1. Set Uas status to FALSE.
2. Call function ‘DoBit’ to perform boot initial task.

##### 10.4.12.7.4 daugwyapp-AppTask-LLR-012

Requirement ID: H398-LLD-GWY-FNC-5785

The function shall loop forever and do the following when App\_emode is APP\_BIT AND clear dau NVRAM is equal to FALSE

1. Set clear dau NVRAM to TRUE.

2. Call the function HwMemset with parameters as reference to App\_A825 of Xram\_data\_buffer,M\_ZERO and size of T\_APP\_A825.

3. Call the function HwMemset with parameters as reference to tx\_cross\_channel\_rs232 of Xram\_data\_buffer,M\_ZERO and size of T\_CROSS\_CHANNEL\_RS232.

4. Call the function HwMemset with parameters as reference to rx\_cross\_channel\_rs232 of Xram\_data\_buffer,M\_ZERO and size of T\_CROSS\_CHANNEL\_RS232.

5. Call the function HwMemset with parameters as reference to currDiscIP of Xram\_data\_buffer,M\_ZERO and size of currDiscIP of Xram\_data\_buffer.

6. Sets cross excitation transmitter with index M\_CROSS\_EXCITATION minus M\_TWO with value 0x0D.

7. Sets cross excitation transmitter with index M\_CROSS\_EXCITATION minus M\_ONE with value 0x0C.

8. Call TransmitMessage with parameters Crossexcitationtx, M\_CROSS\_EXCITATION and COM4.

##### 10.4.12.7.5 daugwyapp-AppTask-LLR-004

Requirement ID: H398-LLD-GWY-FNC-426

The function shall loop forever and do the following when the App\_emode is APP\_NORMAL.

a) Set Wait\_cnt to M\_ZERO.

b) Set the Uas status to TRUE.

d) Set in DAU switch flag to FALSE.

e) Set Crossexcitationtx with index M\_TWO to Dau\_excited.

f) Set Crossexcitationtx with index M\_THREE to TRUE.

g) Set Manual switching status and Manual switching clear to False when cross excitation receiver with index M\_THREE is NOT equal to M\_ZERO and manual switching clear equal to True. Otherwise manual switching clear and force channel clear is set to True.

##### 10.4.12.7.6 daugwyapp-AppTask-LLR-009

Requirement ID: H398-LLD-GWY-FNC-6233

The function shall loop forever and do the following when the App\_emode is APP\_NORMAL and App\_powerup is equal to FALSE.

* Set Cross\_excitation\_tx with index M\_EIGHT to Force\_channel\_change\_flag
* Call the function TransmitMessage with parameters (Cross\_excitation\_tx, M\_CROSS\_EXCITATION, COM4).

##### 10.4.12.7.7 daugwyapp-AppTask-LLR-013

Requirement ID: H398-LLD-GWY-FNC-6234

The function shall loop forever and do the following when the App\_emode is APP\_NORMAL.

* Set force channel clear and force channel change to False when ((cross excitation receiver with index M\_THREE AND NOT of Force channel change flag and NOT of cross excitation receiver with index M\_EIGHT) and force channel clear equal to True.
  + Call Inita429Parameters, set Inita429 to TRUE when Inita429 is FALSE

        - Call ‘DoNormal’ to perform gathering, processing and distribution of indicator's parameters.

##### 10.4.12.7.8 daugwyapp-AppTask-LLR-014

Requirement ID: H398-LLD-GWY-FNC-6235

The function shall loop forever and do the following when the App\_emode is APP\_NORMAL.

1) Do the following when App\_strap is equal to TRUE

⦁ Call RxLabelsToCMU to receive Afcs axis to CMU.

⦁ Call NonLabelParametersToCMU to transmit non label parameters to CMU when Init\_param is equal to TRUE AND Two\_sequence\_done is equal to TRUE.

⦁ Call ‘Limmon’ for limit monitoring with parameter reference to limit table.

##### 10.4.12.7.9 daugwyapp-AppTask-LLR-007

Requirement ID: H398-LLD-GWY-FNC-429

The function shall loop forever and call ‘DoCmDownload’ to request for the data to be downloaded from CMU when the App\_emode is APP\_DOWNLOAD.

##### 10.4.12.7.10 daugwyapp-AppTask-LLR-008

Requirement ID: H398-LLD-GWY-FNC-430

The function shall loop forever and do the following when the App\_emode is APP\_EXCITE:

1. Set start excite to TRUE.
2. Call ‘DoExcite’ to set the corresponding DAU Excitation.
3. Set App\_emode to APP\_NORMAL.

##### 10.4.12.7.11 daugwyapp-AppTask-LLR-010

Requirement ID: H398-LLD-GWY-FNC-432

The function shall loop forever and call ‘DoError’ to display the error for invalid call and shutdown indicator when the App\_emode other than APP\_BIT, APP\_NORMAL, APP\_MAINT, APP\_ERROR, APP\_DOWNLOAD, APP\_EXCITE and APP\_WAIT.

##### 10.4.12.7.12 daugwyapp-AppTask-LLR-011

Requirement ID: H398-LLD-GWY-FNC-433

The function shall call function ‘DoError’ to display the error for invalid call and shutdown indicator.

data.

### 10.4.13 DoInit

Low Level Design Details about CSU DoInit will follow in the sub sections.

#### 10.4.13.1 Brief Description

The DoInit function initializes the task control parameters and error signals to default values.

#### 10.4.13.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.4.13.3 List of global variables accessed and modified

Accessed : None

Modified : Bit\_mode

Cm\_data

App\_emode

#### 10.4.13.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.4.13.5 Return Value

None

#### 10.4.13.6 Other CSUs called by this CSU

ReCorrctPointers

CorrctPointers

#### 10.4.13.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to DoInit.

##### 10.4.13.7.1 daugwyapp-DoInit-LLR-001

Requirement ID: H398-LLD-GWY-FNC-442

The function shall

1. Set UAS status to FALSE.
2. Set app\_error of error of Cm\_data to FALSE.
3. Set dis\_error of error of Cm\_data to FALSE.
4. Set App\_emode to APP\_BIT.
5. Set Bit\_mode to BIT\_INIT.

### 10.4.14 DoBit

Low Level Design Details about CSU DoBit will follow in the sub sections.

#### 10.4.14.1 Brief Description

The DoBit function performs the Boot Initial Task and the following tasks

- Check XRAM

- Check RCI

- Check WarmBoot

- Check RCI with CMU

- Set Discrete and Analog RCI

- Set errors if any

- Do Checkup on communication hardware.

#### 10.4.14.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.4.14.3 List of global variables accessed and modified

Accessed : Cm\_data

Cm\_init

Bit\_mode

Cmdownload\_lp\_cmd\_data,Nvm\_cmu\_tx

Modified : Cm\_data

Cm\_mode

Cm\_init

Bit\_mode

App\_emode

Ex\_mode

#### 10.4.14.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.4.14.5 Return Value

None

#### 10.4.14.6 Other CSUs called by this CSU

HwMemset

XRamTest

GpioSetBits

GpioReadInputDataBit

IadcRead

WdogKickWatchDog

OsTimeDly

RterrError

RterrForever

A825Xmit

#### 10.4.14.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to DoBit.

##### 10.4.14.7.1 daugwyapp-DoBit-LLR-001

Requirement ID: H398-LLD-GWY-FNC-451

The function shall initialize the Arinc 825 message payload to zero by calling ‘HwMemset’ with parameters Arinc 825 message payload, M\_ZERO and M\_DLC\_SIZE\_EIGHT.

##### 10.4.14.7.2 daugwyapp-DoBit-LLR-002

Requirement ID: H398-LLD-GWY-FNC-452

The function shall do the following when the Bit\_mode is BIT\_INIT:

1. perform PBIT for XRAM by calling function 'XRamTest'.
2. Call ‘GpioSetBits’(M\_HW\_RED\_LED\_ON) with parameters M\_GPIOC and M\_GPIOC\_RED\_LED.
3. Set dau\_rci of Cm\_data to return value of ‘GpioReadInputDataBit’(M\_HW\_GET\_RCI) with parameters M\_GPIOC and M\_GPIOC\_SYS\_SEL.
4. Call ‘GpioSetBits’ (M\_HW\_CHARGE\_ON) with parameters M\_GPIOC and M\_GPIOC\_CHARGE\_CNTRL.
5. Call GpioResetBits(M\_HW\_RS232\_TEST\_MODE) with parameters M\_GPIOC and M\_GPIOC\_RS232\_MODE.

g) Set TRUE to Lpbk\_tx\_com with index RS232 COM2

Call CommPutChars with parameter("TWO", M\_DATA\_LENGTH1, M\_TIMEOUT10, RS232 COM2)

Call OsTimeDly with parameter M\_DLY\_200.

e) Call ‘GpioSetBits’(M\_HW\_RS232\_NORM\_MODE) with parameters M\_GPIOC and M\_GPIOC\_RS232\_MODE.

f) Call ‘OsTimeDly’ with parameter M\_DLY\_1000.

##### 10.4.14.7.3 daugwyapp-DoBit-LLR-007

Requirement ID: H398-LLD-GWY-FNC-6237

The function shall do the following when the Bit\_mode is BIT\_INIT:

a) Call 'OsTimeDly’ with parameter M\_DLY\_100

b) Set the A825 message fields as mentioned below:

- lcc to NSC

- cfid of pid of sid to M\_HOWELL\_DAU\_FID

- smt of pid of sid to M\_ONE

- lcl of pid of sid to M\_ONE

- pvt of pid of sid to M\_ONE

- sfid of pid of sid to M\_HOWELL\_DAU\_FID

- sid of pid of sid to CONFIG\_SID

- rci of pid of sid to M\_ZERO

- payload size u8\_paysize to M\_TWO

-u8\_payload:

i.First byte to NSC\_GET\_CM\_RCI bitwise AND with M\_EXTRCT\_LSB

ii.Second byte to NSC\_GET\_CM\_RCI bit shifted to right by M\_SHIFT\_BY\_8 bitwise AND with M\_EXTRCT\_LSB

c) Set Cm\_mode to CM\_INIT.

d) Set Bit\_mode to BIT\_WAIT.

e) Call ‘A825Xmit’ with reference to A825 message.

##### 10.4.14.7.4 daugwyapp-DoBit-LLR-003

Requirement ID: H398-LLD-GWY-FNC-453

The function shall do the following when the Bit\_mode is BIT\_WAIT

a) Increment the bit wait count.

b) Set the following when bit wait count is greater than or equal to (M\_APP\_RATE multiplied by M\_FOUR)

* Set Cm\_init to FALSE.
* Set bridge\_usable of Cm\_data to FALSE.
* Increment re try count by one.
* Set bit wait count to M\_ZERO and Bit mode to RCI\_BIT\_SET when re try count is greater than (M\_APP\_RATE multiplied by M\_TWO).

##### 10.4.14.7.5 daugwyapp-DoBit-LLR-008

Requirement ID: H398-LLD-GWY-FNC-6238

The function shall do the following when the Bit\_mode is BIT\_WAIT

i) Set the following when bit wait count is greater than or equal to (M\_APP\_RATE multiplied by M\_FOUR)

ii) Do the following when re try count is not greater than (M\_APP\_RATE multiplied by M\_TWO)

iii) The A825 message fields are set as mentioned below

⦁ lcc to NSC

⦁ cfid of pid of sid to M\_HOWELL\_DAU\_FID

⦁ smt of pid of sid to to M\_ONE

⦁ lcl of pid of sid to to M\_ONE

⦁ pvt of pid of sid to to M\_ONE

⦁ sfid of pid of sid to to M\_HOWELL\_DAU\_FID

⦁ sid of pid of sid to to CONFIG\_SID

⦁ rci of pid of sid to to M\_ZERO.

⦁ u8\_paysize to M\_TWO

⦁ u8\_payload

iv). First byte to (NSC\_GET\_CM\_RCI Bitwise AND with M\_EXTRCT\_LSB).

v). Second byte to ((NSC\_GET\_CM\_RCI bit shifted to right by M\_SHIFT\_BY\_8) Bitwise AND with M\_EXTRCT\_LSB).

vi). Cm\_mode to CM\_INIT.

vii). Bit\_mode to BIT\_WAIT.

viii). Call ‘A825Xmit’ with reference to A825 message.

##### 10.4.14.7.6 daugwyapp-DoBit-LLR-009

Requirement ID: H398-LLD-GWY-FNC-6239

The function shall do the following when the Bit\_mode is BIT\_WAIT

a) Do nothing when bit wait count is lesser than (M\_APP\_RATE multiplied by M\_FOUR).

##### 10.4.14.7.7 daugwyapp-DoBit-LLR-004

Requirement ID: H398-LLD-GWY-FNC-454

The function shall do the following when the Bit\_mode is RCI\_BIT\_SET

a)Do the following when Cm\_init is FALSE.

⦁ Set cm\_operational of Cm\_data to FALSE.

⦁ Set cm\_rci of Cm\_data to M\_THREE

⦁ Set cm\_data\_valid of Cm\_data to FALSE

b)Do nothing when Cm\_init is not equal to FALSE.

##### 10.4.14.7.8 daugwyapp-DoBit-LLR-010

Requirement ID: H398-LLD-GWY-FNC-6240

The function shall do the following when the Bit\_mode is RCI\_BIT\_SET

a) Call ‘OsTimeDly’ with parameter M\_FIFTY.

b) Do the following when cm\_operational of Cm\_data is FALSE:

⦁ Set Cmdownload\_lp\_cmd\_data to M\_MEMMAP\_CMU\_ADDR.

⦁ Call ReCorrctPointers()

⦁ Call ‘DoMirrorCheck’ with parameter M\_MEMMAP\_CMU\_ADDR.

⦁ Call CorrctPointers()

##### 10.4.14.7.9 daugwyapp-DoBit-LLR-013

Requirement ID: H398-LLD-GWY-FNC-6241

The function shall do the following when the Bit\_mode is RCI\_BIT\_SET

i) Set cm\_operational of Cm\_data is equal to FALSE.

⦁ Set cmu\_stat of Cm\_data to INVALID and call RterrError with parameters CMU\_ERR, M\_TWO, RterrForever,when cmu\_ok of Cm\_data is FALSE otherwsie Set Cmdownload\_lp\_cmd\_data to M\_MEMMAP\_CMU\_ADDR and cmu\_stat of Cm\_data to OK.

⦁ Set Nvm\_cmu\_tx to M\_MEMMAP\_CMU\_NVM\_STORE.

⦁ Call DoNvmCRCcheck()

⦁ Set App\_emode to APP\_EXCITE.

⦁ Set Start\_excite to TRUE.

##### 10.4.14.7.10 daugwyapp-DoBit-LLR-012

Requirement ID: H398-LLD-GWY-FNC-6242

The function shall do the following when the Bit\_mode is RCI\_BIT\_SET

a) Do the following when cm\_operational of Cm\_data is not equal to FALSE:

⦁ Set App\_emode to APP\_DOWNLOAD.

⦁ Set Cm\_mode to CM\_REQUEST.

##### 10.4.14.7.11 daugwyapp-DoBit-LLR-005

Requirement ID: H398-LLD-GWY-FNC-455

The function shall call ‘RterrError’ with parameters BIT\_ERR, M\_ZERO and pointer to function RterrForever to flash code accordingly when Bit\_mode is other than BIT\_INIT, BIT\_WAIT and RCI\_BIT\_SET.

### 10.4.15 DoAcquire

Low Level Design Details about CSU DoAcquire will follow in the sub sections.

#### 10.4.15.1 Brief Description

The DoAcquire function sends an acquire command via the Arinc 825 to obtain values from other boards.

#### 10.4.15.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.4.15.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.4.15.4 Parameter list (Input/Output)

Inputs : T\_UINT8 rci - the on side rci

Outputs : None

#### 10.4.15.5 Return Value

None

#### 10.4.15.6 Other CSUs called by this CSU

A825Xmit2

#### 10.4.15.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to DoAcquire.

##### 10.4.15.7.1 daugwyapp-DoAcquire-LLR-002

Requirement ID: H398-LLD-GWY-FNC-465

The function shall

a) Set A825 message fields as mentioned below:

- lcc to NOC

- sfid of bid of sid to M\_HOWELL\_DAU\_FID

- rsd of bid of sid to M\_ZERO

- lcl of bid of sid to M\_ONE

- pvt of bid of sid to M\_ONE

- doc of bid of sid to ACQUIRE

- rci of bid of sid to on side rci bitwise AND with M\_THREE

- payload size u8\_paysize to M\_ZERO.

b) Call ‘A825Xmit2’ with parameter reference to A825 message.

c) Set Acquire\_cmd\_sent\_flag to TRUE.

### 10.4.16 DoAcquireCrcPn

Low Level Design Details about CSU DoAcquireCrcPn will follow in the sub sections.

#### 10.4.16.1 Brief Description

This function sends acquire crc and part number command via the Arinc 825 to obtain values from other boards.

#### 10.4.16.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.4.16.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.4.16.4 Parameter list (Input/Output)

Inputs : T\_UINT8 rci - the on side rci

T\_UINT8 bus – bus status

Outputs : None

#### 10.4.16.5 Return Value

None

#### 10.4.16.6 Other CSUs called by this CSU

A825Xmit

A825Xmit2

#### 10.4.16.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to DoAcquireCrcPn.

##### 10.4.16.7.1 daugwyapp-DoAcquireCrcPn-LLR-001

Requirement ID: H398-LLD-GWY-FNC-475

The function shall send a arinc 825 broadcast message to obtain Partnumbers and CRCs

from other boards.

The A825 message fields are set as mentioned below

- lcc to NOC

- sfid of bid of sid to M\_HOWELL\_DAU\_FID

- rsd of bid of sid to M\_ZERO

- lcl of bid of sid to M\_ONE

- pvt of bid of sid to M\_ONE

- doc of bid of sid to ACQUIRE\_CRC\_PN

- rci of bid of sid to rci bitwise AND with M\_THREE.

- payload size u8\_paysize to M\_ZERO

##### 10.4.16.7.2 daugwyapp-DoAcquireCrcPn-LLR-002

Requirement ID: H398-LLD-GWY-FNC-476

The function shall do the following:

* Call A825Xmit with parameter reference of A825 message when bus is equal to M\_ONE
* Call A825Xmit2 with parameter reference of A825 message when bus is not equal to M\_ONE

### 10.4.17 DoAcquireCmuNvram

Low Level Design Details about CSU DoAcquireCmuNvram will follow in the sub sections.

#### 10.4.17.1 Brief Description

This function sends acquire cmu nvram command via the Arinc 825 to cmu+.

#### 10.4.17.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.4.17.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.4.17.4 Parameter list (Input/Output)

Inputs : T\_UINT8 rci - the on side rci

Outputs : None

#### 10.4.17.5 Return Value

None

#### 10.4.17.6 Other CSUs called by this CSU

A825Xmit

#### 10.4.17.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to DoAcquireCmuNvram.

##### 10.4.17.7.1 daugwyapp-DoAcquireCmuNvram-LLR-001

Requirement ID: H398-LLD-GWY-FNC-485

The function shall send a arinc 825 broadcast message to obtain Partnumbers and CRCs

from other boards by calling A825Xmit with parameter A825 message

1. The A825 message fields are set as mentioned below

- lcc to NOC

- sfid of bid of sid to M\_HOWELL\_DAU\_FID

- rsd of bid of sid to M\_ZERO

- lcl of bid of sid to M\_ONE

- pvt of bid of sid to M\_ONE

- doc of bid of sid to CMU\_NVRAM\_REQ

- rci of bid of sid to rci bitwise AND with M\_THREE.

- payload size u8\_paysize to M\_ZERO

Call A825Xmit2 with parameter reference of A825 message

### 10.4.18 DoAppUpdate

Low Level Design Details about CSU DoAppUpdate will follow in the sub sections.

#### 10.4.18.1 Brief Description

The DoAppUpdate updates and sends P2P message to discrete board

#### 10.4.18.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.4.18.3 List of global variables accessed and modified

Accessed : Cm\_data

Modified : None

#### 10.4.18.4 Parameter list (Input/Output)

Inputs : T\_NSCODE number - channel number of Node service code

T\_DISO\_STATE \*data - Message of DISO state

Outputs : None

#### 10.4.18.5 Return Value

None

#### 10.4.18.6 Other CSUs called by this CSU

HwCopy

AppSet825P2PMessage

#### 10.4.18.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to APPupdateDO.

##### 10.4.18.7.1 daugwyapp-DoAppUpdate-LLR-001

Requirement ID: H398-LLD-GWY-FNC-494

The function shall call ‘HwCopy’ to copy data from source buffer to destination buffer with parameters P2P Message, reference to number and M\_TWO.

##### 10.4.18.7.2 daugwyapp-DoAppUpdate-LLR-002

Requirement ID: H398-LLD-GWY-FNC-495

The function shall loop through loop counter1 from M\_TWO to M\_SIX and loop counter2 starts from M\_ZERO.

1. Set P2P Message with index as loop counter1 to data with index as loop counter2.
2. Increment loop counter2 by one.
3. Set P2P message with index as loop counter1 to (P2P message with index as loop counter1 added to (data with loop counter2 as index bit shifted to left by M\_SHIFT\_BY\_2))
4. Increment loop counter2 by one
5. Set P2P message with index as loop counter1 to (P2P message with index as loop counter1 added to (data with loop counter2 as index bit shifted to left by M\_SHIFT\_BY\_4))
6. Increment loop counter2 by one
7. Set P2P message with index as loop counter1 to (P2P message with index as loop counter1 added to (data with loop counter2 as index bit shifted to left by M\_SHIFT\_BY\_6))

##### 10.4.18.7.3 daugwyapp-DoAppUpdate-LLR-003

Requirement ID: H398-LLD-GWY-FNC-496

The function shall call ‘AppSet825P2PMessage’ with parameters DIU\_SID, cm\_rci of Cm\_data, P2P message and M\_SIX.

### 10.4.24 ExciteStatus

Low Level Design Details about ExciteStatus will follow in the sub sections.

#### 10.4.24.1 Brief Description

The ExciteStatus function determine when both are channels are Active or In-active.

#### 10.4.24.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.4.24.3 List of global variables accessed and modified

Accessed : Crossexcitationtx, Crossexcitationrx

Modified : Crossexcitationtx, App\_switching

#### 10.4.24.4 Parameter list (Input/Output)

None

#### 10.4.24.5 Return Value

None

#### 10.4.24.6 Other CSUs called by this CSU

TransmitMessage

#### 10.4.24.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to ExciteStatus.

##### 10.4.24.7.1 daugwyapp-ExciteStatus-LLR-001

Requirement ID: H398-LLD-GWY-FNC-6070

The function shall do the following in auto switching mode:

⦁   Returns from the loop when on airborne of Xram data buffer AND (Force channel change flag OR cross excitation receiver with index M\_EIGHT) is True.

##### 10.4.24.7.2 daugwyapp-ExciteStatus-LLR-002

Requirement ID: H398-LLD-GWY-FNC-6243

The function shall do the following :

a) Increment excite fail counter by one when ((Crossexcitationtx with index M\_TWO is equal to Crossexcitationrx with index M\_TWO) AND ((Crossexcitationtx with index M\_THREE is NOT equal to M\_ZERO) AND (Crossexcitationrx with index M\_THREE is NOT equal to M\_ZERO))) and excite fail counter is less than M\_APP\_RATE) returns TRUE.

##### 10.4.24.7.3 daugwyapp-ExciteStatus-LLR-003

Requirement ID: H398-LLD-GWY-FNC-6244

The function shall do the following :

⦁ Set Dau excited to TRUE, App\_switching to TRUE and Excite init to FALSE when ((Crossexcitationtx with index M\_TWO is equal to Crossexcitationrx with index M\_TWO AND Crossexcitationtx with index M\_THREE is equal to Crossexcitationrx with index M\_THREE) and excite fail counter is greater than or equal to M\_APP\_RATE and M\_HW\_GET\_RCI is equal to M\_ZERO) returns TRUE.

⦁ Set Dau excited to FALSE, App\_switching to TRUE and Excite init2 to FALSE when ((Crossexcitationtx with index M\_TWO is equal to Crossexcitationrx with index M\_TWO AND Crossexcitationtx with index M\_THREE is equal to Crossexcitationrx with index M\_THREE) and excite fail counter is greater than or equal to M\_APP\_RATE and M\_HW\_GET\_RCI is not equal to M\_ZERO) returns TRUE.

⦁ Set excite fail counter to M\_ZERO when ((Crossexcitationtx with index M\_TWO is equal to Crossexcitationrx with index M\_TWO AND Crossexcitationtx with index M\_THREE is equal to Crossexcitationrx with index M\_THREE) and excite fail counter is greater than or equal to M\_APP\_RATE ) returns TRUE.

⦁ Set Crossexcitationtx with index M\_TWO to Dau excited and call function TransmitMessage() with parameters Crossexcitationtx, M\_CROSS\_EXCITATION, COM4 when Crossexcitationtx with index M\_TWO is equal to Crossexcitationrx with index M\_TWO AND Crossexcitationtx with index M\_THREE is equal to Crossexcitationrx with index M\_THREE.

##### 10.4.24.7.4 daugwyapp-ExciteStatus-LLR-004

Requirement ID: H398-LLD-GWY-FNC-6245

The function shall do the following :

⦁ Set excite fail counter to M\_ZERO when (Crossexcitationtx with index M\_TWO is equal to Crossexcitationrx with index M\_TWO AND Crossexcitationtx with index M\_THREE is equal to Crossexcitationrx with index M\_THREE) returns FALSE.

⦁ Set Crossexcitationtx with index M\_TWO to Dau excited.

### 10.4.25 AppExciteConditions

Low Level Design Details about AppExciteConditions in auto switch mode will follow in the sub sections.

#### 10.4.25.1 Brief Description

This function performs the excitation conditions for RCI, Xram data buffer and discrete input status as TRUE or FALSE.

#### 10.4.25.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.4.25.3 List of global variables accessed and modified

Accessed : App\_strap, Rs232\_transmit\_data, Rs232\_present, Xram\_data\_buffer, Crossexcitationrx, Force\_channel\_change

Modified : App\_switching, Crossexcitationtx, Force\_channel\_change, Force\_channel\_change\_flag

#### 10.4.25.4 Parameter list (Input/Output)

Input: None

Output: None

#### 10.4.25.5 Return Value

None

#### 10.4.25.6 Other CSUs called by this CSU

ExciteStatus

ReconfigurationFromDU

ManualSwitching

ExciteFaultDetect

InvalidRtdTimer

ChipEventTimer

AppPow

#### 10.4.25.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to AppExciteConditions

##### 10.4.25.7.1 daugwyapp-AppExciteConditions-LLR-001

Requirement ID: H398-LLD-GWY-FNC-6079

The function shall do the following :

* Increment Counter\_initial\_excite by one when Counter\_initial\_excite is less than or equal to the value of M\_APP\_RATE multiplied by M\_TWO.
* Set Excite\_done to TRUE when Counter\_initial\_excite is greater than the value of M\_APP\_RATE multiplied by M\_TWO.
* Call the function ExciteStatus.
* Call the function ReconfigurationFromDU

##### 10.4.25.7.7 daugwyapp-AppExciteConditions-LLR-003

Requirement ID: H398-LLD-GWY-FNC-6081

The function shall do the following

⦁ Return from the function when ((on\_airborne of Xram\_data\_buffer is equal to FALSE) OR (Rs232\_transmit\_data is equal to FALSE) OR ((Rs232\_present is equal to TRUE )AND (App\_strap is equal to FALSE))).

### 10.4.26 DoNormal

Low Level Design Details about CSU DoNormal will follow in the sub sections.

#### 10.4.26.1 Brief Description

The DoNormal function performs gathering, processing and distribution of indicator's parameters

The function performs the following operation

- Turn Red LED off, Flash Green

- Acquire data,

- Send Fuel data to GPS

- Calculate data for Power assurance

- Backup data

- Check for missing data.

#### 10.4.26.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.4.26.3 List of global variables accessed and modified

Accessed : App\_switching

Cm\_data

Xram\_data\_buffer

App\_powerup

Modified : App\_switching

#### 10.4.26.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.4.26.5 Return Value

None

#### 10.4.26.6 Other CSUs called by this CSU

AppExciteConditions

AppExciteON

AppExciteOff

DoAcquire

CheckMissing

DoAcquireCrcPn

DoCheckup

GpioResetBits

GpioSetBits

#### 10.4.26.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to DoNormal.

##### 10.4.26.7.1 daugwyapp-DoNormal-LLR-002

Requirement ID: H398-LLD-GWY-FNC-518

The function shall do the following

* When App\_switching is TRUE:

1. Increment Switch count.
2. Set Switch count to M\_ZERO, App\_switching to FALSE and App\_powerup to TRUE

when Switch count is greater than (M\_APP\_RATE multiplied by M\_FIVE) Otherwise

Set switch count to M\_ZERO

##### 10.4.26.7.2 daugwyapp-DoNormal-LLR-003

Requirement ID: H398-LLD-GWY-FNC-519

The function shall

1. Call function 'GpioResetBits' (M\_HW\_RED\_LED\_OFF) with parameters M\_GPIOC and M\_GPIOC\_RED\_LED.
2. Call function 'DoAcquire' with parameter cm\_rci of Cm\_data.
3. Call function ‘AppExciteConditions’ to check for excitation condition when FALSE is equal to Excite done.
4. Set excitationState of Xram\_data\_buffer to Dau excited.
5. Return from the function when Excite done is FALSE.

##### 10.4.26.7.3 daugwyapp-DoNormal-LLR-004

Requirement ID: H398-LLD-GWY-FNC-520

The function shall do the following when flag Dau excited is TRUE:

1. When excite flag is TRUE OR Excite init is FALSE

* Set App\_switching to TRUE.
* Call OsTimeDly function with parameter M\_TWENTY.
* Call ‘AppExciteON’ with parameter cm\_rci of Cm\_data.
* Set Switch count to M\_ZERO.
* Set excite flag to FALSE.
* Set Excite init to TRUE.

b) Call ‘GpioSetBits‘(M\_HW\_GREEN\_LED\_ON) with parameters M\_GPIOC and M\_GPIOC\_GREEN\_LED when Green led count is less than the (M\_APP\_RATE divided by M\_TWO).

c) Call ‘GpioSetBits‘(M\_HW\_GREEN\_LED\_OFF) with parameters M\_GPIOC and M\_GPIOC\_GREEN\_LED when Green led count is not less than the (M\_APP\_RATE divided by M\_TWO).

##### 10.4.26.7.4 daugwyapp-DoNormal-LLR-005

Requirement ID: H398-LLD-GWY-FNC-521

The function shall do the following when flag Dau excited is not equal to TRUE:

1. Call ‘GpioSetBits‘(M\_HW\_GREEN\_LED\_ON) with parameters M\_GPIOC and M\_GPIOC\_GREEN\_LED.
2. When excite flag is FALSE OR Excite init2 is FALSE

* Set App\_switching to TRUE
* Call OsTimeDly function with parameter M\_TWENTY
* Set Switch count to M\_ZERO
* Call ‘AppExciteoff’ with parameter cm\_rci of Cm\_data.
* Set excite flag to TRUE.
* Set Excite init2 to TRUE.

##### 10.4.26.7.5 daugwyapp-DoNormal-LLR-006

Requirement ID: H398-LLD-GWY-FNC-522

The function shall

1. Call ‘DoAcquire’ with parameter cm\_rci of Cm\_data.
2. Call ‘CheckMissing’ to check for any missing data.
3. Increment the counter by one when it is greater than M\_APP\_RATE do the following:

* Set counter to M\_ZERO.
* Call ‘DoAcquireCrcPn’ with parameter cm\_rci of Cm\_data and M\_TWO

##### 10.4.26.7.6 daugwyapp-DoNormal-LLR-007

Requirement ID: H398-LLD-GWY-FNC-523

The function shall

1. Increment green led count by one.
2. Set green led count to M\_ZERO when green led count is greater than or equal to M\_APP\_RATE.

### 10.4.27 CheckMissing

Low Level Design Details about CSU CheckMissing will follow in the sub sections.

#### 10.4.27.1 Brief Description

The CheckMissing function checks if there is any missing data.

#### 10.4.27.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.4.27.3 List of global variables accessed and modified

Accessed : Xram\_data\_buffer, Cmu\_data\_counter

Modified : Xram\_data\_buffer, Cmu\_data\_counter, Analog\_data\_missing, Cm\_data, Cmu\_data\_missing, Discrete\_data\_missing

#### 10.4.27.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.4.27.5 Return Value

None

#### 10.4.27.6 Other CSUs called by this CSU

AppExciteOff  
AppExciteON

RterrError

#### 10.4.27.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to CheckMissing.

##### 10.4.27.7.1 daugwyapp-CheckMissing-LLR-002

Requirement ID: H398-LLD-GWY-FNC-6114

The function shall increment Discrete data counter and set Discrete\_data\_missing to FALSE when Discrete data counter is less than (M\_THREE mulitiplied by M\_APP\_RATE) . Otherwise calls RterrError with index A825\_INTER\_ERR, M\_ONE and RterrForever when Discrete\_data\_missing is False and set Discrete\_data\_missing to TRUE.

##### 10.4.27.7.2 daugwyapp-CheckMissing-LLR-003

Requirement ID: H398-LLD-GWY-FNC-6113

The function shall increment Analog data counter and set Analog\_data\_missing to FALSE when Analog data counter is less than (M\_THREE mulitiplied by M\_APP\_RATE) otherwise calls RterrError with index A825\_INTER\_ERR, M\_TWO and RterrForever when Analog\_data\_missing is False and set Analog\_data\_missing to TRUE.

##### 10.4.27.7.3 daugwyapp-CheckMissing-LLR-004

Requirement ID: H398-LLD-GWY-FNC-6112

The function shall increment Cmu\_data\_counter and set Cmu\_data\_missing to FALSE

* when Cmu\_data\_counter is less than (M\_THREE mulitiplied by M\_APP\_RATE)

otherwise call RterrError with parameters (A825\_INTER\_ERR, M\_THREE and RterrForever) when

Cmu\_data\_missing is False

* Set Cmu\_data\_missing to TRUE

##### 10.4.27.7.4 daugwyapp-CheckMissing-LLR-005

Requirement ID: H398-LLD-GWY-FNC-6111

The function shall perform as follows when TRUE is equal to Discrete\_data\_missing OR TRUE is equal to Analog\_data\_missing:

* Call the function AppExciteON with parameters cm\_rci of Cm\_data when TRUE is equal to Dau excited.
* Call the function AppExciteOff with parameters cm\_rci of Cm\_data when FALSE is equal to Dau excited.

##### 10.4.27.7.5 daugwyapp-CheckMissing-LLR-001

Requirement ID: H398-LLD-GWY-FNC-532

The function shall do the following by looping through M\_ZERO to M\_HUNDREAD to perform the below:

1. Increment count of analmiss with index as loop counter of App\_A825 of Xram\_data\_buffer.
2. Set state of analmiss with index as loop counter of App\_A825 of Xram\_data\_buffer to FALSE and analStat with index as loop counter of App\_A825 of Xram\_data\_buffer to MISSING when count of analmiss with index as loop counter of App\_A825 of Xram\_data\_buffer is greater than M\_TEN.

### 10.4.28 AppGetAnalog

Low Level Design Details about CSU AppGetAnalog will follow in the sub sections.

#### 10.4.28.1 Brief Description

The APPgetAnalog function performs the signal calcualtions, sensor unit values, Engine status and channel status for Analog .

#### 10.4.28.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.4.28.3 List of global variables accessed and modified

Accessed : Cmdownload\_lp\_cmd\_data

Appdata\_appdata

Xram\_data\_buffer

Modified : Dau\_eng\_status

#### 10.4.28.4 Parameter list (Input/Output)

Inputs : channel, table

Outputs : table

#### 10.4.28.5 Return Value

T\_SINT32 - returns the Analog value

#### 10.4.28.6 Other CSUs called by this CSU

LookupTableLookup

WragleFilter

HwAbs32

#### 10.4.28.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to APPgetAnalog.

##### 10.4.28.7.1 daugwyapp-AppGetAnalog-LLR-001

Requirement ID: H398-LLD-GWY-FNC-541

The function shall perform the following operations

1. Set table to table added to table\_offset with index channel of Cmdownload\_lp\_cmd\_data.
2. Set pi32\_x of lookup table to stable\_x of tables with index table of Appdata\_appdata.
3. Set pi32\_y of lookup table to stable\_y of tables with index table of Appdata\_appdata.
4. Set variable value to analog with index channel of App\_A825 of Xram\_data\_buffer

##### 10.4.28.7.2 daugwyapp-AppGetAnalog-LLR-004

Requirement ID: H398-LLD-GWY-FNC-544

The function shall call ‘LookupTableLookup’ with parameters reference to lookup table, variable value and

Reference of status and return value of function ‘LookupTableLookup’ gets stored in variable value when table is greater than M\_ZERO.

##### 10.4.28.7.3 daugwyapp-AppGetAnalog-LLR-005

Requirement ID: H398-LLD-GWY-FNC-545

The function shall

1. Set Dau\_eng\_status with index channel to CHAN\_ERR and variable value to M\_ZERO when status is equal to LS\_OUTSIDE.
2. Set Dau\_eng\_status with index channel to OK when when status is other than LS\_OUTSIDE.

##### 10.4.28.7.4 daugwyapp-AppGetAnalog-LLR-007

Requirement ID: H398-LLD-GWY-FNC-6115

The function shall perform as follows when channel is greater than or equal to M\_ONE AND channel is less than or equal to M\_FORTY\_FOUR:

a) Set variable value to return value of WragleFilter with parameters (channel, variable value, M\_SIX, M\_SIX).

b) Set force change value with index as channel to M\_DEC\_ZERO and set last value with index as channel to variable value when (return value of HwAbs32 with parameters (variable value subtracted from last value with index as channel) is greater than M\_DEC\_TWO) OR (force change value with index as channel is greater than M\_FORCE\_CHANGE\_CNT\_100) otherwise increment force change value with index as channel.

c) Set variable value to last value with index as channel.

##### 10.4.28.7.6 daugwyapp-AppGetAnalog-LLR-010

Requirement ID: H398-LLD-GWY-FNC-7325

The function shall perform the following operations when Analog channel is M\_FIVE.

* The Analog value set to the product of ( ratio of M\_TEN\_THOUSAND to analog with index Twenty of App\_A825 of Xram\_data\_buffer) and analog with index Five of App\_A825 of Xram\_data\_buffer.

##### 10.4.28.7.7 daugwyapp-AppGetAnalog-LLR-011

Requirement ID: H398-LLD-GWY-FNC-7326

The function shall perform the following operations When Analog channel is M\_SIX:

* The Analog value set to the product of ( ratio of M\_TEN\_THOUSAND to analog with index Twenty two of App\_A825 of Xram\_data\_buffer) and analog with index Six of App\_A825 of Xram\_data\_buffer.

##### 10.4.28.7.8 daugwyapp-AppGetAnalog-LLR-012

Requirement ID: H398-LLD-GWY-FNC-7327

The function shall perform the following operations when Analog channel is M\_TWENTY\_SIX:

* The Analog value set to the product of ( ratio of M\_TEN\_THOUSAND to analog with index Twenty Five of App\_A825 of Xram\_data\_buffer)

and analog with index Twenty Six of App\_A825 of Xram\_data\_buffer.

##### 10.4.28.7.9 daugwyapp-AppGetAnalog-LLR-013

Requirement ID: H398-LLD-GWY-FNC-7328

The function shall perform the following operations when Analog channel is M\_TWENTY\_EIGHT:

* The Analog value set to the product of ( ratio of M\_TEN\_THOUSAND to analog with index Twenty seven of App\_A825 of Xram\_data\_buffer)

and analog with index Twenty Eight of App\_A825 of Xram\_data\_buffer.

##### 10.4.28.7.10 daugwyapp-AppGetAnalog-LLR-014

Requirement ID: H398-LLD-GWY-FNC-7329

The function shall perform the following operations when Analog channel is M\_THIRTY:

* The Analog value set to the product of ( ratio of M\_TEN\_THOUSAND to analog with index Twenty Nine of App\_A825 of Xram\_data\_buffer)

and analog with index Thirty of App\_A825 of Xram\_data\_buffer.

##### 10.4.28.7.11 daugwyapp-AppGetAnalog-LLR-015

Requirement ID: H398-LLD-GWY-FNC-7330

The function shall perform the following operations when Analog channel is M\_THIRTY\_THREE:

* The Analog value set to the product of ( ratio of M\_TEN\_THOUSAND to analog with index Thirty one of App\_A825 of Xram\_data\_buffer)

and analog with index Thirty three of App\_A825 of Xram\_data\_buffer.

##### 10.4.28.7.12 daugwyapp-AppGetAnalog-LLR-016

Requirement ID: H398-LLD-GWY-FNC-7331

The function shall perform the following operations when Analog channel is M\_THIRTY\_FIVE:

* The Analog value set to the product of ( ratio of M\_TEN\_THOUSAND to analog with index Thirty Four of App\_A825 of Xram\_data\_buffer)

and analog with index Thirty Five of App\_A825 of Xram\_data\_buffer.

### 10.4.29 DoCheckup

Low Level Design Details about CSU DoCheckup will follow in the sub sections.

#### 10.4.29.1 Brief Description

The DoCheckup function runs the Checkup on subsystems

- Sets RS232/422 in test mode

- Audio Test

- Xram Test

- COM Loop Back Test

- A429 Loop Back Test

- Set RS232/422 in normal mode

- if CMU is Operational Turn on excitation, else error.

#### 10.4.29.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.4.29.3 List of global variables accessed and modified

Accessed : Cm\_data

Modified : Cm\_data  
 App\_emode

#### 10.4.29.4 Parameter list (Input/Output)

#### 10.4.29.5 Return Value

None

#### 10.4.29.6 Other CSUs called by this CSU

None

#### 10.4.29.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to DoCheckup.

##### 10.4.29.7.1 daugwyapp-DoCheckup-LLR-001

Requirement ID: H398-LLD-GWY-FNC-561

The function shall set the App\_emode to APP\_EXCITE.

### 10.4.30 DoExcite

Low Level Design Details about CSU DoExcite will follow in the sub sections.

#### 10.4.30.1 Brief Description

The DoExcite function sets DAU Excitation.

The module performs the following operations

- Check Excite mode

- Set off Side Excitation

- Turn OFF off-side Excitation

- Set Onside

- Test Discrete Outs if BIT.

#### 10.4.30.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.4.30.3 List of global variables accessed and modified

Accessed : Cm\_data

Rs232\_present

Xram\_data\_buffer

Crossexcitationtx

reconfiguration\_state\_sent

Modified : Crossexcitationtx

App\_switching

reconfiguration\_state\_sent

configuration\_mode

mode\_sent

default\_configuration\_mode

#### 10.4.30.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.4.30.5 Return Value

None

#### 10.4.30.6 Other CSUs called by this CSU

AppExciteOff

OsTimeDly

DoAcquire

TransmitMessage

#### 10.4.30.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to DoExcite.

##### 10.4.30.7.1 daugwyapp-DoExcite-LLR-001

Requirement ID: H398-LLD-GWY-FNC-570

The function shall do the following in Auto Switching:

* Set Dau excited to Negation of (excitationState of Xram\_data\_buffer) and set powerup config mode to Dau excited

when on\_airborne of Xram\_data\_buffer is FALSE otherwise set Dau excited to excitationState of Xram\_data\_buffer.

##### 10.4.30.7.2 daugwyapp-DoExcite-LLR-002

Requirement ID: H398-LLD-GWY-FNC-571

The function shall do the following in Manual Switching:

Call AppExciteOff with parameter cm\_rci of Cm\_data when Dau excited is TRUE otherwise call AppExciteOff with parameter cm\_rci of Cm\_data.

##### 10.4.30.7.3 daugwyapp-DoExcite-LLR-003

Requirement ID: H398-LLD-GWY-FNC-572

The function shall do the following:

⦁ Call DoAcquire with parameter cm\_rci of Cm\_data.

⦁ Set App\_switching to TRUE.

⦁ Call OsTimeDly() function with parameter M\_DLY\_100.

⦁ Set Crossexcitationtx with index M\_TWO to Dau excited.

⦁ Call TransmitMessage with parameters Crossexcitationtx, M\_CROSS\_EXCITATION, COM4.

##### 10.4.30.7.4 daugwyapp-DoExcite-LLR-004

Requirement ID: H398-LLD-GWY-FNC-7322

The function shall do the following:

* Set default\_configuration\_mode to DAU\_1\_MODE
* Set source\_selection\_sent to EIU\_1

When ((powerup\_config\_mode is equal to TRUE ) AND ( by calling the function GpioReadInputDataBit of parameters

(M\_GPIOC and M\_GPIOC\_SYS\_SEL) returns M\_ZERO)) OR ((powerup\_config\_mode is equal to FALSE) AND

(by calling the function GpioReadInputDataBit of parameters (M\_GPIOC and M\_GPIOC\_SYS\_SEL) returns M\_ONE)))

##### 10.4.30.7.5 daugwyapp-DoExcite-LLR-005

Requirement ID: H398-LLD-GWY-FNC-7323

The function shall do the following:

* Set default\_configuration\_mode to DAU\_2\_MODE
* Set source\_selection\_sent to EIU\_2

When ((powerup\_config\_mode is equal to TRUE ) AND ( by calling the function GpioReadInputDataBit of parameters

(M\_GPIOC and M\_GPIOC\_SYS\_SEL) returns M\_ ONE)) OR ((powerup\_config\_mode is equal to FALSE) AND

(by calling the function GpioReadInputDataBit of parameters (M\_GPIOC and M\_GPIOC\_SYS\_SEL) returns M\_ZERO)))

##### 10.4.30.7.6 daugwyapp-DoExcite-LLR-006

Requirement ID: H398-LLD-GWY-FNC-7324

The function shall do the following:

* Set reconfiguration\_state\_sent to DAU\_N\_MODE;
* Set configuration\_mode to reconfiguration\_state\_sent
* Set mode\_sent to DAU\_N\_MODE

##### 10.4.30.7.7 daugwyapp-DoExcite-LLR-007

Requirement ID: H398-LLD-GWY-FNC-7464

The function shall perform the following when on\_airborne of Xram\_data\_buffer is equal to FALSE.

1. The function shall do the following when M\_HW\_GET\_RCI is equal to M\_ZERO.

* + Set powerup master to EIU\_1 when excitationState of Xram\_data\_buffer is equal to TRUE otherwise set powerup\_master to EIU\_2.

##### 10.4.30.7.8 daugwyapp-DoExcite-LLR-008

Requirement ID: H398-LLD-GWY-FNC-7465

The function shall perform the following when on\_airborne of Xram\_data\_buffer is equal to FALSE.

1. The function shall do the following when M\_HW\_GET\_RCI is not equal to M\_ZERO.

* + Set powerup master to EIU\_2 when excitationState of Xram\_data\_buffer is equal to TRUE otherwise set powerup\_master to EIU\_1.

##### 10.4.30.7.9 daugwyapp-DoExcite-LLR-009

Requirement ID: H398-LLD-GWY-FNC-7466

The function shall perform the following when on\_airborne of Xram\_data\_buffer is not equal to FALSE.

* Set Reconfig\_capability\_test\_fail to TRUE when on\_airborne of Xram\_data\_buffer is equal to FALSE AND powerup\_master is equal to Source\_selection\_sent.

### 10.4.31 AppExciteOff

Low Level Design Details about CSU AppExciteOff will follow in the sub sections.

#### 10.4.31.1 Brief Description

The AppExciteOff function transmits arinc 825 message to Analog and digital board to turn off the excitation and discrete drive respectively for the inactive DAU.

#### 10.4.31.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.4.31.3 List of global variables accessed and modified

Accessed : Cm\_data

Modified : None

#### 10.4.31.4 Parameter list (Input/Output)

Inputs : T\_BOOL dau\_switch - parameter to determine if call to AppDauSwitch is to be made.

Outputs : None

#### 10.4.31.5 Return Value

None

#### 10.4.31.6 Other CSUs called by this CSU

A825Xmit2

OsTimeDly

#### 10.4.31.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to AppExciteOff.

##### 10.4.31.7.1 daugwyapp-AppExciteOff-LLR-001

Requirement ID: H398-LLD-GWY-FNC-596

The function shall

1. Set the A825 message fields as mentioned below:

- lcc to NSC

- client FID(cfid) of pid of sid to M\_HOWELL\_DAU\_FID

- smt of pid of sid to M\_ONE

- lcl of pid of sid to M\_ONE

- pvt of pid of sid to M\_ONE

- server FID(sfid) of pid of sid to M\_HOWELL\_DAU\_FID

- sid of pid of sid to MULTICAST\_SID

- rci of pid of sid to cm\_rci of Cm\_data bitwise AND with M\_THREE

- payload size (u8\_paysize) to M\_THREE

- u8\_payload:

1. First byte to NSC\_EXCITE\_SWITCH bitwise AND with M\_EXTRCT\_LSB
2. Second byte to NSC\_EXCITE\_SWITCH bit shifted to right by M\_SHIFT\_BY\_8
3. Third byte to M\_SWITCH\_OFF\_EXCITE

b) Call ‘A825Xmit2’ with parameter reference to A825 message.

##### 10.4.31.7.2 daugwyapp-AppExciteOff-LLR-002

Requirement ID: H398-LLD-GWY-FNC-597

The function shall call ‘OsTimeDly’ with parameter M\_DLY\_100 to delay tasks for a specific number of clock ticks.

##### 10.4.31.7.3 daugwyapp-AppExciteOff-LLR-003

Requirement ID: H398-LLD-GWY-FNC-598

The function shall do the following when M\_ZERO not equal to dau\_switch..

* call ‘OsTimeDly’ with parameter M\_DLY\_100 to delay tasks for a specific number of clock ticks.

### 10.4.32 AppExciteON

Low Level Design Details about CSU AppExciteON will follow in the sub sections.

#### 10.4.32.1 Brief Description

The AppExciteON function performs the excitation mode and switches the DAU .

#### 10.4.32.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.4.32.3 List of global variables accessed and modified

Accessed : Cm\_data

Modified : None

#### 10.4.32.4 Parameter list (Input/Output)

Inputs : T\_BOOL dau\_switch switch the dau state to True or Flase

Outputs : None

#### 10.4.32.5 Return Value

None

#### 10.4.32.6 Other CSUs called by this CSU

A825Xmit2

OsTimeDly

#### 10.4.32.7 Description of list of LLRs allocated

##### 10.4.32.7.1 daugwyapp-AppExciteON-LLR-001

Requirement ID: H398-LLD-GWY-FNC-607

The function shall

1. Set the A825 message fields as mentioned below:

- lcc to NSC

- client FID(cfid) of pid of sid to M\_HOWELL\_DAU\_FID

- smt of pid of sid to M\_ONE

- lcl of pid of sid to M\_ONE

- pvt of pid of sid to M\_ONE

- server FID(sfid) of pid of sid to M\_HOWELL\_DAU\_FID

- sid of pid of sid to MULTICAST\_SID

- rci of pid of sid to cm\_rci of Cm\_data bitwise AND with M\_THREE

- payload size (u8\_paysize) to M\_THREE

- u8\_payload:

1. First byte to NSC\_EXCITE\_SWITCH bitwise AND with M\_EXTRCT\_LSB
2. Second byte to NSC\_EXCITE\_SWITCH bit shifted to right by M\_SHIFT\_BY\_8
3. Third byte to M\_SWITCH\_ON\_EXCITE

b) Call ‘A825Xmit2’ with parameter reference to A825 message.

##### 10.4.32.7.2 daugwyapp-AppExciteON-LLR-002

Requirement ID: H398-LLD-GWY-FNC-608

The function shall call ‘OsTimeDly’ with parameter M\_DLY\_100 to delay tasks for a specific number of clock ticks.

##### 10.4.32.7.3 daugwyapp-AppExciteON-LLR-003

Requirement ID: H398-LLD-GWY-FNC-609

The function shall do the following when M\_ZERO not equal to dau\_switch.

* call ‘OsTimeDly’ with parameter M\_DLY\_100 to delay tasks for a specific number of clock ticks.

### 10.4.33 DoError

Low Level Design Details about CSU DoError will follow in the sub sections.

#### 10.4.33.1 Brief Description

The DoError function displays Error for invalid call and shutdown indicator.

The function performs the following operations

- Set error flag,

- Send dlu message,

- Shutdown drivers and

- Flash appropriate error.

#### 10.4.33.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.4.33.3 List of global variables accessed and modified

Accessed : None

Modified : Cm\_data

#### 10.4.33.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.4.33.5 Return Value

None

#### 10.4.33.6 Other CSUs called by this CSU

DoShutdown

RterrError

RterrForever

#### 10.4.33.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to DoError.

##### 10.4.33.7.1 daugwyapp-DoError-LLR-001

Requirement ID: H398-LLD-GWY-FNC-639

The function shall set the application error of structure app\_error of error of Cm\_data to TRUE.

##### 10.4.33.7.2 daugwyapp-DoError-LLR-003

Requirement ID: H398-LLD-GWY-FNC-641

The function shall call 'RterrError' with CMU\_ERR, M\_ZERO and function pointer to RterrForever as parameters to Flash the led for appropriate error type.

### 10.4.34 DoCmDownload

Low Level Design Details about CSU DoCmDownload will follow in the sub sections

#### 10.4.34.1 Brief Description

The DoCmDownload function performs the following operations

- Requests Download from CMU

- Sends acknowledgement to CMU

- CRC check of Downloaded data.

#### 10.4.34.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.4.34.3 List of global variables accessed and modified

Accessed : Cm\_data

Cm\_mode

Download\_size

T\_CMDOWNLOAD\_CMD\_DATA

Modified : Cm\_data

Cm\_mode

Cmdownload\_lp\_cmd\_data

#### 10.4.34.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.4.34.5 Return Value

#### 10.4.34.6 Other CSUs called by this CSU

DoMirrorCheck

OsTimeDly

RterrError

RterrForever

CorrctPointers

DoCheckup

A825Xmit

DoAcquireCrcPn

DoAcquireCmuNvram

#### 10.4.34.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to DoCmDownload.

##### 10.4.34.7.1 daugwyapp-DoCmDownload-LLR-001

Requirement ID: H398-LLD-GWY-FNC-650

The function shall do the following when the Cm\_mode is CM\_REQUEST:

a) The A825 message fields set as mentioned below

- lcc to NSC

- client FID(cfid) of pid of sid to M\_HOWELL\_DAU\_FID

- smt of pid of sid to M\_ONE

- lcl of pid of sid to M\_ONE

- pvt of pid of sid to M\_ZERO

- server FID(sfid) of pid of sid to M\_HOWELL\_DAU\_FID

- sid of pid of sid to CONFIG\_SID

- rci of pid of sid to cm\_rci of Cm\_data bitwise AND with M\_THREE

- payload size (u8\_paysize) to M\_DLC\_SIZE\_EIGHT

- u8\_payload

First byte to M\_THREE,

Second byte to M\_ZERO,

Third byte to (SDI\_CONFIG\_DATA Bitwise AND with M\_EXTRCT\_LSB),

Fourth byte to (SDI\_CONFIG\_DATA bit shifted to right by M\_SHIFT\_BY\_8),

and the rest of the payload is set to M\_ZERO respectively.

b) Call 'A825Xmit' with parameter reference to A825 message.

c) set Cm\_mode to CM\_SETUP.

##### 10.4.34.7.2 daugwyapp-DoCmDownload-LLR-002

Requirement ID: H398-LLD-GWY-FNC-651

The function shall do the following when the Cm\_mode is CM\_TRACK:

a) The A825 message fields set as mentioned below

- lcc to NSC

- client FID(cfid) of pid of sid to M\_HOWELL\_DAU\_FID

- smt of pid of sid to M\_ONE

- lcl of pid of sid to M\_ONE

- pvt of pid of sid to M\_ZERO

- server FID(sfid) of pid of sid to M\_HOWELL\_DAU\_FID

- sid of pid of sid to CONFIG\_SID

- rci of pid of sid to cm\_rci of Cm\_data bitwise AND with M\_THREE

- payload size (u8\_paysize) to M\_DLC\_SIZE\_EIGHT

- u8\_payload

First byte to M\_THREE,

Second byte to M\_ZERO,

Third byte to M\_ONE,

Fourth byte to M\_ZERO,

Fifth byte to (Download\_size Bitwise AND with M\_EXTRCT\_LSB),

Sixth byte to (Download\_size bit shifted to right by M\_SHIFT\_BY\_8),

and the rest of the payload is set to M\_ZERO respectively.

b) Call 'A825Xmit' with parameter reference to A825 message.

c) set Cm\_mode to CM\_WAIT.

##### 10.4.34.7.3 daugwyapp-DoCmDownload-LLR-003

Requirement ID: H398-LLD-GWY-FNC-652

The function shall do the following when the Cm\_mode is CM\_DATAACK:

a) Call ‘DoMirrorcheck’ to perform a CRC check on data received with parameter M\_MEMMAP\_CMU\_ADDR.

b) The A825 message fields set as mentioned below

- lcc to NSC

- client FID(cfid) of pid of sid to M\_HOWELL\_DAU\_FID

- smt of pid of sid to M\_ONE

- lcl of pid of sid to M\_ONE

- pvt of pid of sid to M\_ZERO

- server FID(sfid) of pid of sid to M\_HOWELL\_DAU\_FID

- sid of pid of sid to CONFIG\_SID

- rci of pid of sid to cm\_rci of Cm\_data bitwise AND with M\_THREE

- payload size (u8\_paysize) to M\_THREE

- u8\_payload:

i. First byte to M\_THREE

ii. Second byte to M\_ZERO

##### 10.4.34.7.4 daugwyapp-DoCmDownload-LLR-005

Requirement ID: H398-LLD-GWY-FNC-6253

The function shall do the following:

i. Perform as mentioned below when cmu\_ok of Cm\_data is equal to FALSE:

⦁ Third byte of u8\_payload to M\_ZERO.

⦁ cm\_download\_error of error of Cm\_data to TRUE.

⦁ Call ‘OsTimeDly ‘to delay for a specific number of clock ticks with parameter M\_HUNDREAD.

⦁ Set Cmdownload\_lp\_cmd\_data to M\_MEMMAP\_CMU\_ADDR.

⦁ Set cmu\_stat of Cm\_data to FALSE.

⦁ Call ‘RterrError’ with parameters CMU\_ERR,M\_TWO and function RterrForever.

##### 10.4.34.7.5 daugwyapp-DoCmDownload-LLR-006

Requirement ID: H398-LLD-GWY-FNC-6254

The function shall do the following:

a). Perform as mentioned below when cmu\_ok of Cm\_data is not equal to FALSE:

⦁ Call 'DoAcquireCrcPn' with parameters cm\_rci of Cm\_data and M\_ONE.

⦁ Third byte of u8\_payload to M\_ONE.

⦁ cm\_download\_error of error of Cm\_data to FALSE.

⦁ Call ‘OsTimeDly’ to delay a tasfor a specific number of clock ticks with parameter M\_HUNDREAD.

⦁ Set Cmdownload\_lp\_cmd\_data to M\_MEMMAP\_CMU\_ADDR.

⦁ Set cmu\_stat of Cm\_data to OK.

b) Call ‘A825Xmit´ with parameter reference to A825 message.

c) Call ‘CorrctPointers’ to update the cas stack.

d) Call ‘OsTimeDly’ to delay for a specific number of clock ticks with parameter M\_TEN.

e) Call ‘DoAcquireCmuNvram’ with parameter cm\_rci of Cm\_data.

f) Call ‘OsTimeDly’ to delay for a specific number of clock ticks with parameter M\_TEN.

g) Call ‘DoCheckup’ to start the system checkup.

##### 10.4.34.7.6 daugwyapp-DoCmDownload-LLR-004

Requirement ID: H398-LLD-GWY-FNC-653

The function shall do nothing when Cm\_mode is other than CM\_DATAACK, CM\_TRACK and CM\_REQUEST.

### 10.4.35 ReCorrctPointers

Low Level Design Details about ReCorrctPointers will follow in the sub sections.

#### 10.4.35.1 Brief Description

The ReCorrctPointers function ReCorrects pointers to calculate CRC when CMU is not connected.

#### 10.4.35.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.4.35.3 List of global variables accessed and modified

Accessed : Cmdownload\_lp\_cmd\_data

Modified : Cmdownload\_lp\_cmd\_data

#### 10.4.35.4 Parameter list (Input/Output)

None

#### 10.4.35.5 Return Value

None

#### 10.4.35.6 Other CSUs called by this CSU

ProcessEngineNoCMU

#### 10.4.35.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to ReCorrctPointers

##### 10.4.35.7.1 daugwyapp-ReCorrctPointers-LLR-001

Requirement ID: H398-LLD-GWY-FNC-6009

The function shall do the following:

a) Set Offset variable to M\_MEMMAP\_OFFSET\_VAL.

b) Set base variable to (engine of Cmdownload\_lp\_cmd\_data subtracted by M\_MEMMAP\_CMU\_START).

##### 10.4.35.7.2 daugwyapp-ReCorrctPointers-LLR-002

Requirement ID: H398-LLD-GWY-FNC-6010

The function shall call ‘ProcessEngineNoCMU’ by lopping through M\_ZERO to numofengines of Cmdownload\_lp\_cmd\_data with parameter reference to engine with index as loop counter of Cmdownload\_lp\_cmd\_data.

##### 10.4.35.7.3 daugwyapp-ReCorrctPointers-LLR-003

Requirement ID: H398-LLD-GWY-FNC-6011

The function shall set engine of Cmdownload\_lp\_cmd\_data to (Offset variable added to base Variable).

##### 10.4.35.7.4 daugwyapp-ReCorrctPointers-LLR-004

Requirement ID: H398-LLD-GWY-FNC-6012

The function shall set paramData of aircraft of Cmdownload\_lp\_cmd\_data to offset variable added to paramData of aircraft of Cmdownload\_lp\_cmd\_data substracted by M\_MEMMAP\_CMU\_START.

##### 10.4.35.7.5 daugwyapp-ReCorrctPointers-LLR-005

Requirement ID: H398-LLD-GWY-FNC-6013

The function shall do the following:

a) Set the pointer A429in of Cmdownload\_lp\_cmd\_data to (Offset variable added to the pointer A429in of Cmdownload\_lp\_cmd\_data substracted by M\_MEMMAP\_CMU\_START).

b) Set the pointer A429out\_bnr of Cmdownload\_lp\_cmd\_data to (Offset variable added to the pointer A429out\_bnr of Cmdownload\_lp\_cmd\_data substracted by M\_MEMMAP\_CMU\_START).

c) Set the pointer A429out\_ddw of Cmdownload\_lp\_cmd\_data to (Offset variable added to the pointer A429out\_ddw of Cmdownload\_lp\_cmd\_data substracted by M\_MEMMAP\_CMU\_START).

d)Set the base as exc\_mon\_data of Cmdownload\_lp\_cmd\_data substracted by M\_MEMMAP\_CMU\_START

##### 10.4.35.7.6 daugwyapp-ReCorrctPointers-LLR-006

Requirement ID: H398-LLD-GWY-FNC-6152

The function shall loop through M\_ZERO to numOf\_exc\_mon\_data of Cmdownload\_lp\_cmd\_data and perform the following

* call ‘ProcessExciteNoCMU’ with parameter reference of exc\_mon\_data with index as loop count of Cmdownload\_lp\_cmd\_data.

##### 10.4.35.7.7 daugwyapp-ReCorrctPointers-LLR-007

Requirement ID: H398-LLD-GWY-FNC-6153

The function shall do the following:

a)Set the pointer exc\_mon\_data of Cmdownload\_lp\_cmd\_data to Offset variable added to the base.

b)Return from the function.

### 10.4.36 CorrctPointers

Low Level Design Details about CSU CorrctPointers will follow in the sub sections.

#### 10.4.36.1 Brief Description

The CorrctPointers function corrects pointers present in ACD to the correct reference address

#### 10.4.36.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.4.36.3 List of global variables accessed and modified

Accessed : Cmdownload\_lp\_cmd\_data

Modified : Cmdownload\_lp\_cmd\_data

#### 10.4.36.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.4.36.5 Return Value

None

#### 10.4.36.6 Other CSUs called by this CSU

SetMonLimits

ProcessEngine

ProcessExcite

#### 10.4.36.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to CorrctPointers.

##### 10.4.36.7.1 daugwyapp-CorrctPointers-LLR-001

Requirement ID: H398-LLD-GWY-FNC-662

The function shall do the following:

1. Set Offset variable to M\_MEMMAP\_CMU\_START subtracted by M\_MEMMAP\_OFFSET\_VAL.
2. Set base variable to (engine of Cmdownload\_lp\_cmd\_data)
3. Set engine of Cmdownload\_lp\_cmd\_data to (Offset variable added to base Variable).

##### 10.4.36.7.2 daugwyapp-CorrctPointers-LLR-002

Requirement ID: H398-LLD-GWY-FNC-663

The function shall call ‘ProcessEngine’ by lopping through M\_ZERO to numofengines of Cmdownload\_lp\_cmd\_data with parameter reference to engine with index as loop counter of Cmdownload\_lp\_cmd\_data

##### 10.4.36.7.3 daugwyapp-CorrctPointers-LLR-003

Requirement ID: H398-LLD-GWY-FNC-664

The function shall set paramData of aircraft of Cmdownload\_lp\_cmd\_data to offset variable added to paramData of aircraft of Cmdownload\_lp\_cmd\_data.

##### 10.4.36.7.4 daugwyapp-CorrctPointers-LLR-005

Requirement ID: H398-LLD-GWY-FNC-666

The function shall do the following:

a)Set the pointer A429in of Cmdownload\_lp\_cmd\_data to (Offset variable added to the pointer A429in of Cmdownload\_lp\_cmd\_data).

b)Set the pointer A429out\_bnr of Cmdownload\_lp\_cmd\_data to (Offset variable added to the pointer A429out\_bnr of Cmdownload\_lp\_cmd\_data).

c)Set the pointer A429out\_ddw of Cmdownload\_lp\_cmd\_data to (Offset variable added to the pointer A429out\_ddw of Cmdownload\_lp\_cmd\_data).

d)Set the pointer exc\_mon\_data of Cmdownload\_lp\_cmd\_data to (Offset variable added to the pointer exc\_mon\_data of Cmdownload\_lp\_cmd\_data).

##### 10.4.36.7.5 daugwyapp-CorrctPointers-LLR-006

Requirement ID: H398-LLD-GWY-FNC-6154

The function shall loop through M\_ZERO to numOf\_exc\_mon\_data of Cmdownload\_lp\_cmd\_data and perform the following

* call ‘ProcessExcite’ function with parameter reference of exc\_mon\_data with index as loop count of Cmdownload\_lp\_cmd\_data

##### 10.4.36.7.6 daugwyapp-CorrctPointers-LLR-007

Requirement ID: H398-LLD-GWY-FNC-6155

The function shall do the following:

a)call the function SetMonLimits to set the limits for monitoring from ACD.

b)Return from the function.

### 10.4.37 ProcessEngineNoCMU

Low Level Design Details about ProcessEngineNoCMU will follow in the sub sections.

#### 10.4.37.1 Brief Description

The ProcessEngineNoCMU function process the engine NOCMU parameter data.

#### 10.4.37.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.4.37.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.4.37.4 Parameter list (Input/Output)

Inputs : T\_CMDOWNLOAD\_ENGINE\_DATA \*engine - engine parameter data

Outputs : T\_CMDOWNLOAD\_ENGINE\_DATA \*engine - engine parameter data

#### 10.4.37.5 Return Value

None

#### 10.4.37.6 Other CSUs called by this CSU

None

#### 10.4.37.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to ProcessEngineNoCMU

##### 10.4.37.7.1 daugwyapp-ProcessEngineNoCMU-LLR-001

Requirement ID: H398-LLD-GWY-FNC-6000

The function shall set paramData of engine to (Offset value added to (paramData of engine substracted by M\_MEMMAP\_CMU\_START)).

### 10.4.38 ProcessExciteNoCMU

Low Level Design Details about ProcessExciteNoCMU will follow in the sub sections.

#### 10.4.38.1 Brief Description

This function process the exceedance NOCMU parameter data.

#### 10.4.38.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.4.38.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 10.4.38.4 Parameter list (Input/Output)

Input: T\_LIM\_MON\_TABLE \*exc\_mon

Output: T\_LIM\_MON\_TABLE \*exc\_mon

#### 10.4.38.5 Return Value

None

#### 10.4.38.6 Other CSUs called by this CSU

None

#### 10.4.38.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to ProcessExciteNoCMU

##### 10.4.38.7.1 daugwyapp - ProcessExciteNoCMU - LLR-001

Requirement ID: H398-LLD-GWY-FNC-6164

The function shall loop from M\_ZERO to CAS of entry with index loop counter of exc\_mon not equal to EOT and increment loop counter.

1) set lim\_others of entry with index loop counter of exc\_mon to (Offset value added to (lim\_others of entry with index loop counter of exc\_mon substracted by M\_MEMMAP\_CMU\_START)) when lim\_others of entry with index loop counter of exc\_mon is NOT equal to M\_NULL.

2)set entry of exc\_mon to (Offset value added to (entry of exc\_mon substracted by M\_MEMMAP\_CMU\_START)).

### 10.4.39 ProcessEngine

Low Level Design Details about CSU ProcessEngine will follow in the sub sections.

#### 10.4.39.1 Brief Description

The ProcessEngine function processes the engine parameter data .

#### 10.4.39.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.4.39.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.4.39.4 Parameter list (Input/Output)

Inputs : T\_CMDOWNLOAD\_ENGINE\_DATA \*engine - engine parameter data

Outputs : T\_CMDOWNLOAD\_ENGINE\_DATA \*engine - engine parameter data

#### 10.4.39.5 Return Value

None

#### 10.4.39.6 Other CSUs called by this CSU

None

#### 10.4.39.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to ProcessEngine.

##### 10.4.39.7.1 daugwyapp-ProcessEngine-LLR-001

Requirement ID: H398-LLD-GWY-FNC-675

The function shall set paramData of engine to (Offset value added to paramData of engine).

### 10.4.40 ProcessExcite

Low Level Design Details about ProcessExcite will follow in the sub sections.

#### 10.4.40.1 Brief Description

This function process the exceedance parameter data.

#### 10.4.40.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.4.40.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 10.4.40.4 4 Parameter list (Input/Output)

Input: T\_LIM\_MON\_TABLE \*exc\_mon

Output: T\_LIM\_MON\_TABLE \*exc\_mon

#### 10.4.40.5 Return Value

None

#### 10.4.40.6 Other CSUs called by this CSU

None

#### 10.4.40.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to ProcessExcite

##### 10.4.40.7.1 daugwyapp-ProcessExcite-LLR-001

Requirement ID: H398-LLD-GWY-FNC-6173

The function shall do the following:

1) Set entry of exc\_mon to additon of Offset value to entry of exc\_mon

2) Loop from M\_ZERO to CAS of entry with index loop counter of exc\_mon is not equal to EOT and increment loop counter

3) Set lim\_others of entry with index loop counter of exc\_mon to (Offset value added to lim\_others of entry with index loop counter of exc\_mon) when lim\_others of entry with index loop counter of exc\_mon is not equal to M\_NULL.

### 10.4.41 ReconfigurationFromDU

Low Level Design Details about ReconfigurationFromDU will follow in the sub sections.

#### 10.4.41.1 Brief Description

This function checks the current configuration mode

#### 10.4.41.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.4.41.3 List of global variables accessed and modified

Accessed : configuration\_mode

default\_configuration\_mode

mode\_sent

Rx\_a429

Modified : configuration\_mode

source\_selection\_sent

reconfiguration\_state\_sent

mode\_sent

#### 10.4.41.4 Parameter list (Input/Output)

Input: None

Output: None

#### 10.4.41.5 Return Value

None

#### 10.4.41.6 Other CSUs called by this CSU

None

#### 10.4.41.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to ReconfigurationFromDU

##### 10.4.41.7.1 daugwyapp-ReconfigurationFromDU-LLR-001

Requirement ID: H398-LLD-GWY-FNC-7287

The function shall perform the following:

* Set configuration\_mode to DAU\_1\_MODE
* Set configuration\_mode to DAU\_N\_MODE and source\_selection\_sent to EIU\_1 when mode\_sent is equal to DAU\_N\_MODE

when (Dau\_excited is equal to TRUE ) AND(by calling the function GpioReadInputDataBit of parameters (M\_GPIOC and M\_GPIOC\_SYS\_SEL) returns the value M\_ZERO)) OR ((Dau\_excited is equal to FALSE ) AND (by calling the function GpioReadInputDataBit of parameters (M\_GPIOC and M\_GPIOC\_SYS\_SEL) returns the value M\_ONE))

##### 10.4.41.7.2 daugwyapp-ReconfigurationFromDU-LLR-002

Requirement ID: H398-LLD-GWY-FNC-7288

The function shall perform the following:

* Set configuration\_mode to DAU\_2\_MODE
* Set configuration\_mode to DAU\_N\_MODE and source\_selection\_sent to EIU\_2 when mode\_sent is equal to DAU\_N\_MODE

when (Dau\_excited is equal to TRUE ) AND(by calling the function GpioReadInputDataBit of parameters (M\_GPIOC and M\_GPIOC\_SYS\_SEL) returns the value M\_ONE)) OR ((Dau\_excited is equal to FALSE ) AND (by calling the function GpioReadInputDataBit of parameters (M\_GPIOC and M\_GPIOC\_SYS\_SEL) returns the value M\_ZERO))

##### 10.4.41.7.3 daugwyapp-ReconfigurationFromDU-LLR-019

Requirement ID: H398-LLD-GWY-FNC-7407

The function shall perform the following when request valid is FALSE

* Set Previous reconfiguration state to reconfiguration\_state\_sent.
* Set previous source selection to source\_selection\_sent.
* Set previous Dauexcited to Dau\_excited.

##### 10.4.41.7.4 daugwyapp-ReconfigurationFromDU-LLR-020

Requirement ID: H398-LLD-GWY-FNC-7408

The function shall perform the following

* Set Reconfiguration\_state\_sent to previous reconfiguration state.
* Set Source\_selection\_sent to previous source selection.
* Set Configuration\_mode to previous Reconfiguration\_state.
* Set Dauexcited to previous Dau\_excited

When the following conditions are satisified

1. request valid is FALSE
2. (reconfig\_discrepancy is equal to TRUE) OR (master\_discrepancy is equal toTRUE) OR (master\_reconfig\_fault is equal to TRUE)

##### 10.4.41.7.5 daugwyapp-ReconfigurationFromDU-LLR-003

Requirement ID: H398-LLD-GWY-FNC-7289

The function shall perform the following when request valid is equal to FALSE

* Set reconfig request1 to BITWISE AND of (data of Rx\_a429 LEFT SHIFT of index [Priority[IP\_LABEL\_011\_SYSMSG\_1]]and [IP\_LABEL\_011\_SYSMSG\_1] by M\_TEN) and M\_RECONFIG\_REQUEST\_MASK\_2

when status of Rx\_a429 of index [Priority[IP\_LABEL\_011\_SYSMSG\_1]] and [IP\_LABEL\_011\_SYSMSG\_1] is OK

##### 10.4.41.7.6 daugwyapp-ReconfigurationFromDU-LLR-004

Requirement ID: H398-LLD-GWY-FNC-7290

The function shall perform the following when request valid is equal to FALSE

* Set reconfig request2 to BITWISE AND of (data of Rx\_a429 of index [Priority[IP\_LABEL\_011\_SYSMSG\_2]]and [IP\_LABEL\_011\_SYSMSG\_2] RIGHT SHIFT by M\_TEN) and M\_RECONFIG\_REQUEST\_MASK\_2

when status of Rx\_a429 of index [Priority[IP\_LABEL\_011\_SYSMSG\_2]] and [IP\_LABEL\_011\_SYSMSG\_2] is OK

##### 10.4.41.7.7 daugwyapp-ReconfigurationFromDU-LLR-005

Requirement ID: H398-LLD-GWY-FNC-7291

The function shall perform the following when request valid is equal to FALSE

* Set reconfig request to reconfig request1

when (reconfig request1 is not equal to configuration\_mode) AND (status of Rx\_a429 of index [Priority[IP\_LABEL\_011\_SYSMSG\_1]] and

[IP\_LABEL\_011\_SYSMSG\_1] is equal to OK)

##### 10.4.41.7.8 daugwyapp-ReconfigurationFromDU-LLR-006

Requirement ID: H398-LLD-GWY-FNC-7292

The function shall perform the following when request valid is equal to FALSE

* Set reconfig request to reconfig request2 when (reconfig reques2 is not equal to configuration\_mode) AND (status of Rx\_a429 of index [Priority[IP\_LABEL\_011\_SYSMSG\_2]] and [IP\_LABEL\_011\_SYSMSG\_2] is equal to OK)

##### 10.4.41.7.9 daugwyapp-ReconfigurationFromDU-LLR-025

Requirement ID: H398-LLD-GWY-FNC-7450

The function shall perform the following when request valid is not equal to FALSE.

* Increment request timer
* set request timer to M\_ZERO and request valid to FALSE when request timer is greater than or equal to M\_500\_MILLISECOND

##### 10.4.41.7.10 daugwyapp-ReconfigurationFromDU-LLR-007

Requirement ID: H398-LLD-GWY-FNC-7293

The function shall perform the following

* Set request valid to TRUE.

when reconfig request is not equal to configuration\_mode

##### 10.4.41.7.11 daugwyapp-ReconfigurationFromDU-LLR-008

Requirement ID: H398-LLD-GWY-FNC-7294

1. The function shall perform the following

* Set source selection to EIU\_1

when the following conditions are satisified:

1. reconfig request is not equal to configuration\_mode
2. reconfig request is equal to DAU\_N\_MODE .
3. (configuration\_mode is equal to DAU\_1\_MODE) AND(default\_configuration\_mode is equal to DAU\_1\_MODE))

##### 10.4.41.7.12 daugwyapp-ReconfigurationFromDU-LLR-009

Requirement ID: H398-LLD-GWY-FNC-7295

The function shall perform the following

* Set source selection to EIU\_2 and Dauexcite to NOT of Dauexcite.

when the following conditions are satisified:

1. reconfig request is not equal to configuration\_mode
2. reconfig request is equal to DAU\_N\_MODE .
3. (configuration\_mode is equal to DAU\_1\_MODE) AND(default\_configuration\_mode is equal to DAU\_2\_MODE))

##### 10.4.41.7.13 daugwyapp-ReconfigurationFromDU-LLR-010

Requirement ID: H398-LLD-GWY-FNC-7296

The function shall perform the following

* Set source selection to EIU\_1 and Dauexcite to NOT of Dauexcite

when the following conditions are satisified:

1. reconfig request is not equal to configuration\_mode.
2. reconfig request is equal to DAU\_N\_MODE .
3. (configuration\_mode is equal to DAU\_2\_MODE) AND(default\_configuration\_mode is equal to DAU\_1\_MODE).

##### 10.4.41.7.14 daugwyapp-ReconfigurationFromDU-LLR-011

Requirement ID: H398-LLD-GWY-FNC-7297

* The function shall perform the following .

1. Set source selection to EIU\_2 when (configuration\_mode is equal to DAU\_2\_MODE) AND (default\_configuration\_mode is equal to DAU\_2\_MODE))
2. Set reconfiguration\_state to DAU\_N\_MODE.
3. Set mode\_sent to DAU\_N\_MODE.

when the following conditions are satisified:

* reconfig request is not equal to configuration\_mode.
* reconfig request is equal to DAU\_N\_MODE

##### 10.4.41.7.15 daugwyapp-ReconfigurationFromDU-LLR-012

Requirement ID: H398-LLD-GWY-FNC-7298

The function shall perform the following

* Do Nothing when the following conditions are satisified:

1. reconfig request is not equal to configuration\_mode.
2. reconfig request is equal to DAU\_1\_MODE.
3. (configuration\_mode is equal to DAU\_N\_MODE) AND (default\_configuration\_mode is equal to DAU\_1\_MODE)

##### 10.4.41.7.16 daugwyapp-ReconfigurationFromDU-LLR-013

Requirement ID: H398-LLD-GWY-FNC-7299

The function shall perform the following

* Set Dau\_excited to NOT of Dau\_excited when the following conditions are satisified:

1. reconfig request is not equal to configuration\_mode.
2. reconfig request is equal to DAU\_1\_MODE.
3. (configuration\_mode is equal to DAU\_N\_MODE) AND (default\_configuration\_mode is equal to DAU\_2\_MODE).

##### 10.4.41.7.17 daugwyapp-ReconfigurationFromDU-LLR-014

Requirement ID: H398-LLD-GWY-FNC-7300

* The function shall perform the following

1. Set reconfiguration state to DAU\_1\_MODE
2. Set source selection to EIU\_1
3. Set mode\_sent to DAU\_1\_MODE
4. Set Dauexcite to NOT of Dauexcite when configuration\_mode is equal to DAU\_2\_MODE

when the below conditions are satisified:

* reconfig request is not equal to configuration\_mode.
* reconfig request is equal to DAU\_1\_MODE.

##### 10.4.41.7.18 daugwyapp-ReconfigurationFromDU-LLR-015

Requirement ID: H398-LLD-GWY-FNC-7301

The function shall perform the following

* Set Dau\_excited to NOT of Dau\_excited when the below conditions are satisified:

1. reconfig request is not equal to configuration\_mode.
2. reconfig request is equal to DAU\_2\_MODE
3. (configuration\_mode is equal to DAU\_N\_MODE) AND (default\_configuration\_mode is equal to DAU\_1\_MODE)

##### 10.4.41.7.19 daugwyapp-ReconfigurationFromDU-LLR-016

Requirement ID: H398-LLD-GWY-FNC-7302

The function shall perform the following

* Do Nothing when the below conditions are satisified:

1. reconfig request is not equal to configuration\_mode.
2. reconfig request is equal to DAU\_2\_MODE
3. (configuration\_mode is equal to DAU\_N\_MODE) AND (default\_configuration\_mode is equal to DAU\_2\_MODE)

##### 10.4.41.7.20 daugwyapp-ReconfigurationFromDU-LLR-017

Requirement ID: H398-LLD-GWY-FNC-7303

The function shall perform the following

* Set reconfiguration state to DAU\_2\_MODE
* Set source selection to EIU\_2
* Set mode\_sent to DAU\_2\_MODE
* Set Dau\_excited to NOT of Dau\_excited when configuration\_mode is equal to DAU\_1\_MODE

when the below conditions are satisified:

1. reconfig request is not equal to configuration\_mode.
2. reconfig request is equal to DAU\_2\_MODE

##### 10.4.41.7.21 daugwyapp-ReconfigurationFromDU-LLR-018

Requirement ID: H398-LLD-GWY-FNC-7304

The function shall perform the following

* Set reconfiguration\_state\_sent to reconfiguration state
* Set source\_selection\_sent to source selection
* Set configuration\_mode to mode\_sent

when reconfig request is not equal to configuration\_mode

##### 10.4.41.7.22 daugwyapp-ReconfigurationFromDU-LLR-026

Requirement ID: H398-LLD-GWY-FNC-7463

The function shall set Master\_reconfig\_fault to TRUE when following conditions are satisfied

1. Mode\_sent is not equal to reconfig request
2. reconfig\_request is not equal to Configuration\_mode

otherwise set Master\_reconfig\_fault to FALSE.

##### 10.4.41.7.23 daugwyapp-ReconfigurationFromDU-LLR-021

Requirement ID: H398-LLD-GWY-FNC-7409

The function shall perform the following

* Set DAU\_reconfig\_discrepancy of tx\_cross\_channel\_rs232 of Xram\_data\_buffer to mode\_sent
* Set crosschannel mode to DAU\_reconfig\_discrepancy of tx\_cross\_channel\_rs232 of Xram\_data\_buffer.

##### 10.4.41.7.24 daugwyapp-ReconfigurationFromDU-LLR-022

Requirement ID: H398-LLD-GWY-FNC-7410

The function shall perform the following

* Set Reconfig\_discrepancy to TRUE When Mode\_sent is not equal to crosschannel mode AND Rs232\_transmit\_data is equal to TRUE OR source selection discrepancy is equal to TRUE.

Otherwise Set Reconfig\_discrepancy to FALSE.

##### 10.4.41.7.25 daugwyapp-ReconfigurationFromDU-LLR-023

Requirement ID: H398-LLD-GWY-FNC-7411

The function shall perform the following

* Set DAU\_reconfig\_discrepancy of tx\_cross\_channel\_rs232 of Xram\_data\_buffer to source\_selection\_sent .
* Set crosschannel mode to DAU\_reconfig\_discrepancy of rx\_cross\_channel\_rs232 of Xram\_data\_buffer.

##### 10.4.41.7.26 daugwyapp-ReconfigurationFromDU-LLR-024

Requirement ID: H398-LLD-GWY-FNC-7412

The function shall perform the following

* Set master\_discrepancy to TRUE When source\_selection\_sent is not equal to crosschannel master AND Rs232\_transmit\_data is equal to TRUE,otherwise Set master\_discrepancy to FALSE.

##### 10.4.41.7.27 daugwyapp-ReconfigurationFromDU-LLR-027

Requirement ID: H398-LLD-GWY-FNC-7472

The funciton shall set Channel\_master to EIU\_1 when (Dau\_excited is equal to TRUE ) AND(by calling the function GpioReadInputDataBit of parameters (M\_GPIOC and M\_GPIOC\_SYS\_SEL) returns the value M\_ZERO)) OR ((Dau\_excited is equal to FALSE ) AND (by calling the function GpioReadInputDataBit of parameters (M\_GPIOC and M\_GPIOC\_SYS\_SEL) returns the value M\_ONE)).

##### 10.4.41.7.28 daugwyapp-ReconfigurationFromDU-LLR-028

Requirement ID: H398-LLD-GWY-FNC-7473

The funciton shall set Channel\_master to EIU\_2 when (Dau\_excited is equal to TRUE ) AND(by calling the function GpioReadInputDataBit of parameters (M\_GPIOC and M\_GPIOC\_SYS\_SEL) returns the value M\_ONE)) OR ((Dau\_excited is equal to FALSE ) AND (by calling the function GpioReadInputDataBit of parameters (M\_GPIOC and M\_GPIOC\_SYS\_SEL) returns the value M\_ZERO)).

##### 10.4.41.7.29 daugwyapp-ReconfigurationFromDU-LLR-029

Requirement ID: H398-LLD-GWY-FNC-7474

The function shall set temporary source selection disc to TRUE when source selection sent is not equal to Channel\_master otherwise set temporary source selection disc to FALSE.

##### 10.4.41.7.30 daugwyapp-ReconfigurationFromDU-LLR-030

Requirement ID: H398-LLD-GWY-FNC-7475

The function shall do the following when temporary source selection disc is not equal to FALSE

1. Pre increment source selection timer when source selection timer is less than multiplication of (M\_APP\_RATE and M\_THREE) otherwise set source selection discrepancy to TRUE.

Otherwise set source selection discrepancy to FALSE.

## 10.5 daugwyapplogic

The daugwyapplogic CSC manages the routines for XRAM for use by the DAU.

### 10.5.1 TransmitGwyPartNums

Low Level Design Details about CSU TransmitGwyPartNums will follow in the sub sections.

#### 10.5.1.1 Brief Description

The TransmitGwyPartNums function transmits gateway partnumbers and CRC to DLU.

#### 10.5.1.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.5.1.3 List of global variables accessed and modified

Accessed : Revno\_id

Appdata\_appdata

Modified : None

#### 10.5.1.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.5.1.5 Return Value

None

#### 10.5.1.6 Other CSUs called by this CSU

A825Xmit

OsTimeDly

#### 10.5.1.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TransmitGwyPartNums.

##### 10.5.1.7.1 daugwyapplogic-TransmitGwyPartNums-LLR-001

Requirement ID: H398-LLD-GWY-FNC-694

The function shall get the revision number from CRC address by setting rev no to the return value of M\_GETREVNO with parameter M\_MEMMAP\_BOOT\_CRC\_ADDR .

* Set lcc of transmit message to NOC
* Set the elements sfid to M\_HOWELL\_DAU\_FID, rsd to M\_ZERO, lcl and pvt to M\_ONE and rci to M\_ZERO of bid of sid of transmit message .
* Set u8\_paysize of transmit message to M\_EIGHT
* Set doc of bid of sid of transmit message to PN\_MESSAGES

##### 10.5.1.7.2 daugwyapplogic-TransmitGwyPartNums-LLR-002

Requirement ID: H398-LLD-GWY-FNC-695

The function shall prepare the transmit message for Gateway Bootloader Application Part number as follows.

* Set all indices of u8\_payload of transmit message to first 8 bytes of rev no

##### 10.5.1.7.3 daugwyapplogic-TransmitGwyPartNums-LLR-003

Requirement ID: H398-LLD-GWY-FNC-696

The function shall

1. transmit the first 8 bytes of Gateway Bootloader Application Part number by calling the function A825Xmit with parameter transmit message.
2. Call OsTimeDly with parameter M\_FIVE.
3. Set doc of bid of sid of transmit message to PN\_MESSAGES + M\_ONE

##### 10.5.1.7.4 daugwyapplogic-TransmitGwyPartNums-LLR-004

Requirement ID: H398-LLD-GWY-FNC-697

The function shall set all the indices of u8\_payload of transmit message to next 8 bytes of rev no

##### 10.5.1.7.5 daugwyapplogic-TransmitGwyPartNums-LLR-005

Requirement ID: H398-LLD-GWY-FNC-698

The function shall do the following

1. transmit the next 8 bytes of Gateway Bootloader Application Part number by calling the function A825Xmit with parameter transmit message.

2. Call OsTimeDly with parameter M\_FIVE.

##### 10.5.1.7.6 daugwyapplogic-TransmitGwyPartNums-LLR-006

Requirement ID: H398-LLD-GWY-FNC-699

The function shall update the transmit message with Gateway Bootloader Application CRC as follows.

* Set doc of bid of sid of transmit message to PN\_MESSAGES + M\_TWO
* Get the crc word present in address M\_MEMMAP\_BOOT\_CRC\_ADDR
* Set 0th index of u8\_payload of transmit message to first lower byte of crc word
* Set 1st index of u8\_payload of transmit message to second byte of crc word from LSB
* Set 2nd index of u8\_payload of transmit message to third byte of crc word from LSB
* Set 3rd index of u8\_payload of transmit message to fourth byte of crc word from LSB
* Set remaining indices (from 4-7) of u8\_payload of transmit message to M\_ZERO

##### 10.5.1.7.7 daugwyapplogic-TransmitGwyPartNums-LLR-007

Requirement ID: H398-LLD-GWY-FNC-700

The function shall do the following

1. Transmit the Gateway Bootloader Application CRC by calling the function A825Xmit with parameter transmit message.

2. Call OsTimeDly with parameter M\_TWO.

##### 10.5.1.7.8 daugwyapplogic-TransmitGwyPartNums-LLR-008

Requirement ID: H398-LLD-GWY-FNC-701

The function shall update the transmit message with first 8 bytes of Gateway Flight Application Part number as follows.

* Set doc of bid of sid of transmit message to PN\_MESSAGES + M\_THREE
* Set all the indices of u8\_payload of transmit message to first 8 bytes of Revision number ID Revno\_id

##### 10.5.1.7.9 daugwyapplogic-TransmitGwyPartNums-LLR-009

Requirement ID: H398-LLD-GWY-FNC-702

The function shall do the following

1. Transmit the first 8 bytes of Gateway Flight Application Part number by calling the function A825Xmit with parameter transmit message.

2. Call OsTimeDly with parameter M\_TWO.

##### 10.5.1.7.10 daugwyapplogic-TransmitGwyPartNums-LLR-010

Requirement ID: H398-LLD-GWY-FNC-703

The function shall update the transmit message with next 8 bytes of Gateway Flight Application Part number as follows.

* Set doc of bid of sid of transmit message to PN\_MESSAGES + M\_FOUR
* Set all the indices of u8\_payload of transmit message to next 8 bytes of Revision number ID Revno\_id

##### 10.5.1.7.11 daugwyapplogic-TransmitGwyPartNums-LLR-011

Requirement ID: H398-LLD-GWY-FNC-704

The function shall do the following

1. Transmit the next 8 bytes of Gateway Flight Application Part number by calling the function A825Xmit with parameter reference to transmit message.

2. Call OsTimeDly with parameter M\_TWO.

##### 10.5.1.7.12 daugwyapplogic-TransmitGwyPartNums-LLR-012

Requirement ID: H398-LLD-GWY-FNC-705

The function shall update the transmit message with Gateway Flight Application CRC as follows.

* Set doc of bid of sid of transmit message to PN\_MESSAGES + M\_FIVE
* Get the crc word present in address M\_MEMMAP\_FLIGHT\_CRC\_ADDR
* Set 0th index of u8\_payload of transmit message to first lower byte of crc word
* Set 1st index of u8\_payload of transmit message to second byte of crc word from LSB
* Set 2nd index of u8\_payload of transmit message to third byte of crc word from LSB
* Set 3rd index of u8\_payload of transmit message to fourth byte of crc word from LSB
* Set remaining indices (from 4-7) of u8\_payload of transmit message to M\_ZERO

##### 10.5.1.7.13 daugwyapplogic-TransmitGwyPartNums-LLR-013

Requirement ID: H398-LLD-GWY-FNC-706

The function shall transmit the Gateway Flight Application CRC by calling the function A825Xmit with parameter transmit message.

##### 10.5.1.7.14 daugwyapplogic-TransmitGwyPartNums-LLR-014

Requirement ID: H398-LLD-GWY-FNC-707

The function shall update the transmit message with first 8 bytes of Gateway ACD Configuration Part number as follows.

* Set doc of bid of sid of transmit message to PN\_MESSAGES + M\_SIX
* Set all indices of u8\_payload of transmit message to first 8 bytes of rev\_no of Appdata\_appdata

##### 10.5.1.7.15 daugwyapplogic-TransmitGwyPartNums-LLR-015

Requirement ID: H398-LLD-GWY-FNC-708

The function shall transmit the first 8 bytes of Gateway ACD Configuration Part number by calling the function A825Xmit with parameter reference to transmit message.

##### 10.5.1.7.16 daugwyapplogic-TransmitGwyPartNums-LLR-016

Requirement ID: H398-LLD-GWY-FNC-709

The function shall update the transmit message with next 8 bytes of Gateway ACD Configuration Part number as follows.

* Set doc of bid of sid of transmit message to PN\_MESSAGES + M\_SEVEN
* Set all indices of u8\_payload of transmit message to next 8 bytes of rev\_no of Appdata\_appdata

##### 10.5.1.7.17 daugwyapplogic-TransmitGwyPartNums-LLR-017

Requirement ID: H398-LLD-GWY-FNC-710

The function shall transmit the next 8 bytes of Gateway ACD Configuration Part number by calling the function A825Xmit with parameter reference to transmit message.

##### 10.5.1.7.18 daugwyapplogic-TransmitGwyPartNums-LLR-018

Requirement ID: H398-LLD-GWY-FNC-711

The function shall update the transmit message with Gateway ACD Configuration CRC as follows.

* Set doc of bid of sid of transmit message to PN\_MESSAGES + M\_EIGHT
* Get the crc word present in address M\_MEMMAP\_DATA\_1\_CRC\_ADDR
* Set 0th index of u8\_payload of transmit message to first lower byte of crc word
* Set 1st index of u8\_payload of transmit message to second byte of crc word from LSB
* Set 2nd index of u8\_payload of transmit message to third byte of crc word from LSB
* Set 3rd index of u8\_payload of transmit message to fourth byte of crc word from LSB
* Set remaining indices (from 4-7) of u8\_payload of transmit message to M\_ZERO

##### 10.5.1.7.19 daugwyapplogic-TransmitGwyPartNums-LLR-019

Requirement ID: H398-LLD-GWY-FNC-712

The function shall do the following:

1. Call ‘A825Xmit’ with parameter reference to transmit message.
2. Set rev no to the return value of M\_GETREVNO with parameter M\_MEMMAP\_LOADER\_DATA\_CRC\_ADDR.
3. Set doc of bid of sid of transmit message to PN\_MESSAGES added to M\_DEC\_FIFTEEN.

##### 10.5.1.7.20 daugwyapplogic-TransmitGwyPartNums-LLR-020

Requirement ID: H398-LLD-GWY-FNC-713

The function shall update the transmit message with first 8 bytes of Gateway Embedded Software loader Application Part number as follows.

* Get the revision number from CRC address by updating rev no with the return value of M\_GETREVNO with parameter M\_MEMMAP\_LOADER\_DATA\_CRC\_ADDR
* Set doc of bid of sid of transmit message to PN\_MESSAGES + M\_DEC\_FIFTEEN
* Set all indices of u8\_payload of transmit message to first 8 bytes of rev no

##### 10.5.1.7.21 daugwyapplogic-TransmitGwyPartNums-LLR-021

Requirement ID: H398-LLD-GWY-FNC-714

The function shall transmit the first 8 bytes of Gateway Embedded Software loader Application Part number by calling the function A825Xmit with parameter reference to transmit message.

##### 10.5.1.7.22 daugwyapplogic-TransmitGwyPartNums-LLR-022

Requirement ID: H398-LLD-GWY-FNC-715

The function shall update the transmit message with next 8 bytes of Gateway Embedded Software loader Application Part number as follows.

* Set doc of bid of sid of transmit message to PN\_MESSAGES + M\_SIXTEEN
* Set all the indices of u8\_payload of transmit message to next 8 bytes of rev no

##### 10.5.1.7.23 daugwyapplogic-TransmitGwyPartNums-LLR-023

Requirement ID: H398-LLD-GWY-FNC-716

The function shall transmit the next 8 bytes of Gateway Embedded Software loader Application Part number by calling the function A825Xmit with parameter reference to transmit message.

##### 10.5.1.7.24 daugwyapplogic-TransmitGwyPartNums-LLR-024

Requirement ID: H398-LLD-GWY-FNC-717

The function shall update the transmit message with Gateway Embedded Software loader Application CRC as follows.

* Set doc of bid of sid of transmit message to PN\_MESSAGES + M\_SEVENTEEN
* Get the crc word present in address M\_MEMMAP\_LOADER\_DATA\_CRC\_ADDR
* Set 0th index of u8\_payload of transmit message to first lower byte of crc word
* Set 1st index of u8\_payload of transmit message to second byte of crc word from LSB
* Set 2nd index of u8\_payload of transmit message to third byte of crc word from LSB
* Set 3rd index of u8\_payload of transmit message to fourth byte of crc word from LSB
* Set remaining indices (from 4-7) of u8\_payload of transmit message to M\_ZERO

##### 10.5.1.7.25 daugwyapplogic-TransmitGwyPartNums-LLR-025

Requirement ID: H398-LLD-GWY-FNC-718

The function shall transmit the Gateway Embedded Software loader Application CRC by calling the function A825Xmit with parameter transmit message.

##### 10.5.1.7.26 daugwyapplogic-TransmitGwyPartNums-LLR-026

Requirement ID: H398-LLD-GWY-FNC-719

The function shall update the transmit message with first 8 bytes of Boot Config Application Part number as follows.

* Get the revision number by updating rev no with the value present in address M\_MEMMAP\_BOOT\_CONFIG\_PN\_ADDR
* Set doc of bid of sid of transmit message to PN\_MESSAGES + M\_NINE
* Set all indices of u8\_payload of transmit message to first 8 bytes of rev no

##### 10.5.1.7.27 daugwyapplogic-TransmitGwyPartNums-LLR-027

Requirement ID: H398-LLD-GWY-FNC-720

The function shall transmit the first 8 bytes of Boot Config Application Part number by calling the function A825Xmit with parameter reference to transmit message.

##### 10.5.1.7.28 daugwyapplogic-TransmitGwyPartNums-LLR-028

Requirement ID: H398-LLD-GWY-FNC-721

The function shall update the transmit message with next 8 bytes of Boot Config Application Part number as follows.

* Set doc of bid of sid of transmit message to PN\_MESSAGES + M\_TEN
* Set all the indices of u8\_payload of transmit message to next 8 bytes of rev no

##### 10.5.1.7.29 daugwyapplogic-TransmitGwyPartNums-LLR-029

Requirement ID: H398-LLD-GWY-FNC-722

The function shall transmit the next 8 bytes of Boot Config Application Part number by calling the function A825Xmit with parameter reference to transmit message.

##### 10.5.1.7.30 daugwyapplogic-TransmitGwyPartNums-LLR-030

Requirement ID: H398-LLD-GWY-FNC-723

The function shall update the transmit message with Boot Config Application CRC as follows.

* Set doc of bid of sid of transmit message to PN\_MESSAGES + M\_ELEVEN
* Get the crc word present in address M\_MEMMAP\_BOOT\_CONFIG\_CRC\_ADDR
* Set 0th index of u8\_payload of transmit message to first lower byte of crc word
* Set 1st index of u8\_payload of transmit message to second byte of crc word from LSB
* Set 2nd index of u8\_payload of transmit message to third byte of crc word from LSB
* Set 3rd index of u8\_payload of transmit message to fourth byte of crc word from LSB
* Set remaining indices (from 4-7) of u8\_payload of transmit message to M\_ZERO

##### 10.5.1.7.31 daugwyapplogic-TransmitGwyPartNums-LLR-031

Requirement ID: H398-LLD-GWY-FNC-724

The function shall call ‘A825Xmit’ with parameter reference to transmit message.

##### 10.5.1.7.32 daugwyapplogic-TransmitGwyPartNums-LLR-032

Requirement ID: H398-LLD-GWY-FNC-725

The function shall update the transmit message with first 8 bytes of Gateway Calibration Application Part number as follows.

* Get the revision number from CRC address by updating rev no with the return value of M\_GETREVNO with parameter M\_MEMMAP\_SEC\_DATA\_CRC\_ADDR
* Set doc of bid of sid of transmit message to PN\_MESSAGES + M\_EIGHTEEN
* Set all indices of u8\_payload of transmit message to first 8 bytes of rev no

##### 10.5.1.7.33 daugwyapplogic-TransmitGwyPartNums-LLR-033

Requirement ID: H398-LLD-GWY-FNC-726

The function shall transmit the first 8 bytes of Gateway Calibration Application Part number by calling the function A825Xmit with parameter transmit message.

##### 10.5.1.7.34 daugwyapplogic-TransmitGwyPartNums-LLR-034

Requirement ID: H398-LLD-GWY-FNC-727

The function shall update the transmit message with next 8 bytes of Gateway Calibration Application Part number as follows.

* Set doc of bid of sid of transmit message to PN\_MESSAGES + M\_NINETEEN
* Set all the indices of u8\_payload of transmit message to next 8 bytes of rev no

##### 10.5.1.7.35 daugwyapplogic-TransmitGwyPartNums-LLR-035

Requirement ID: H398-LLD-GWY-FNC-728

The function shall transmit the next 8 bytes of Gateway Calibration Application Part number by calling the function A825Xmit with parameter transmit message.

##### 10.5.1.7.36 daugwyapplogic-TransmitGwyPartNums-LLR-036

Requirement ID: H398-LLD-GWY-FNC-729

The function shall update the transmit message with Gateway Calibration Application CRC as follows.

* Set doc of bid of sid of transmit message to PN\_MESSAGES + M\_TWENTY
* Get the crc word present in address M\_MEMMAP\_SEC\_DATA\_CRC\_ADDR
* Set 0th index of u8\_payload of transmit message to first lower byte of crc word
* Set 1st index of u8\_payload of transmit message to second byte of crc word from LSB
* Set 2nd index of u8\_payload of transmit message to third byte of crc word from LSB
* Set 3rd index of u8\_payload of transmit message to fourth byte of crc word from LSB
* Set remaining indices (from 4-7) of u8\_payload of transmit message to M\_ZERO

##### 10.5.1.7.37 daugwyapplogic-TransmitGwyPartNums-LLR-037

Requirement ID: H398-LLD-GWY-FNC-730

The function shall transmit the Gateway Calibration Application CRC by calling the function A825Xmit with parameter transmit message.

### 10.5.2 TransmitAnalogPartNums

Low Level Design Details about CSU TransmitAnalogPartNums will follow in the sub sections.

#### 10.5.2.1 Brief Description

The function TransmitAnalogPartNums transmits analog part numbers and CRC to DLU.

#### 10.5.2.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.5.2.3 List of global variables accessed and modified

Accessed : App\_analogboot

App\_analogbootcrc

App\_analogbootconfig

App\_analogbootconfigcrc

App\_analogapp

App\_analogappcrc

App\_analogconfig

App\_analogconfigcrc

App\_analogswl

App\_analogswlcrc

App\_analogcalib

App\_analogcalibcrc

Modified : None

#### 10.5.2.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.5.2.5 Return Value

None

#### 10.5.2.6 Other CSUs called by this CSU

A825Xmit

#### 10.5.2.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TransmitAnalogPartNums.

##### 10.5.2.7.1 daugwyapplogic-TransmitAnalogPartNums-LLR-001

Requirement ID: H398-LLD-GWY-FNC-739

The function shall prepare the transmit message for Analog Bootloader Application Part number as follows.

* Set lcc of transmit message to NOC
* Set the elements sfid to M\_HOWELL\_DAU\_FID, rsd to M\_ZERO, lcl and pvt to M\_ONE and rci to M\_ZERO of bid of sid of transmit message .
* Set u8\_paysize of transmit message to M\_EIGHT
* Set doc of bid of sid of transmit message to PN\_MESSAGES + M\_APPLICATION\_PN\_FORTY\_TWO
* Set all indices of u8\_payload of transmit message to first 8 bytes of App\_analogboot

##### 10.5.2.7.2 daugwyapplogic-TransmitAnalogPartNums-LLR-002

Requirement ID: H398-LLD-GWY-FNC-740

The function shall transmit the first 8 bytes of Analog Bootloader Application Part number by calling the function A825Xmit with parameter transmit message.

##### 10.5.2.7.3 daugwyapplogic-TransmitAnalogPartNums-LLR-003

Requirement ID: H398-LLD-GWY-FNC-741

The function shall update the transmit message with next 8 bytes of Analog Bootloader Application Part number as follows.

* Set doc of bid of sid of transmit message to PN\_MESSAGES + M\_APPLICATION\_PN\_FORTY\_THREE
* Set all the indices of u8\_payload of transmit message to next 8 bytes of App\_analogboot

##### 10.5.2.7.4 daugwyapplogic-TransmitAnalogPartNums-LLR-004

Requirement ID: H398-LLD-GWY-FNC-742

The function shall transmit the next 8 bytes of Analog Bootloader Application Part number by calling the function A825Xmit with parameter transmit message.

##### 10.5.2.7.5 daugwyapplogic-TransmitAnalogPartNums-LLR-005

Requirement ID: H398-LLD-GWY-FNC-743

The function shall update the transmit message with Analog Bootloader Application CRC as follows.

* Set doc of bid of sid of transmit message to PN\_MESSAGES + M\_APPLICATION\_CRC\_FORTY\_FOUR
* Set 0th index of u8\_payload of transmit message to first lower byte of App\_analogbootcrc
* Set 1st index of u8\_payload of transmit message to second byte (from LSB) of App\_analogbootcrc
* Set 2nd index of u8\_payload of transmit message to third byte (from LSB) of App\_analogbootcrc
* Set 3rd index of u8\_payload of transmit message to fourth byte (from LSB) of App\_analogbootcrc
* Set remaining indices (from 4-7) of u8\_payload of transmit message to M\_ZERO

##### 10.5.2.7.6 daugwyapplogic-TransmitAnalogPartNums-LLR-006

Requirement ID: H398-LLD-GWY-FNC-744

The function shall transmit the Analog Bootloader Application CRC by calling the function A825Xmit with parameter transmit message.

##### 10.5.2.7.7 daugwyapplogic-TransmitAnalogPartNums-LLR-007

Requirement ID: H398-LLD-GWY-FNC-745

The function shall update the transmit message with first 8 bytes of Analog Boot Config Application Part number as follows.

* Set doc of bid of sid of transmit message to PN\_MESSAGES + M\_APPLICATION\_PN\_FIFTY\_ONE
* Set all the indices of u8\_payload of transmit message to first 8 bytes of App\_analogbootconfig

##### 10.5.2.7.8 daugwyapplogic-TransmitAnalogPartNums-LLR-008

Requirement ID: H398-LLD-GWY-FNC-746

The function shall transmit the first 8 bytes of Analog Boot Config Application Part number by calling the function A825Xmit with parameter transmit message.

##### 10.5.2.7.9 daugwyapplogic-TransmitAnalogPartNums-LLR-009

Requirement ID: H398-LLD-GWY-FNC-747

The function shall update the transmit message with next 8 bytes of Analog Boot Config Application Part number as follows.

* Set doc of bid of sid of transmit message to PN\_MESSAGES + M\_APPLICATION\_PN\_FIFTY\_TWO
* Set all the indices of u8\_payload of transmit message to next 8 bytes of App\_analogbootconfig

##### 10.5.2.7.10 daugwyapplogic-TransmitAnalogPartNums-LLR-010

Requirement ID: H398-LLD-GWY-FNC-748

The function shall transmit the next 8 bytes of Analog Boot Config Application Part number by calling the function A825Xmit with parameter transmit message.

##### 10.5.2.7.11 daugwyapplogic-TransmitAnalogPartNums-LLR-011

Requirement ID: H398-LLD-GWY-FNC-749

The function shall update the transmit message with Analog Boot Config Application CRC as follows.

* Set doc of bid of sid of transmit message to PN\_MESSAGES + M\_APPLICATION\_CRC\_FIFTY\_THREE
* Set 0th index of u8\_payload of transmit message to first lower byte of App\_analogbootconfigcrc
* Set 1st index of u8\_payload of transmit message to second byte (from LSB) of App\_analogbootconfigcrc
* Set 2nd index of u8\_payload of transmit message to third byte (from LSB) of App\_analogbootconfigcrc
* Set 3rd index of u8\_payload of transmit message to fourth byte (from LSB) of App\_analogbootconfigcrc
* Set remaining indices (from 4-7) of u8\_payload of transmit message to M\_ZERO

##### 10.5.2.7.12 daugwyapplogic-TransmitAnalogPartNums-LLR-012

Requirement ID: H398-LLD-GWY-FNC-750

The function shall transmit the Analog Boot Config Application CRC by calling the function A825Xmit with parameter transmit message.

##### 10.5.2.7.13 daugwyapplogic-TransmitAnalogPartNums-LLR-013

Requirement ID: H398-LLD-GWY-FNC-751

The function shall update the transmit message with first 8 bytes of Analog Flight Application Part number as follows.

* Set doc of bid of sid of transmit message to PN\_MESSAGES + M\_APPLICATION\_PN\_FORTY\_FIVE
* Set all the indices of u8\_payload of transmit message to first 8 bytes of App\_analogapp

##### 10.5.2.7.14 daugwyapplogic-TransmitAnalogPartNums-LLR-014

Requirement ID: H398-LLD-GWY-FNC-752

The function shall transmit the first 8 bytes of Analog Flight Application Part number by calling the function A825Xmit with parameter transmit message.

##### 10.5.2.7.15 daugwyapplogic-TransmitAnalogPartNums-LLR-015

Requirement ID: H398-LLD-GWY-FNC-753

The function shall update the transmit message with next 8 bytes of Analog Flight Application Part number as follows.

* Set doc of bid of sid of transmit message to PN\_MESSAGES + M\_APPLICATION\_PN\_FORTY\_SIX
* Set all the indices of u8\_payload of transmit message to next 8 bytes of App\_analogapp

##### 10.5.2.7.16 daugwyapplogic-TransmitAnalogPartNums-LLR-016

Requirement ID: H398-LLD-GWY-FNC-754

The function shall transmit the next 8 bytes of Analog Flight Application Part number by calling the function A825Xmit with parameter transmit message.

##### 10.5.2.7.17 daugwyapplogic-TransmitAnalogPartNums-LLR-017

Requirement ID: H398-LLD-GWY-FNC-755

The function shall update the transmit message with Analog Flight Application CRC as follows.

* Set doc of bid of sid of transmit message to PN\_MESSAGES + M\_APPLICATION\_CRC\_FORTY\_SEVEN
* Set 0th index of u8\_payload of transmit message to first lower byte of App\_analogappcrc
* Set 1st index of u8\_payload of transmit message to second byte (from LSB) of App\_analogappcrc
* Set 2nd index of u8\_payload of transmit message to third byte (from LSB) of App\_analogappcrc
* Set 3rd index of u8\_payload of transmit message to fourth byte (from LSB) of App\_analogappcrc
* Set remaining indices (from 4-7) of u8\_payload of transmit message to M\_ZERO

##### 10.5.2.7.18 daugwyapplogic-TransmitAnalogPartNums-LLR-018

Requirement ID: H398-LLD-GWY-FNC-756

The function shall transmit the Analog Flight Application CRC by calling the function A825Xmit with parameter transmit message.

##### 10.5.2.7.19 daugwyapplogic-TransmitAnalogPartNums-LLR-019

Requirement ID: H398-LLD-GWY-FNC-757

The function shall update the transmit message with first 8 bytes of Analog ACD Application Part number as follows.

* Set doc of bid of sid of transmit message to PN\_MESSAGES + M\_APPLICATION\_PN\_FORTY\_EIGHT
* Set all the indices of u8\_payload of transmit message to first 8 bytes of App\_analogconfig

##### 10.5.2.7.20 daugwyapplogic-TransmitAnalogPartNums-LLR-020

Requirement ID: H398-LLD-GWY-FNC-758

The function shall transmit the first 8 bytes of Analog ACD Application Part number by calling the function A825Xmit with parameter transmit message.

##### 10.5.2.7.21 daugwyapplogic-TransmitAnalogPartNums-LLR-021

Requirement ID: H398-LLD-GWY-FNC-759

The function shall update the transmit message with next 8 bytes of Analog ACD Application Part number as follows.

* Set doc of bid of sid of transmit message to PN\_MESSAGES + M\_APPLICATION\_PN\_FORTY\_NINE
* Set all the indices of u8\_payload of transmit message to next 8 bytes of App\_analogconfig

##### 10.5.2.7.22 daugwyapplogic-TransmitAnalogPartNums-LLR-022

Requirement ID: H398-LLD-GWY-FNC-760

The function shall transmit the next 8 bytes of Analog ACD Application Part number by calling the function A825Xmit with parameter transmit message.

##### 10.5.2.7.23 daugwyapplogic-TransmitAnalogPartNums-LLR-023

Requirement ID: H398-LLD-GWY-FNC-761

The function shall update the transmit message with Analog ACD Application CRC as follows.

* Set doc of bid of sid of transmit message to PN\_MESSAGES + M\_APPLICATION\_CRC\_FIFTY
* Set 0th index of u8\_payload of transmit message to first lower byte of App\_analogconfigcrc
* Set 1st index of u8\_payload of transmit message to second byte (from LSB) of App\_analogconfigcrc
* Set 2nd index of u8\_payload of transmit message to third byte (from LSB) of App\_analogconfigcrc
* Set 3rd index of u8\_payload of transmit message to fourth byte (from LSB) of App\_analogconfigcrc
* Set remaining indices (from 4-7) of u8\_payload of transmit message to M\_ZERO

##### 10.5.2.7.24 daugwyapplogic-TransmitAnalogPartNums-LLR-024

Requirement ID: H398-LLD-GWY-FNC-762

The function shall transmit the Analog ACD Application CRC by calling the function A825Xmit with parameter transmit message.

##### 10.5.2.7.25 daugwyapplogic-TransmitAnalogPartNums-LLR-025

Requirement ID: H398-LLD-GWY-FNC-763

The function shall update the transmit message with first 8 bytes of Analog embedded software loader Application Part number as follows.

* Set doc of bid of sid of transmit message to PN\_MESSAGES + M\_APPLICATION\_PN\_FIFTY\_SEVEN
* Set all the indices of u8\_payload of transmit message to first 8 bytes of App\_analogswl

##### 10.5.2.7.26 daugwyapplogic-TransmitAnalogPartNums-LLR-026

Requirement ID: H398-LLD-GWY-FNC-764

The function shall transmit the first 8 bytes of Analog embedded software loader Application Part number by calling the function A825Xmit with parameter transmit message.

##### 10.5.2.7.27 daugwyapplogic-TransmitAnalogPartNums-LLR-027

Requirement ID: H398-LLD-GWY-FNC-765

The function shall update the transmit message with next 8 bytes of Analog embedded software loader Application Part number as follows.

* Set doc of bid of sid of transmit message to PN\_MESSAGES + M\_APPLICATION\_PN\_FIFTY\_EIGHT
* Set all the indices of u8\_payload of transmit message to next 8 bytes of App\_analogswl

##### 10.5.2.7.28 daugwyapplogic-TransmitAnalogPartNums-LLR-028

Requirement ID: H398-LLD-GWY-FNC-766

The function shall transmit the next 8 bytes of Analog embedded software loader Application Part number by calling the function A825Xmit with parameter transmit message.

##### 10.5.2.7.29 daugwyapplogic-TransmitAnalogPartNums-LLR-029

Requirement ID: H398-LLD-GWY-FNC-767

The function shall update the transmit message with Analog embedded software loader Application CRC as follows.

* Set doc of bid of sid of transmit message to PN\_MESSAGES + M\_APPLICATION\_CRC\_FIFTY\_NINE
* Set 0th index of u8\_payload of transmit message to first lower byte of App\_analogswlcrc
* Set 1st index of u8\_payload of transmit message to second byte (from LSB) of App\_analogswlcrc
* Set 2nd index of u8\_payload of transmit message to third byte (from LSB) of App\_analogswlcrc
* Set 3rd index of u8\_payload of transmit message to fourth byte (from LSB) of App\_analogswlcrc
* Set remaining indices (from 4-7) of u8\_payload of transmit message to M\_ZERO

##### 10.5.2.7.30 daugwyapplogic-TransmitAnalogPartNums-LLR-030

Requirement ID: H398-LLD-GWY-FNC-768

The function shall transmit the Analog embedded software loader Application CRC by calling the function A825Xmit with parameter transmit message.

##### 10.5.2.7.31 daugwyapplogic-TransmitAnalogPartNums-LLR-031

Requirement ID: H398-LLD-GWY-FNC-769

The function shall update the transmit message with first 8 bytes of Analog Calibration Application Part number as follows.

* Set doc of bid of sid of transmit message to PN\_MESSAGES + M\_APPLICATION\_PN\_SIXTY
* Set all the indices of u8\_payload of transmit message to first 8 bytes of App\_analogcalib

##### 10.5.2.7.32 daugwyapplogic-TransmitAnalogPartNums-LLR-032

Requirement ID: H398-LLD-GWY-FNC-770

The function shall transmit the first 8 bytes of Analog Calibration Application Part number by calling the function A825Xmit with parameter transmit message.

##### 10.5.2.7.33 daugwyapplogic-TransmitAnalogPartNums-LLR-033

Requirement ID: H398-LLD-GWY-FNC-771

The function shall update the transmit message with next 8 bytes of Analog Calibration Application Part number as follows.

* Set doc of bid of sid of transmit message to PN\_MESSAGES + M\_APPLICATION\_PN\_SIXTY\_ONE
* Set all the indices of u8\_payload of transmit message to next 8 bytes of App\_analogcalib

##### 10.5.2.7.34 daugwyapplogic-TransmitAnalogPartNums-LLR-034

Requirement ID: H398-LLD-GWY-FNC-772

The function shall transmit the next 8 bytes of Analog Calibration Application Part number by calling the function A825Xmit with parameter transmit message.

##### 10.5.2.7.35 daugwyapplogic-TransmitAnalogPartNums-LLR-035

Requirement ID: H398-LLD-GWY-FNC-773

The function shall update the transmit message with Analog Calibration Application CRC as follows.

* Set doc of bid of sid of transmit message to PN\_MESSAGES + M\_APPLICATION\_CRC\_SIXTY\_TWO
* Set 0th index of u8\_payload of transmit message to first lower byte of App\_analogcalibcrc
* Set 1st index of u8\_payload of transmit message to second byte (from LSB) of App\_analogcalibcrc
* Set 2nd index of u8\_payload of transmit message to third byte (from LSB) of App\_analogcalibcrc
* Set 3rd index of u8\_payload of transmit message to fourth byte (from LSB) of App\_analogcalibcrc
* Set remaining indices (from 4-7) of u8\_payload of transmit message to M\_ZERO

##### 10.5.2.7.36 daugwyapplogic-TransmitAnalogPartNums-LLR-036

Requirement ID: H398-LLD-GWY-FNC-774

The function shall transmit the Analog Calibration Application CRC by calling the function A825Xmit with parameter transmit message.

### 10.5.3 TransmitDiscretePartNums

Low Level Design Details about CSU TransmitDiscretePartNums will follow in the sub sections.

#### 10.5.3.1 Brief Description

The function TransmitAnalogPartNums transmits discrete part numbers and CRC to DLU.

#### 10.5.3.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.5.3.3 List of global variables accessed and modified

Accessed :App\_discreteboot  
 App\_discretebootconfig  
 App\_discreteapp

App\_discreteconfig  
 App\_discreteswl

Modified :App\_discretebootcrc  
 App\_discretebootconfigcrc

App\_discreteappcrc

App\_discreteconfigcrc

App\_discreteswlcrc

#### 10.5.3.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.5.3.5 Return Value

None

#### 10.5.3.6 Other CSUs called by this CSU

A825Xmit

#### 10.5.3.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TransmitDiscretePartNums.

##### 10.5.3.7.1 daugwyapplogic-TransmitDiscretePartNums-LLR-001

Requirement ID: H398-LLD-GWY-FNC-783

The function shall prepare the transmit message for Discrete Bootloader Application Part number as follows.

* Set lcc of transmit message to NOC
* Set the elements sfid to M\_HOWELL\_DAU\_FID, rsd to M\_ZERO, lcl and pvt to M\_ONE and rci to M\_ZERO of bid of sid of transmit message.
* Set u8\_paysize of transmit message to M\_EIGHT
* Set doc of bid of sid of transmit message to PN\_MESSAGES + M\_TWENTY\_ONE
* Set all indices of u8\_payload of transmit message to first 8 bytes of App\_discreteboot

##### 10.5.3.7.2 daugwyapplogic-TransmitDiscretePartNums-LLR-002

Requirement ID: H398-LLD-GWY-FNC-784

The function shall transmit the first 8 bytes of Discrete Bootloader Application Part number by calling the function A825Xmit with parameter transmit message.

##### 10.5.3.7.3 daugwyapplogic-TransmitDiscretePartNums-LLR-003

Requirement ID: H398-LLD-GWY-FNC-785

The function shall update the transmit message with next 8 bytes of Discrete Bootloader Application Part number as follows.

* Set doc of bid of sid of transmit message to PN\_MESSAGES + M\_TWENTY\_TWO
* Set all the indices of u8\_payload of transmit message to next 8 bytes of App\_discreteboot

##### 10.5.3.7.4 daugwyapplogic-TransmitDiscretePartNums-LLR-004

Requirement ID: H398-LLD-GWY-FNC-786

The function shall transmit the next 8 bytes of Discrete Bootloader Application Part number by calling the function A825Xmit with parameter transmit message.

##### 10.5.3.7.5 daugwyapplogic-TransmitDiscretePartNums-LLR-005

Requirement ID: H398-LLD-GWY-FNC-787

The function shall update the transmit message with Analog Bootloader Application CRC as follows.

* Set doc of bid of sid of transmit message to PN\_MESSAGES + M\_TWENTY\_THREE
* Set 0th index of u8\_payload of transmit message to first lower byte of App\_discretebootcrc
* Set 1st index of u8\_payload of transmit message to second byte (from LSB) of App\_discretebootcrc
* Set 2nd index of u8\_payload of transmit message to third byte (from LSB) of App\_discretebootcrc
* Set 3rd index of u8\_payload of transmit message to fourth byte (from LSB) of App\_discretebootcrc
* Set remaining indices (from 4-7) of u8\_payload of transmit message to M\_ZERO

##### 10.5.3.7.6 daugwyapplogic-TransmitDiscretePartNums-LLR-006

Requirement ID: H398-LLD-GWY-FNC-788

The function shall transmit the Discrete Bootloader Application CRC by calling the function A825Xmit with parameter transmit message.

##### 10.5.3.7.7 daugwyapplogic-TransmitDiscretePartNums-LLR-007

Requirement ID: H398-LLD-GWY-FNC-789

The function shall update the transmit message with first 8 bytes of Discrete Boot Config Application Part number as follows.

* Set doc of bid of sid of transmit message to PN\_MESSAGES + M\_THIRTY
* Set all the indices of u8\_payload of transmit message to first 8 bytes of App\_discretebootconfig

##### 10.5.3.7.8 daugwyapplogic-TransmitDiscretePartNums-LLR-008

Requirement ID: H398-LLD-GWY-FNC-790

The function shall transmit the first 8 bytes of Discrete Boot Config Application Part number by calling the function A825Xmit with parameter transmit message.

##### 10.5.3.7.9 daugwyapplogic-TransmitDiscretePartNums-LLR-009

Requirement ID: H398-LLD-GWY-FNC-791

The function shall update the transmit message with next 8 bytes of Discrete Boot Config Application Part number as follows.

* Set doc of bid of sid of transmit message to PN\_MESSAGES + M\_DEC\_THIRTY\_ONE
* Set all the indices of u8\_payload of transmit message to next 8 bytes of App\_discretebootconfig

##### 10.5.3.7.10 daugwyapplogic-TransmitDiscretePartNums-LLR-010

Requirement ID: H398-LLD-GWY-FNC-792

The function shall transmit the next 8 bytes of Discrete Boot Config Application Part number by calling the function A825Xmit with parameter transmit message.

##### 10.5.3.7.11 daugwyapplogic-TransmitDiscretePartNums-LLR-011

Requirement ID: H398-LLD-GWY-FNC-793

The function shall update the transmit message with Discrete Boot Config Application CRC as follows.

* Set doc of bid of sid of transmit message to PN\_MESSAGES + M\_THIRTY\_TWO
* Set 0th index of u8\_payload of transmit message to first lower byte of App\_discretebootconfigcrc
* Set 1st index of u8\_payload of transmit message to second byte (from LSB) of App\_discretebootconfigcrc
* Set 2nd index of u8\_payload of transmit message to third byte (from LSB) of App\_discretebootconfigcrc
* Set 3rd index of u8\_payload of transmit message to fourth byte (from LSB) of App\_discretebootconfigcrc
* Set remaining indices (from 4-7) of u8\_payload of transmit message to M\_ZERO

##### 10.5.3.7.12 daugwyapplogic-TransmitDiscretePartNums-LLR-012

Requirement ID: H398-LLD-GWY-FNC-794

The function shall transmit the Discrete Boot Config Application CRC by calling the function A825Xmit with parameter transmit message.

##### 10.5.3.7.13 daugwyapplogic-TransmitDiscretePartNums-LLR-013

Requirement ID: H398-LLD-GWY-FNC-795

The function shall update the transmit message with first 8 bytes of Discrete Flight Application Part number as follows.

* Set doc of bid of sid of transmit message to PN\_MESSAGES + M\_TWENTY\_FOUR
* Set all the indices of u8\_payload of transmit message to first 8 bytes of App\_discreteapp

##### 10.5.3.7.14 daugwyapplogic-TransmitDiscretePartNums-LLR-014

Requirement ID: H398-LLD-GWY-FNC-796

The function shall transmit the first 8 bytes of Discrete Flight Application Part number by calling the function A825Xmit with parameter transmit message.

##### 10.5.3.7.15 daugwyapplogic-TransmitDiscretePartNums-LLR-015

Requirement ID: H398-LLD-GWY-FNC-797

The function shall update the transmit message with next 8 bytes of Discrete Flight Application Part number as follows.

* Set doc of bid of sid of transmit message to PN\_MESSAGES + M\_TWENTY\_FIVE
* Set all the indices of u8\_payload of transmit message to next 8 bytes of App\_discreteapp

##### 10.5.3.7.16 daugwyapplogic-TransmitDiscretePartNums-LLR-016

Requirement ID: H398-LLD-GWY-FNC-798

The function shall transmit the next 8 bytes of Discrete Flight Application Part number by calling the function A825Xmit with parameter transmit message.

##### 10.5.3.7.17 daugwyapplogic-TransmitDiscretePartNums-LLR-017

Requirement ID: H398-LLD-GWY-FNC-799

The function shall update the transmit message with Discrete Flight Application CRC as follows.

* Set doc of bid of sid of transmit message to PN\_MESSAGES + M\_TWENTY\_SIX
* Set 0th index of u8\_payload of transmit message to first lower byte of App\_discreteappcrc
* Set 1st index of u8\_payload of transmit message to second byte (from LSB) of App\_discreteappcrc
* Set 2nd index of u8\_payload of transmit message to third byte (from LSB) of App\_discreteappcrc
* Set 3rd index of u8\_payload of transmit message to fourth byte (from LSB) of App\_discreteappcrc
* Set remaining indices (from 4-7) of u8\_payload of transmit message to M\_ZERO

##### 10.5.3.7.18 daugwyapplogic-TransmitDiscretePartNums-LLR-018

Requirement ID: H398-LLD-GWY-FNC-800

The function shall transmit the Discrete Flight Application CRC by calling the function A825Xmit with parameter transmit message.

##### 10.5.3.7.19 daugwyapplogic-TransmitDiscretePartNums-LLR-019

Requirement ID: H398-LLD-GWY-FNC-801

The function shall update the transmit message with first 8 bytes of Discrete ACD Application Part number as follows.

* Set doc of bid of sid of transmit message to PN\_MESSAGES + M\_TWENTY\_SEVEN
* Set all the indices of u8\_payload of transmit message to first 8 bytes of App\_discreteconfig

##### 10.5.3.7.20 daugwyapplogic-TransmitDiscretePartNums-LLR-020

Requirement ID: H398-LLD-GWY-FNC-802

The function shall transmit the first 8 bytes of Discrete ACD Application Part number by calling the function A825Xmit with parameter transmit message.

##### 10.5.3.7.21 daugwyapplogic-TransmitDiscretePartNums-LLR-021

Requirement ID: H398-LLD-GWY-FNC-803

The function shall update the transmit message with next 8 bytes of Discrete ACD Application Part number as follows.

* Set doc of bid of sid of transmit message to PN\_MESSAGES + M\_TWENTY\_EIGHT
* Set all the indices of u8\_payload of transmit message to next 8 bytes of App\_discreteconfig

##### 10.5.3.7.22 daugwyapplogic-TransmitDiscretePartNums-LLR-022

Requirement ID: H398-LLD-GWY-FNC-804

The function shall transmit the next 8 bytes of Discrete ACD Application Part number by calling the function A825Xmit with parameter transmit message.

##### 10.5.3.7.23 daugwyapplogic-TransmitDiscretePartNums-LLR-023

Requirement ID: H398-LLD-GWY-FNC-805

The function shall update the transmit message with Discrete ACD Application CRC as follows.

* Set doc of bid of sid of transmit message to PN\_MESSAGES + M\_TWENTY\_NINE
* Set 0th index of u8\_payload of transmit message to first lower byte of App\_discreteconfigcrc
* Set 1st index of u8\_payload of transmit message to second byte (from LSB) of App\_discreteconfigcrc
* Set 2nd index of u8\_payload of transmit message to third byte (from LSB) of App\_discreteconfigcrc
* Set 3rd index of u8\_payload of transmit message to fourth byte (from LSB) of App\_discreteconfigcrc
* Set remaining indices (from 4-7) of u8\_payload of transmit message to M\_ZERO

##### 10.5.3.7.24 daugwyapplogic-TransmitDiscretePartNums-LLR-024

Requirement ID: H398-LLD-GWY-FNC-806

The function shall transmit the Discrete ACD Application CRC by calling the function A825Xmit with parameter transmit message.

##### 10.5.3.7.25 daugwyapplogic-TransmitDiscretePartNums-LLR-025

Requirement ID: H398-LLD-GWY-FNC-807

The function shall update the transmit message with first 8 bytes of Discrete embedded software loader Application Part number as follows.

* Set doc of bid of sid of transmit message to PN\_MESSAGES + M\_THIRTY\_SIX
* Set all the indices of u8\_payload of transmit message to first 8 bytes of App\_discreteswl

##### 10.5.3.7.26 daugwyapplogic-TransmitDiscretePartNums-LLR-026

Requirement ID: H398-LLD-GWY-FNC-808

The function shall transmit the first 8 bytes of Discrete embedded software loader Application Part number by calling the function A825Xmit with parameter transmit message.

##### 10.5.3.7.27 daugwyapplogic-TransmitDiscretePartNums-LLR-027

Requirement ID: H398-LLD-GWY-FNC-809

The function shall update the transmit message with next 8 bytes of Discrete embedded software loader Application Part number as follows.

* Set doc of bid of sid of transmit message to PN\_MESSAGES + M\_THIRTY\_SEVEN
* Set all the indices of u8\_payload of transmit message to next 8 bytes of App\_discreteswl

##### 10.5.3.7.28 daugwyapplogic-TransmitDiscretePartNums-LLR-028

Requirement ID: H398-LLD-GWY-FNC-810

The function shall transmit the next 8 bytes of Discrete embedded software loader Application Part number by calling the function A825Xmit with parameter transmit message.

##### 10.5.3.7.29 daugwyapplogic-TransmitDiscretePartNums-LLR-029

Requirement ID: H398-LLD-GWY-FNC-811

The function shall update the transmit message with Discrete embedded software loader Application CRC as follows.

* Set doc of bid of sid of transmit message to PN\_MESSAGES + M\_THIRTY\_EIGHT
* Set 0th index of u8\_payload of transmit message to first lower byte of App\_discreteswlcrc
* Set 1st index of u8\_payload of transmit message to second byte (from LSB) of App\_discreteswlcrc
* Set 2nd index of u8\_payload of transmit message to third byte (from LSB) of App\_discreteswlcrc
* Set 3rd index of u8\_payload of transmit message to fourth byte (from LSB) of App\_discreteswlcrc
* Set remaining indices (from 4-7) of u8\_payload of transmit message to M\_ZERO

##### 10.5.3.7.30 daugwyapplogic-TransmitDiscretePartNums-LLR-030

Requirement ID: H398-LLD-GWY-FNC-812

The function shall transmit the Discrete embedded software loader Application CRC by calling the function A825Xmit with parameter transmit message.

### 10.5.4 TransmitSystemCheckSum

Low Level Design Details about CSU TransmitSystemCheckSum will follow in the sub sections.

#### 10.5.4.1 Brief Description

The function TransmitSystemCheckSum transmits System Checksum.

#### 10.5.4.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.5.4.3 List of global variables accessed and modified

None

#### 10.5.4.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.5.4.5 Return Value

None

#### 10.5.4.6 Other CSUs called by this CSU

A825Xmit

HwCopy

CalcSysChecksum

#### 10.5.4.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TransmitSystemCheckSum.

##### 10.5.4.7.1 daugwyapplogic-TransmitSystemCheckSum-LLR-001

Requirement ID: H398-LLD-GWY-FNC-5794

The function shall do the following

1. Set lcc of transmit message to NOC.

2. Set sfid of bid of sid of transmit message to M\_HOWELL\_DAU\_FID.

3. Set rsd of bid of sid of transmit message to M\_ZERO.

4. Set lcl of bid of sid of transmit message to M\_ONE.

5. Set pvt of bid of sid of transmit message to M\_ONE.

6. Set rci of bid of sid of transmit message to M\_ZERO.

7. Set u8\_paysize of transmit message to M\_EIGHT.

8. Set doc of bid of sid of transmit message to the result of addition of PN\_MESSAGES and M\_SIXTY\_FIVE.

9. Set system checksum to the return value of the function CalcSysChecksum.

##### 10.5.4.7.2 daugwyapplogic-TransmitSystemCheckSum-LLR-002

Requirement ID: H398-LLD-GWY-FNC-5795

The function shall do the following

1. Call the function HwCopy with parameters as reference to u8\_payload with index M\_ZERO of transmit message, reference to system checksum and M\_FOUR.

2. Set u8\_payload with index from M\_FOUR to M\_SEVEN of transmit message to M\_ZERO.

3. Call the function A825Xmit with parameter as reference to transmit message.

### 10.5.5 PacketizedDataToDisplay

Low Level Design Details about CSU PacketizedDataToDisplay will follow in the sub sections.

#### 10.5.5.1 Brief Description

The PacketizedDataToDisplay function transmits NVM log data to display unit through A429.

#### 10.5.5.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.5.5.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.5.5.4 Parameter list (Input/Output)

Inputs : T\_UBYTE u8\_chan - cahnnel data infromation

T\_SINT32 i32\_arinc\_word - arinc word information

Outputs : None

#### 10.5.5.5 Return Value

T\_UBYTE - Return the status of transmit FIFO,

return FALSE if transmit FIFO is full,

return TRUE if data is loaded to transmit FIFO successfully.

#### 10.5.5.6 Other CSUs called by this CSU

None

#### 10.5.5.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to PacketizedDataToDisplay.

##### 10.5.5.7.1 daugwyapplogic-PacketizedDataToDisplay-LLR-001

Requirement ID: H398-LLD-GWY-FNC-821

The function shall get the base address of the chip by setting address to sum of (M\_A429\_BASE\_ADDR and (product of u8\_chan and M\_A429\_BLOCK\_SIZE)).

##### 10.5.5.7.2 daugwyapplogic-PacketizedDataToDisplay-LLR-002

Requirement ID: H398-LLD-GWY-FNC-822

The function shall return FALSE when transmit FIFO is full bit i.e, the value at address with offset M\_A429\_STATUS\_REG\_RD which is Bitwise AND with M\_A429\_STATUS\_TX\_FULL\_MASK is not equal to M\_ZERO. Otherwise load transmit FIFO and return TRUE.

##### 10.5.5.7.3 daugwyapplogic-PacketizedDataToDisplay-LLR-003

Requirement ID: H398-LLD-GWY-FNC-823

The function shall load transmit FIFO LSB by setting the lower byte of i32\_arinc\_word AND M\_LOAD\_TRANSMIT\_FIFO\_MASK at address with offset M\_A429\_TX\_FIFO\_LSB and load transmit FIFO MSB by setting the higher byte of i32\_arinc\_word AND M\_LOAD\_TRANSMIT\_FIFO\_MASK at address with offset M\_A429\_TX\_FIFO\_MSB.

### 10.5.6 DateTimeCmp

Low Level Design Details about CSU DateTimeCmp will follow in the sub sections.

#### 10.5.6.1 Brief Description

The DateTimeCmp function compares date and time received from cmu and display unit.

#### 10.5.6.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.5.6.3 List of global variables accessed and modified

Accessed : Cmu\_time\_st

Cmu\_time

Du\_time

Cmu\_date

Du\_date

Cmu\_time\_calc

Rx\_a429  
 Priority

Modified : Cmu\_time\_st

Cmu\_time

Cmu\_date

Rx\_label\_received

Cmu\_time\_stat

Cmu\_time\_calc

Cmu\_date\_stat

#### 10.5.6.4 Parameter list (Input/Output)

Inputs : T\_SINT32 i32\_date\_time - date and time information

T\_UINT8 dt - checks the conditions as zero for real time clock and one for clock date

Outputs : T\_SINT32 i32\_date\_time - date and time information

#### 10.5.6.5 Return Value

None

#### 10.5.6.6 Other CSUs called by this CSU

None

#### 10.5.6.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to DateTimeCmp.

##### 10.5.6.7.1 daugwyapplogic-DateTimeCmp-LLR-001

Requirement ID: H398-LLD-GWY-FNC-832

The function shall copy real time clock time from CMU when dt is equal to M\_ZERO as follows.

* Set Cmu\_time\_stat to the result of (i32\_date\_time right shifted by M\_NINE) bitwise ANDed with M\_THREE.
* Update i32\_date\_time by right shifting it by (M\_32BIT\_ARINCWORD - M\_SEVENTEEN) times
* Set Cmu\_time to i32\_date\_time.
* Set hours of Cmu\_time\_st to (i32\_date\_time right shifted by M\_TWELVE times) Bitwise AND with M\_TIME\_HOURS\_MASK
* Set minutes of Cmu\_time\_st to (i32\_date\_time right shifted by M\_SIX times) Bitwise AND with M\_TIME\_MTS\_SEC\_MASK
* Set seconds of Cmu\_time\_st to (i32\_date\_time Bitwise ANDed with M\_TIME\_MTS\_SEC\_MASK).

##### 10.5.6.7.2 daugwyapplogic-DateTimeCmp-LLR-002

Requirement ID: H398-LLD-GWY-FNC-833

The function shall convert hours and minutes to seconds by setting Cmu\_time\_calc to sum of (hours of Cmu\_time\_st multiplied by product of M\_SIXTY and M\_SIXTY, minutes of Cmu\_time\_st multiplied by M\_SIXTY and seconds of Cmu\_time\_st) and set the Rtc increment fail to TRUE when Cmu\_time\_calc is equal to Cmu\_time\_calc\_temp, otherwise set the Rtc increment fail to FALSE and set the Cmu\_time\_calc\_temp to Cmu\_time\_calc, when dt is equal to M\_ZERO.

##### 10.5.6.7.3 daugwyapplogic-DateTimeCmp-LLR-003

Requirement ID: H398-LLD-GWY-FNC-834

The function shall update the Rx\_label\_received as follows when dt is equal to M\_ZERO AND Du\_time is greater than Cmu\_time\_calc AND status of Rx\_a429 with indices (Priority of index IP\_LABEL\_0150, IP\_LABEL\_0150) is equal to OK

* Init time to TRUE.
* Time counter to M\_ZERO.
* Set Rx\_label\_received with indices (Priority of index IP\_LABEL\_0150, M\_ZERO) to TRUE

when difference between Du\_time and Cmu\_time\_calc is greater than M\_SIX AND less than M\_CMU\_TIME\_VALUE\_86394 AND Time counter is greater than or equal to the result of M\_APP\_RATE multiplied with M\_THREE OR NOT of Inittime is equal to TRUE.

* Otherwise, set Rx\_label\_received with indices (Priority of index IP\_LABEL\_0150, M\_ZERO) to FALSE

##### 10.5.6.7.4 daugwyapplogic-DateTimeCmp-LLR-004

Requirement ID: H398-LLD-GWY-FNC-835

The function shall update the Rx\_label\_received as follows when dt is equal to M\_ZERO AND Du\_time is less than Cmu\_time\_calc AND status of Rx\_a429 with indices (Priority of index IP\_LABEL\_0150, IP\_LABEL\_0150) is equal to OK

* Set Init time to TRUE.
* Set Time counter to M\_ZERO.
* Set Rx\_label\_received with indices (Priority of index IP\_LABEL\_0150, M\_ZERO) to TRUE

when difference between Cmu\_time\_calc and Du\_time is greater than M\_SIX AND less than M\_CMU\_TIME\_VALUE\_86394 AND Time counter is greater than or equal to the result of M\_APP\_RATE multiplied with M\_THREE OR NOT of Init time is equal to TRUE.

* Otherwise, set Rx\_label\_received with indices (Priority of index IP\_LABEL\_0150, M\_ZERO) to FALSE

##### 10.5.6.7.5 daugwyapplogic-DateTimeCmp-LLR-005

Requirement ID: H398-LLD-GWY-FNC-836

The function shall set Rx\_label\_received with indices (Priority of index IP\_LABEL\_0150, M\_ZERO) to FALSE when Du\_time is equal to Cmu\_time and dt is equal to M\_ZERO.

##### 10.5.6.7.6 daugwyapplogic-DateTimeCmp-LLR-006

Requirement ID: H398-LLD-GWY-FNC-837

The function shall copy real time Clock date from CMU when dt is equal to M\_ONE as follows.

* Set Cmu\_date\_stat to the result of (i32\_date\_time right shifted by M\_NINE) bitwise AND with M\_THREE.
* Update i32\_date\_time by right shifting it by (M\_32BIT\_ARINCWORD\_DIS - M\_NINETEEN) times
* Set Cmu\_date to updated i32\_date\_time.

##### 10.5.6.7.7 daugwyapplogic-DateTimeCmp-LLR-007

Requirement ID: H398-LLD-GWY-FNC-838

The function shall update the Rx\_label\_received as follows when dt is equal to M\_ONE

* Set Rx\_label\_received with indices (Priority of index IP\_LABEL\_0260, M\_ONE) to TRUE

when Cmu\_date is not equal to Du\_date AND Cmu\_time\_calc is less than M\_CMU\_TIME\_VALUE\_86390 AND greater than M\_TEN AND status of Rx\_a429 with indices (Priority of index IP\_LABEL\_0260, IP\_LABEL\_0260) is equal to OK.

* Otherwise, set Rx\_label\_received with indices (Priority of index IP\_LABEL\_0260, M\_ONE) to FALSE.

Otherwise, do nothing.

### 10.5.7 DoNvmCRCcheck

Low Level Design Details about CSU DoNvmCRCcheck will follow in the sub sections.

#### 10.5.7.1 Brief Description

This function checks if NVM data in memory is good by computing CRC and checking against store CRC.

#### 10.5.7.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.5.7.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.5.7.4 Parameter list (Input/Output)

None

#### 10.5.7.5 Return Value

None

#### 10.5.7.6 Other CSUs called by this CSU

CrcResetDr

CrcCalcBlockCrc

RterrError

RterrForever

#### 10.5.7.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to DoNvmCRCchec.

##### 10.5.7.7.1 daugwyapplogic-DoNvmCRCcheck-LLR-001

Requirement ID: H398-LLD-GWY-FNC-6219

The function shall do the following

- Call CrcResetDr function

- Set cu8extra bytes to sizeof (T\_NVM\_CMU) bitwise AND M\_BOUNDARY\_MASK.

##### 10.5.7.7.2 daugwyapplogic-DoNvmCRCcheck-LLR-002

Requirement ID: H398-LLD-GWY-FNC-6220

The function shall do the following:

1. if cu8extra bytes is greater than M\_ZERO then ((sizeof(T UINT32) substracted by cu8extra\_bytes) added with sizeof(T\_NVM\_CMU)) is set to nvmdata\_crc\_size

Otherwise, set nvmdata\_crc\_size to sizeof(T\_NVM\_CMU) plus M\_ZERO.

2. Set u32checksum to return value of CrcCalcBlockCrc with parameters as M\_NVM\_STORMEMMAP\_CMU\_E,(nvm\_data\_crc\_size divided by sizeof

unsigned bit 32)

##### 10.5.7.7.3 daugwyapplogic-DoNvmCRCcheck-LLR-003

Requirement ID: H398-LLD-GWY-FNC-6221

The function shall call the RterrError function with parameters (NVM\_DATA\_CRC\_ERR,M\_ZERO, RterrForever) when u32checksum not equal to (sum of M\_MEMMAP\_CMU\_NVM\_STORE and nvmdata\_crc\_size), Otherwise Do nothing.

### 10.5.8 APPu32NOCEx

Low Level Design Details about CSU APPu32NOCEx will follow in the sub sections.

#### 10.5.8.1 Brief Description

The APPu32NOCEx function processes NOC messages received from DLU.

#### 10.5.8.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.5.8.3 List of global variables accessed and modified

Accessed : Cmu\_data\_counter

Modified : Xram\_data\_cmutx\_buffer

Nvm\_cmu\_tx

App\_cmubootcrc

App\_cmubootconfigcrc

App\_cmuappcrc

App\_cmuacdconfigcrc

#### 10.5.8.4 Parameter list (Input/Output)

Inputs : T\_A825\_MSG \*sMess - Pointer to the A825 message

Outputs : None

#### 10.5.8.5 Return Value

T\_UINT32 Returns zero

#### 10.5.8.6 Other CSUs called by this CSU

HwCopy

PacketizedDataToDisplay

DateTimeCmp

TransmitGwyPartNums

TransmitAnalogPartNums

TransmitDiscretePartNums

TransmitSystemCheckSum

#### 10.5.8.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to APPu32NOCEx.

##### 10.5.8.7.1 daugwyapplogic-APPu32NOCEx-LLR-001

Requirement ID: H398-LLD-GWY-FNC-847

The function shall perform the following:

* Call the function HwCopy with parameters as reference to arinca25, reference to u8\_payload[M\_ZERO] of reference to sMess and M\_FOUR
* Call the function PacketizedDataToDisplay with parameters as M\_ZERO and arinca25
* Call the function PacketizedDataToDisplay with parameters as M\_ONE and arinca25

when

a) DOC\_PACKETIZED\_DATA is equals to doc of bid of sid of sMess

b) (M\_PACKETIZED\_DATA\_REQUEST is equals to u8\_payload with index M\_ZERO of sMess is equals to ) OR (M\_EXCEEDANCE\_DATA\_START is lessthan or equals to u8\_payload with index M\_ZERO of sMess) AND M\_EXCEEDANCE\_DATA\_END is greaterthan or equals to u8\_payload with index M\_ZERO of sMess).

##### 10.5.8.7.2 daugwyapplogic-APPu32NOCEx-LLR-002

Requirement ID: H398-LLD-GWY-FNC-848

The function shall do the following

* Call the function HwCopy with parameters as reference to arinca25, reference to u8\_payload with index M\_FOUR of reference to sMess and M\_FOUR
* Call the function PacketizedDataToDisplay with parameters as M\_ZERO and arinca25
* Call the function PacketizedDataToDisplay with parameters as M\_ONE and arinca25

when

a) DOC\_PACKETIZED\_DATA is equals to doc of bid of sid of sMess

b) (M\_PACKETIZED\_DATA\_REQUEST is equals u8\_payload with index M\_FOUR of sMess) OR (M\_EXCEEDANCE\_DATA\_START is lessthan or equals to u8\_payload with index M\_FOUR of sMess) AND (M\_EXCEEDANCE\_DATA\_END is greaterthan or equals to u8\_payload with index M\_FOUR of sMess).

##### 10.5.8.7.3 daugwyapplogic-APPu32NOCEx-LLR-003

Requirement ID: H398-LLD-GWY-FNC-849

The function shall do the following when doc of bid of sid of sMess is DOC\_DATE\_TIME and u8\_payload with index M\_ZERO of sMess is equal to M\_TIME\_LBL\_150\_VAL\_104:

* Call the function HwCopy with parameters as reference to arinca25, reference to u8\_payload with index M\_ZERO of sMess and M\_FOUR
* Call the function DateTimeCmp with parameters as arinca25 and M\_ZERO

##### 10.5.8.7.4 daugwyapplogic-APPu32NOCEx-LLR-004

Requirement ID: H398-LLD-GWY-FNC-850

The function shall do the following when doc of bid of sid of sMess is DOC\_DATE\_TIME and u8\_payload with index M\_FOUR of sMess is equal to M\_TIME\_LBL\_260\_VAL\_176:

* Call the function HwCopy with parameters as reference to arinca25, reference to u8\_payload with index M\_FOUR of sMess and M\_FOUR
* Call the function DateTimeCmp with parameters as arinca25 and M\_ONE

##### 10.5.8.7.5 daugwyapplogic-APPu32NOCEx-LLR-005

Requirement ID: H398-LLD-GWY-FNC-851

The function shall call the following functions when doc of bid of sid of sMess is equal to ACQUIRE\_CRC\_PN

* Call the function TransmitGwyPartNums
* Call the function TransmitAnalogPartNums
* Call the function TransmitDiscretePartNums
* Call the function TransmitSystemCheckSum

##### 10.5.8.7.6 daugwyapplogic-APPu32NOCEx-LLR-008

Requirement ID: H398-LLD-GWY-FNC-6186

The function shall do the following when doc of bid of sid of sMess is equal to CMU\_NVRAM\_START

a) call HwCopy function with parameters reference to temp\_bit\_data, u8\_payload of sMess, M\_EIGHT.

b) set cmu\_nvm\_status to TRUE, call HwCopy function with parameters (reference to Xram\_data\_cmutx\_buffer with index count\_cmutx, u8\_payload of sMess, M\_EIGHT) and Increment cmutx count by M\_EIGHT when M\_ZERO is equals to b\_nvm\_read\_write\_failed of bitwise\_status of pbit of temp\_bit\_data otherwise set cmu\_nvm\_status to FALSE.

##### 10.5.8.7.7 daugwyapplogic-APPu32NOCEx-LLR-006

Requirement ID: H398-LLD-GWY-FNC-852

The function shall perform the following when (cmu\_nvm\_status is TRUE) AND (doc of bid of sid of sMess is greater than CMU\_NVRAM\_START) AND (doc of bid of sid of sMess is less than or equal to CMU\_NVRAM\_END)

* Call the function HwCopy with parameters as reference to Xram\_data\_cmutx\_buffer indexed by cmutx count, u8\_payload of sMess and M\_EIGHT
* Increment cmutx count by M\_EIGHT
* Set G\_ccc to doc of bid of sid of sMess
* Call the function DoNvmCRCcheck. Also set Nvm\_cmu\_tx to M\_MEMMAP\_CMU\_NVM\_STORE when doc of bid of sid of sMess is equal to M\_THREE\_ZERO\_ONE\_FIVE.

##### 10.5.8.7.8 daugwyapplogic-APPu32NOCEx-LLR-007

Requirement ID: H398-LLD-GWY-FNC-853

The function shall perform the following and return M\_ZERO when doc of bid of sid of sMess is greater than or equal to M\_DOC\_CMU\_BOOT\_LOAD\_VAL and doc of bid of sid of sMess is less than or equal to M\_DOC\_CMU\_MAX\_VAL

* Call the function HwCopy with parameters reference to App\_cmubootcrc, u8\_payload of sMess and M\_FOUR when doc of bid of sid of sMess is equal to M\_DOC\_CMU\_BOOT\_LOAD\_VAL
* Call the function HwCopy with parameters reference to App\_cmubootconfigcrc, u8\_payload of sMess and M\_FOUR when doc of bid of sid of sMess is equal to M\_DOC\_CMU\_BOOT\_CFG\_VAL
* Call the function HwCopy with parameters reference to App\_cmuappcrc, u8\_payload of sMess and M\_FOUR when doc of bid of sid of sMess is equal to M\_DOC\_CMU\_FLIGHT\_APP\_VAL
* Call the function HwCopy with parameters reference to App\_cmuacdconfigcrc, u8\_payload of sMess and M\_FOUR when doc of bid of sid of sMess is equal to M\_DOC\_CMU\_AIRCRAFT\_CFG\_VAL
* It does nothing when doc of bid of sid of sMess is other than M\_DOC\_CMU\_BOOT\_LOAD\_VAL, M\_DOC\_CMU\_BOOT\_CFG\_VAL, M\_DOC\_CMU\_FLIGHT\_APP\_VAL and M\_DOC\_CMU\_AIRCRAFT\_CFG\_VAL

##### 10.5.8.7.9 daugwyapplogic-APPu32NOCEx-LLR-009

Requirement ID: H398-LLD-GWY-FNC-6194

The function shall call TransmitFaultstoCmu() function when doc of bid of sid of sMess equal to M\_ACQUIRE\_DOC\_TRANSMIT\_FAULT\_BIT\_VAL is True and then returns M\_ZERO.

### 10.5.9 TransmitFaultstoCmu

Low Level Design Details about CSU TransmitFaultstoCmu will follow in the sub sections.

#### 10.5.9.1 Brief Description

The TransmitFaultstoCmu transmits ETI time in seconds to CMU through Arinc A825.

#### 10.5.9.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.5.9.3 List of Global variables accessed and modified

Accessed : Xram\_data\_buffer

Modified : None

#### 10.5.9.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.5.9.5 Return Value

None

#### 10.5.9.6 Other CSUs called by this CSU

A825Xmit

HwMemset

#### 10.5.9.7 Desciption of List of LLRs allocated

The following section will list the LLRs allocated to TransmitFaultstoCmu.

##### 10.5.9.7.1 daugwyapplogic-TransmitFaultstoCmu-LLR-001

Requirement ID: H398-LLD-GWY-FNC-6203

The function should do the following :

- Set lcc of message\_cmu to NOC.

- Set sfid of bid of sid of message\_cmu to M\_HOWELL\_DAU\_FID.

- Set rsd of bid of sid of message\_cmu to M\_ZERO

- Set lcl of bid of sid of message\_cmu to M\_ONE.

- Set pvt of bid of sid of message\_cmu to M\_ONE.

- Set rci of bid of sid of message\_cmu to M\_ZERO.

##### 10.5.9.7.2 daugwyapplogic-TransmitFaultstoCmu-LLR-002

Requirement ID: H398-LLD-GWY-FNC-6204

The function shall perform the following operations, Process and Transmit Fault Bit Message 1x sec by setting the following:

a) Set doc of bid of sid of message\_cmu to M\_DOC\_TRANSMIT\_FAULT\_BIT\_VAL.

b) Set u8\_paysize of message\_cmu to M\_EIGHT.

c) Call HwMemset with parameters u8\_payload of message\_cmu,M\_ZERO and M\_EIGHT.

##### 10.5.9.7.3 daugwyapplogic-TransmitFaultstoCmu-LLR-003

Requirement ID: H398-LLD-GWY-FNC-6205

The function shall perform the following operations within loop and loop from index i is M\_ZERO to MAX\_ERRORS minus 1, loop from Byte value is M\_ZERO to MAX\_ERRORS minus 1,loop from Bitm value is M\_ZERO to MAX\_ERRORS minus 1 and then perform the following:

a) Loop from index j is M\_ZERO to M\_TWENTY minus 1 and set u8\_payload with index Byte value of message\_cmu to bit shift of Bitm value to left by M\_ONE when error\_count of FAULTS with indices (index i,index j) of Xram\_data\_buffer is greater than M\_ZERO

b) Set Bitm value to M\_ZERO and increment the Byte value when post-increment of Bitm value is greater than or equal to M\_EIGHT.

c) Call A825Xmit function with parameter as reference of message\_cmu.

### 10.5.10 TransmitA429toCMU

Low Level Design Details about CSU TransmitA429toCMU will follow in the sub sections.

#### 10.5.10.1 Brief Description

The TransmitA429toCMU function transmits A429 output data to CMU through arinc A825.

#### 10.5.10.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.5.10.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.5.10.4 Parameter list (Input/Output)

Inputs : T\_UINT8 u8\_label - lable of channel

T\_SINT32 i32\_data - ARINC data

T\_UINT8 u8\_sigbits - Number of Significant bits

T\_UINT16 id - identifier

T\_SINT32 a429\_sdi - source destination index

T\_BOOL sign - significant bits

T\_DATA\_STAT status - status of data

Outputs : T\_SINT32 i32\_data - ARINC data

T\_UINT8 u8\_sigbits - Number of Significant bits

T\_SINT32 a429\_sdi - source destination index

#### 10.5.10.5 Return Value

T\_UINT8 - Returns TRUE

#### 10.5.10.6 Other CSUs called by this CSU

HwCopy

A825Xmit

HwAbs32

#### 10.5.10.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TransmitA429toCMU.

##### 10.5.10.7.1 daugwyapplogic-TransmitA429toCMU-LLR-001

Requirement ID: H398-LLD-GWY-FNC-862

The function shall place SDI (RCI + 1) to data and then move first set of bytes to be used in

SSM area as below when u8\_sigbits is greater than M\_TWENTY

* Update i32\_data by left shifting it by M\_TWO times and Bitwise OR with a429\_sdi
* The move bits is extracted by performing Bitwise AND of 32\_data with M\_MOVE\_BITS\_MASK\_VAL
* Update the move bits by right shifting it by M\_TWELVE times

##### 10.5.10.7.2 daugwyapplogic-TransmitA429toCMU-LLR-002

Requirement ID: H398-LLD-GWY-FNC-863

The function shall do the following when u8\_sigbits is not greater than M\_TWENTY

* Set i32\_data to (i32\_data is left shifted by return value of function HwAbs32 which is called with the parameters M\_TWENTY subtracted by u8\_sigbits) which is Bitwise OR with (id left shifted by M\_TWO) which is again Bitwise OR with a429\_sdi
* Set u8\_sigbits to u8\_sigbits added to (M\_TWENTY subtracted by u8\_sigbits).

##### 10.5.10.7.3 daugwyapplogic-TransmitA429toCMU-LLR-003

Requirement ID: H398-LLD-GWY-FNC-864

The function shall set the 32-bit ARINC word as below when sign is TRUE and status is OK

* 32-bit ARINC word is set to the result of performing Bitwise OR between M\_EXTRACT\_SSM and u8\_label
* 32-bit ARINC word is Bitwise OR with (i32\_data left shifted by (M\_THIRTYONE - u8\_sigbits)) and the result is stored in 32-bit ARINC word

##### 10.5.10.7.4 daugwyapplogic-TransmitA429toCMU-LLR-004

Requirement ID: H398-LLD-GWY-FNC-865

The function shall place moved bits into SSM location as below when sign is FALSE and u8\_sigbits is greater than M\_TWENTY and status is OK

* 32-bit ARINC word is set to the result of performing Bitwise OR between move bits and u8\_label
* 32-bit ARINC word is Bitwise OR with (i32\_data left shifted by M\_ELEVEN) and the result is stored in 32-bit ARINC word

##### 10.5.10.7.5 daugwyapplogic-TransmitA429toCMU-LLR-005

Requirement ID: H398-LLD-GWY-FNC-866

The function shall set the 32-bit ARINC word as below when status is CHAN\_ERR

* 32-bit ARINC word is set to the result of performing Bitwise OR between M\_ARINC\_WORD\_CHAN\_ERR\_MASK and u8\_label
* 32-bit ARINC word is Bitwise OR with (i32\_data left shifted by (M\_THIRTYONE - u8\_sigbits)) and the result is stored in 32-bit ARINC word

##### 10.5.10.7.6 daugwyapplogic-TransmitA429toCMU-LLR-006

Requirement ID: H398-LLD-GWY-FNC-867

The function shall set the 32-bit ARINC word as below when status is MISSING

* 32-bit ARINC word is set to the result of performing Bitwise OR between M\_ARINC\_WORD\_MISSING\_MASK and u8\_label
* 32-bit ARINC word is Bitwise OR with (i32\_data left shifted by (M\_THIRTYONE - u8\_sigbits)) and the result is stored in 32-bit ARINC word

##### 10.5.10.7.7 daugwyapplogic-TransmitA429toCMU-LLR-007

Requirement ID: H398-LLD-GWY-FNC-868

The function shall set the 32-bit ARINC word as below when status is FUNC\_TEST

* 32-bit ARINC word is set to the result of performing Bitwise OR between M\_ARINC\_WORD\_FUNC\_TEST\_MASK and u8\_label
* 32-bit ARINC word is Bitwise OR with (i32\_data left shifted by (M\_THIRTYONE - u8\_sigbits)) and the result is stored in 32-bit ARINC word

##### 10.5.10.7.8 daugwyapplogic-TransmitA429toCMU-LLR-008

Requirement ID: H398-LLD-GWY-FNC-869

The function shall place moved bits into SSM location as below when sign is FALSE and status is OK and u8\_sigbits is not greater than M\_TWENTY

* 32-bit ARINC word is set to the result of performing Bitwise OR between M\_A429\_NORMAL\_OP and u8\_label
* 32-bit ARINC word is Bitwise OR with (i32\_data left shifted by (M\_THIRTYONE - u8\_sigbits)) and the result is stored in 32-bit ARINC word.

##### 10.5.10.7.9 daugwyapplogic-TransmitA429toCMU-LLR-009

Requirement ID: H398-LLD-GWY-FNC-870

The function shall prepare the message to be transmitted as below

* Set lcc of CMU message to NOC
* Set sfid to M\_HOWELL\_DAU\_FID, rsd to M\_ZERO, lcl and pvt to M\_ONE and doc to DOC\_A429 of bid of sid of CMU message
* Set u8\_paysize of message to M\_EIGHT.

##### 10.5.10.7.10 daugwyapplogic-TransmitA429toCMU-LLR-010

Requirement ID: H398-LLD-GWY-FNC-871

The function shall copy the 32-bit ARINC word to temporary 32-bit ARINC word, set pay position to TRUE and returns TRUE when pay position is FALSE.

##### 10.5.10.7.11 daugwyapplogic-TransmitA429toCMU-LLR-011

Requirement ID: H398-LLD-GWY-FNC-872

The function shall transmit the prepared CMU message as below and returns TRUE when pay position is TRUE.

* Call the function HwCopy with parameters (reference to 0th index of u8\_payload of CMU message, reference to temporary 32-bit ARINC word and M\_FOUR)
* Call the function HwCopy with parameters (reference to 4th index of u8\_payload of CMU message, reference to 32-bit ARINC word and M\_FOUR)
* Call the function A825Xmit with parameter reference to CMU message
* Set temporary 32-bit ARINC word to M\_ZERO and pay position to FALSE.

### 10.5.11 InputLabelTx

Low Level Design Details about CSU InputLabelTx will follow in the sub sections.

#### 10.5.11.1 Brief Description

The InputLabelTx function transmits A429 Input data to CMU through arinc A825.

#### 10.5.11.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.5.11.3 List of global variables accessed and modified

Accessed : None

Modified : Rx\_label\_received

#### 10.5.11.4 Parameter list (Input/Output)

Inputs : T\_A429\_CHAN chan - A429 channel number

T\_UINT8 label\_index - Label index

T\_SINT32 arinc\_word - Arinc Word

Outputs : None

#### 10.5.11.5 Return Value

None

#### 10.5.11.6 Other CSUs called by this CSU

HwCopy

A825Xmit

#### 10.5.11.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to InputLabelTx.

##### 10.5.11.7.1 daugwyapplogic-InputLabelTx-LLR-001

Requirement ID: H398-LLD-GWY-FNC-881

The function shall set the Rx\_label\_received as below based on label\_index

* Set Rx\_label\_received of indices (chan, M\_TWO) to TRUE when label\_index is IP\_LABEL\_0306\_1
* Set Rx\_label\_received of indices (chan, M\_THREE) to TRUE when label\_index is IP\_LABEL\_0306\_2
* Set Rx\_label\_received of indices (chan, M\_FOUR) to TRUE when label\_index is IP\_LABEL\_0307
* Returns from the function when label\_index is other than IP\_LABEL\_0306\_1, IP\_LABEL\_0306\_2 and IP\_LABEL\_0307.

### 10.5.12 TxBNRLabelsToCMU

Low Level Design Details about CSU TxBNRLabelsToCMU will follow in the sub sections.

#### 10.5.12.1 Brief Description

The TxBNRLabelsToCMU function transmits Afcs axis to CMU.

#### 10.5.12.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.5.12.3 List of global variables accessed and modified

Accessed : Cmdownload\_lp\_cmd\_data

Tx\_a429\_bnr

Transmit\_labels\_bnr

Modified : None

#### 10.5.12.4 Parameter list (Input/Output)

Inputs : T\_UINT8 loop\_i - Number of Transmit BNR labels to CMU.

Outputs : None

#### 10.5.12.5 Return Value

None

#### 10.5.12.6 Other CSUs called by this CSU

TransmitA429toCMU

#### 10.5.12.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TxBNRLabelsToCMU.

##### 10.5.12.7.1 daugwyapplogic-TxBNRLabelsToCMU-LLR-001

Requirement ID: H398-LLD-GWY-FNC-892

The function shall do the following

1. Set label\_i to Transmit\_labels\_bnr with index label\_i.

2. Return from the function if resolution of A429out\_bnr with index label\_i of Cmdownload\_lp\_cmd\_data is equal to M\_ZERO OR loop\_i is greater than or equal to M\_NO\_TX\_LABEL\_CMU\_BNR.

##### 10.5.12.7.2 daugwyapplogic-TxBNRLabelsToCMU-LLR-002

Requirement ID: H398-LLD-GWY-FNC-893

The function shall call function TransmitA429toCMU with parameters label of (A429out\_bnr with index (Transmit\_labels\_bnr with index current loop)) of Cmdownload\_lp\_cmd\_data, data of (Tx\_a429\_bnr with index (Transmit\_labels\_bnr with index current loop)), offsetValue of (A429out\_bnr with index (Transmit\_labels\_bnr with index current loop)) of Cmdownload\_lp\_cmd\_data, resolution of (A429out\_bnr with index (Transmit\_labels\_bnr with index current loop)) of Cmdownload\_lp\_cmd\_data, sig\_bits of (A429out\_bnr with index (Transmit\_labels\_bnr with index current loop)) of Cmdownload\_lp\_cmd\_data, M\_DEC\_ZERO, id of (Tx\_a429\_bnr with index (Transmit\_labels\_bnr with index current loop)), signBit of (A429out\_bnr with index (Transmit\_labels\_bnr with index current loop)) of Cmdownload\_lp\_cmd\_data, status of (Tx\_a429\_bnr with index (Transmit\_labels\_bnr with index current loop)).

### 10.5.13 TxDDWLabelsToCMU

Low Level Design Details about CSU TxDDWLabelsToCMU will follow in the sub sections.

#### 10.5.13.1 Brief Description

The TxDDWLabelsToCMU function transmits Afcs axis to CMU.

#### 10.5.13.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.5.13.3 List of global variables accessed and modified

Accessed : Cmdownload\_lp\_cmd\_data

Tx\_a429\_ddw

Transmit\_labels\_ddw

Modified : None

#### 10.5.13.4 Parameter list (Input/Output)

Inputs : T\_UINT8 loop\_i - Number of Transmit DDW labels to CMU.

Outputs : None

#### 10.5.13.5 Return Value

None

#### 10.5.13.6 Other CSUs called by this CSU

TransmitA429toCMU

#### 10.5.13.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TxDDWLabelsToCMU.

##### 10.5.13.7.1 daugwyapplogic-TxDDWLabelsToCMU-LLR-001

Requirement ID: H398-LLD-GWY-FNC-5804

The function shall do the following,

1. Set status to status of Tx\_a429\_ddw with index (Transmit\_labels\_ddw with index loop\_j).

2. Set label i to Transmit\_labels\_ddw with index loop\_j.

3. Return from the function when ((M\_ZERO is equal to resolution of A429out\_ddw with index lable i of Cmdownload\_lp\_cmd\_data by calling the macro M\_FP\_COMPARE)

OR (loop j is greater than or equal to M\_NO\_TX\_LABEL\_CMU\_DDW)).

##### 10.5.13.7.2 daugwyapplogic-TxDDWLabelsToCMU-LLR-002

Requirement ID: H398-LLD-GWY-FNC-5805

The function shall do the following,

1. Set status to CHAN\_ERR if status is OK.

2. Set status to OK if status is CHAN\_ERR.

3. Call function TransmitA429toCMU with parameters label of (A429out\_ddw with index (Transmit\_labels\_ddw with index current loop)) of Cmdownload\_lp\_cmd\_data, data of (Tx\_a429\_ddw with index (Transmit\_labels\_ddw with index current loop)), offsetValue of (A429out\_ddw with index (Transmit\_labels\_ddw with index current loop)) of Cmdownload\_lp\_cmd\_data, resolution of (A429out\_ddw with index (Transmit\_labels\_ddw with index current loop)) of Cmdownload\_lp\_cmd\_data, sig\_bits of (A429out\_ddw with index (Transmit\_labels\_ddw with index current loop)) of Cmdownload\_lp\_cmd\_data, M\_DEC\_ZERO, id of (Tx\_a429\_ddw with index (Transmit\_labels\_ddw with index current loop)), signBit of (A429out\_ddw with index (Transmit\_labels\_ddw with index current loop)) of Cmdownload\_lp\_cmd\_data, status.

### 10.5.14 RxLabelsToCMU

Low Level Design Details about CSU RxLabelsToCMU will follow in the sub sections.

#### 10.5.14.1 Brief Description

The RxLabelsToCMU function recives Afcs axis to CMU.

#### 10.5.14.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.5.14.3 List of global variables accessed and modified

Accessed : Rx\_a429

A429\_aas\_chans

Rx\_label\_received

Transmit\_labels\_ip

Priority

Modified : None

#### 10.5.14.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.5.14.5 Return Value

None

#### 10.5.14.6 Other CSUs called by this CSU

TransmitA429toCMU

#### 10.5.14.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to RxLabelsToCMU.

##### 10.5.14.7.1 daugwyapplogic-RxLabelsToCMU-LLR-001

Requirement ID: H398-LLD-GWY-FNC-902

The function shall loop from M\_ZERO to M\_NO\_TX\_LABEL\_CMU\_IP minus 1 to check the status and type of all the labels as below

1. Do nothing when status of Rx\_a429 with indices (Priority with index (Transmit\_labels\_ip with index (loop counter))), (Transmit\_labels\_ip with index (loop counter)) is MISSING
2. Do the following when status of Rx\_a429 with indices (Priority with index (Transmit\_labels\_ip with index (loop counter))), (Transmit\_labels\_ip with index (loop counter)) is not MISSING

* Set the value of status from status of Rx\_a429 with indices (Priority with index (Transmit\_labels\_ip with index (loop counter))), (Transmit\_labels\_ip with index (loop counter))
* Set status to CHAN\_ERR when status is OK and type of A429\_aas\_chans with indices (Priority with index (Transmit\_labels\_ip with index (loop counter))), (Transmit\_labels\_ip (loop counter)) is DIS
* Set status to OK when status is CHAN\_ERR and type of A429\_aas\_chans with indices (Priority with index (Transmit\_labels\_ip with index (loop counter))), (Transmit\_labels\_ip (loop counter)) is DIS
* Decrement u8\_sigbits of A429\_aas\_chans with indices (Priority with index (Transmit\_labels\_ip with index (loop counter))), (Transmit\_labels\_ip (loop counter)) by M\_ONE and store it in significant bits when type of A429\_aas\_chans with indices (Priority with index (Transmit\_labels\_ip with index (loop counter))), (Transmit\_labels\_ip (loop counter)) is DIS and status is other than OK or CHAN\_ERR
* Otherwise, set significant bits to u8\_sigbits of A429\_aas\_chans with indices (Priority with index (Transmit\_labels\_ip with index (loop counter))), (Transmit\_labels\_ip (loop counter)).

##### 10.5.14.7.2 daugwyapplogic-RxLabelsToCMU-LLR-002

Requirement ID: H398-LLD-GWY-FNC-903

The function shall loop through all the labels from M\_ZERO to M\_NO\_TX\_LABEL\_CMU\_IP to transmit the received Input Label and do the following when status of Rx\_a429 with indices (Priority with index (Transmit\_labels\_ip with index (loop counter))), (Transmit\_labels\_ip with index (loop counter)) is not MISSING

* Do nothing when i32\_scalefactor of A429\_aas\_chans with indices (Priority with index (Transmit\_labels\_ip with index (loop counter))), (Transmit\_labels\_ip (loop counter)) is M\_ZERO
* Otherwise, call function TransmitA429toCMU with parameters u8\_label of A429\_aas\_chans with indices (Priority with index (Transmit\_labels\_ip with index (loop counter))), (Transmit\_labels\_ip (loop counter)), data of Rx\_a429 with indices (Priority with index (Transmit\_labels\_ip with index (loop counter))), (Transmit\_labels\_ip with index (loop counter)), default\_value of A429\_aas\_chans with indices (Priority with index (Transmit\_labels\_ip with index (loop counter))), (Transmit\_labels\_ip (loop counter)), i32\_scalefactor of A429\_aas\_chans with indices (Priority with index (Transmit\_labels\_ip with index (loop counter))), (Transmit\_labels\_ip (loop counter)), significant bits, M\_DEC\_ZERO, id of Rx\_a429 with indices (Priority with index (Transmit\_labels\_ip with index (loop counter))), (Transmit\_labels\_ip with index (loop counter)), sign\_bit of A429\_aas\_chans with indices (Priority with index (Transmit\_labels\_ip with index (loop counter))), (Transmit\_labels\_ip (loop counter)), status. Also set Rx\_label\_received with indices (Priority with index(Transmit\_labels\_ip with index(loop counter))), (loop counter) to FALSE when Rx\_label\_received with indices (Priority with index(Transmit\_labels\_ip with index(loop counter))), (loop counter) is TRUE.

### 10.5.15 NonLabelParametersToCMU

Low Level Design Details about CSU NonLabelParametersToCMU will follow in the sub sections.

#### 10.5.15.1 Brief Description

The NonLabelParametersToCMU function transmits Non lable parameters to CMU.

#### 10.5.15.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.5.15.3 List of global variables accessed and modified

Accessed : Afcs\_axis

Shutdown\_ex\_cas

Filter\_clog

Force\_channel\_change\_flag

Modified : None

#### 10.5.15.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.5.15.5 Return Value

None

#### 10.5.15.6 Other CSUs called by this CSU

A825Xmit

#### 10.5.15.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to NonLabelParametersToCMU.

##### 10.5.15.7.1 daugwyapplogic-NonLabelParametersToCMU-LLR-001

Requirement ID: H398-LLD-GWY-FNC-912

The function shall prepare the message to be transmitted as below

* Set lcc of NVM message to NOC
* Set sfid to M\_HOWELL\_DAU\_FID, rsd to M\_ZERO, lcl and pvt to M\_ONE and doc to DOC\_PARAMETER\_TO\_CMU of bid of sid of NVM message
* Set u8\_paysize of message to M\_EIGHT.
* Increment Time counter by one if Time counter is less than or equal to the result of multiplication of M\_APP\_RATE and M\_THREE.

##### 10.5.15.7.2 daugwyapplogic-NonLabelParametersToCMU-LLR-002

Requirement ID: H398-LLD-GWY-FNC-913

The function shall transmit the prepared message as below

* Loop from M\_ZERO to NUM\_OF\_CAS times and set special exceedance value to the OR of special exceedance value with the result of Shutdown\_ex\_cas with index loop\_ex left shifted by current loop times.
* Update u8\_payload with index (M\_ZERO) of NVM message to u8\_payload with index (M\_ZERO) OR the result of ((Filter\_clog left shifted by M\_ONE) bitwise ANDed with M\_HYD\_FILTER\_BIT\_TWO\_MASK).
* Update u8\_payload with index (M\_ZERO) of NVM message to the OR of 0th index of u8\_payload of NVM message with the result of ((Force\_channel\_change\_flag left shifted by M\_TWO) bitwise ANDed with M\_HEX\_FOUR).
* Set u8\_payload with index (M\_ONE) of NVM message to special exceedance value.
* Set temparary value to 0th index of u8\_payload of NVM message , special exceedance value to 1st index of u8\_payload of NVM message and transmit the prepared NVM message by calling the function A825Xmit with parameter NVM message when 0th index of u8\_payload of NVM message is not equal to temparary value OR special exceedance value is not equal to 1st index of u8\_payload of NVM message.

## 10.6 daugwyarinc825

The daugwyarinc825 CSC implements the A825 communications portion.

### 10.6.1 RxQueueRemove

Low Level Design Details about CSU RxQueueRemove will follow in the sub sections.

#### 10.6.1.1 Brief Description

The RxQueueRemove function retrieves a message from the specified receive queue and decompose the CAN message into an ARINC 825 message.

#### 10.6.1.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.6.1.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.6.1.4 Parameter list (Input/Output)

Inputs : T\_CAN\_RX\_QUEUE \*ps\_which\_q - The receive queue to retrieve from

Outputs : T\_CAN\_RX\_QUEUE \*ps\_which\_q - The receive queue to retrieve from

T\_A825\_MSG \*ps\_destination - Destination to place the retrieved message

#### 10.6.1.5 Return Value

T\_A825\_REC\_RESULT -Returns status

#### 10.6.1.6 Other CSUs called by this CSU

CanItConfig

#### 10.6.1.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to RxQueueRemove.

##### 10.6.1.7.1 daugwyarinc825-RxQueueRemove-LLR-001

Requirement ID: H398-LLD-GWY-FNC-923

The function shall return A825\_RX\_QUEUE\_EMPTY when cntr of ps\_which\_q is equal to M\_ZERO,otherwise do nothing.

##### 10.6.1.7.2 daugwyarinc825-RxQueueRemove-LLR-002

Requirement ID: H398-LLD-GWY-FNC-924

The function shall call function ‘CanItConfig’ with parameter (M\_CANX, M\_CAN\_IT\_FMP0, DISABLE) to disable the Can Receiver.

##### 10.6.1.7.3 daugwyarinc825-RxQueueRemove-LLR-003

Requirement ID: H398-LLD-GWY-FNC-925

The function shall retrieve the CAN message from the receive queue.

##### 10.6.1.7.4 daugwyarinc825-RxQueueRemove-LLR-004

Requirement ID: H398-LLD-GWY-FNC-926

The function shall

1. Decrement message counter as one message is read from the queue (i.e Set cntr of ps\_which\_q to (cntr of ps\_which\_q subtracted by M\_ONE)).
2. Set head of ps\_which\_q to (head of ps\_which\_q added to M\_ONE).

##### 10.6.1.7.5 daugwyarinc825-RxQueueRemove-LLR-005

Requirement ID: H398-LLD-GWY-FNC-927

The function shall set head of ps\_which\_q to buff of ps\_which\_q when head of ps\_which\_q is equal to head of ps\_which\_q Bitwise AND with buff of ps\_which\_q with size M\_CAN\_R\_Q\_SIZE, otherwise do nothing.

##### 10.6.1.7.6 daugwyarinc825-RxQueueRemove-LLR-006

Requirement ID: H398-LLD-GWY-FNC-928

The function shall call function ‘CanItConfig’ with parameter (M\_CANX, M\_CAN\_IT\_FMP0, ENABLE) to enable the CAN Receiver.

##### 10.6.1.7.7 daugwyarinc825-RxQueueRemove-LLR-007

Requirement ID: H398-LLD-GWY-FNC-929

The function shall set lcc of arinc\_message to ext\_id of can\_message right shifted by M\_LCC\_LOC bitwise AND with M\_LCC\_MSK.

##### 10.6.1.7.8 daugwyarinc825-RxQueueRemove-LLR-008

Requirement ID: H398-LLD-GWY-FNC-930

The function shall translate CAN MESSAGE to ARINC 825 Broadcast MESSAGE when the message is a broadcast message i.e. lcc of ARINC 825 message is NOC.

- (can message extended identifier bit shifted to right by M\_RCI\_LOC) bitwise AND with M\_RCI\_MSK)

- rci [bit 0...1]

- (can message extended identifier bit shifted to right by M\_DOC\_LOC) bitwise AND with M\_DOC\_MSK)

- doc [bit 2 ... 15]

- (can message extended identifier bit shifted to right by M\_PVT\_LOC) bitwise AND with M\_PVT\_MSK)

- pvt [bit 16]

- (can message extended identifier bit shifted to right by M\_LCL\_LOC) bitwise AND with M\_LCL\_MSK)

- lcl [bit 17]

- (can message extended identifier bit shifted to right by M\_RSD\_LOC) bitwise AND with M\_RSD\_MSK)

- rsd [bit 18]

- (can message extended identifier bit shifted to right by M\_SRC\_FID\_LOC) bitwise AND with M\_SRC\_FID\_MSK)

- sfid [bit 19 ... 25]

-Other bits to spare i.e. ZERO.

##### 10.6.1.7.9 daugwyarinc825-RxQueueRemove-LLR-009

Requirement ID: H398-LLD-GWY-FNC-931

The function shall translate CAN MESSAGE to ARINC 825 Peer to Peer MESSAGE when the message is a peer to peer message i.e. lcc of ARINC 825 message is NSC.

- (can message extended identifier bit shifted to right by M\_RCI\_LOC) bitwise AND with M\_RCI\_MSK)

- rci [bit 0...1]

- (can message extended identifier bit shifted to right by M\_SID\_LOC) bitwise AND with M\_SID\_MSK)

- sid [bit 2 ... 8]

- (can message extended identifier bit shifted to right by M\_SER\_FID\_LOC) bitwise AND with M\_SER\_FID\_MSK)

- sfid [bit 9 ... 15]

- (can message extended identifier bit shifted to right by M\_PVT\_LOC) bitwise AND with M\_PVT\_MSK)

- pvt [bit 16]

- (can message extended identifier bit shifted to right by M\_LCL\_LOC) bitwise AND with M\_LCL\_MSK)

- lcl [bit 17]

- (can message extended identifier bit shifted to right by M\_SMT\_LOC) bitwise AND with M\_SMT\_MSK)

- smt [bit 18]

- (can message extended identifier bit shifted to right by M\_CFID\_LOC) bitwise AND with M\_CFID\_MSK)

- cfid [bit 19 ... 25]

-Other bits to spare i.e. ZERO.

##### 10.6.1.7.10 daugwyarinc825-RxQueueRemove-LLR-010

Requirement ID: H398-LLD-GWY-FNC-932

The function shall set u8\_paysize of arinc message to dlc of CAN message.

##### 10.6.1.7.11 daugwyarinc825-RxQueueRemove-LLR-011

Requirement ID: H398-LLD-GWY-FNC-933

This function shall set u8\_paysize of arinc message to M\_BYTES\_OF\_PYLD, when u8\_paysize of arinc message is greater than M\_BYTES\_OF\_PYLD otherwise do nothing.

##### 10.6.1.7.12 daugwyarinc825-RxQueueRemove-LLR-012

Requirement ID: H398-LLD-GWY-FNC-934

This function shall loop from zero to u8\_paysize of arinc message, and copy the data present in the CAN message to u8\_payload of arinc message.

##### 10.6.1.7.13 daugwyarinc825-RxQueueRemove-LLR-013

Requirement ID: H398-LLD-GWY-FNC-935

The function shall load the complete arinc message to the destination address and return A825\_RX\_OK.

### 10.6.2 A825GetMessage

Low Level Design Details about CSU A825GetMessage will follow in the sub sections.

#### 10.6.2.1 Brief Description

The A825GetMessage function retrieves a message from the specified communication channel.

#### 10.6.2.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.6.2.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.6.2.4 Parameter list (Input/Output)

Inputs : T\_LCC\_TYPE echannel - The communication channel

Outputs : T\_A825\_MSG \*ps\_destination - The pointer to data to be extracted

#### 10.6.2.5 Return Value

T\_A825\_REC\_RESULT - A825 Receiver routines result

#### 10.6.2.6 Other CSUs called by this CSU

RxQueueRemove

#### 10.6.2.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to A825GetMessage.

##### 10.6.2.7.1 daugwyarinc825-A825GetMessage-LLR-001

Requirement ID: H398-LLD-GWY-FNC-944

The function shall call the function ‘RxQueueRemove’ with parameter (ps\_destination, Reference to NOC message queue) and set the function return value to return value of function RxQueueRemove when the selected LCC is NOC.

##### 10.6.2.7.2 daugwyarinc825-A825GetMessage-LLR-002

Requirement ID: H398-LLD-GWY-FNC-945

The function shall call the function ‘RxQueueRemove’ with parameter (ps\_destination, Reference to NSC message queue) and set the function return value to return value of function RxQueueRemove when the selected LCC is NSC .

##### 10.6.2.7.3 daugwyarinc825-A825GetMessage-LLR-003

Requirement ID: H398-LLD-GWY-FNC-946

The function shall set the return value to A825\_RX\_BAD\_CHANNEL when the selected LCC is other than NOC, NSC.

##### 10.6.2.7.4 daugwyarinc825-A825GetMessage-LLR-004

Requirement ID: H398-LLD-GWY-FNC-947

The function shall return the computed return value.

### 10.6.3 RxQueueInsert

Low Level Design Details about CSU RxQueueInsert will follow in the sub sections.

#### 10.6.3.1 Brief Description

The RxQueueInsert function inserts a CAN message into the selected queue.

#### 10.6.3.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.6.3.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.6.3.4 Parameter list (Input/Output)

Inputs : T\_CAN\_RX\_MSG \*ps\_new\_mess - The new message to insert

T\_CAN\_RX\_QUEUE \*ps\_which\_q - The queue to place the message in

Outputs : T\_CAN\_RX\_QUEUE \*ps\_which\_q - The queue to place the message in

#### 10.6.3.5 Return Value

T\_A825\_REC\_RESULT - A825 Receiver routines result

#### 10.6.3.6 Other CSUs called by this CSU

None

#### 10.6.3.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to A825GetMessage.

##### 10.6.3.7.1 daugwyarinc825-RxQueueInsert-LLR-001

Requirement ID: H398-LLD-GWY-FNC-956

The function return A825\_RX\_OVERFLOW when the queue is full ,when the buffer queue counter is equal to M\_CAN\_R\_Q\_SIZE, otherwise do nothing.

##### 10.6.3.7.2 daugwyarinc825-RxQueueInsert-LLR-002

Requirement ID: H398-LLD-GWY-FNC-957

The function shall insert the new message to the next empty space of the queue.

##### 10.6.3.7.3 daugwyarinc825-RxQueueInsert-LLR-003

Requirement ID: H398-LLD-GWY-FNC-958

The function shall Increment message counter and next message by M\_ONE .

##### 10.6.3.7.4 daugwyarinc825-RxQueueInsert-LLR-004

Requirement ID: H398-LLD-GWY-FNC-959

The function shall set next queue address to the starting of the buffer address when queue size is in range of buffer otherwise do nothing.

##### 10.6.3.7.5 daugwyarinc825-RxQueueInsert-LLR-005

Requirement ID: H398-LLD-GWY-FNC-960

The function shall return A825\_RX\_OK when the received message is successfully placed in the queue.

### 10.6.4 A825ReceiveMessage

Low Level Design Details about CSU A825ReceiveMessage will follow in the sub sections.

#### 10.6.4.1 Brief Description

The A825ReceiveMessage function is called directly from the interrupt service routine to buffer incoming messages for later processing.The LCC must conform to one of the four main A825 channels or the message will be discarded.

#### 10.6.4.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.6.4.3 List of global variables accessed and modified

Accessed : T\_OS\_EVENT \*Sem\_a825\_task

Modified : None

#### 10.6.4.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.6.4.5 Return Value

None

#### 10.6.4.6 Other CSUs called by this CSU

OsIntEnter

CanReceive

RxQueueInsert

OsSemPost

OsIntExit

#### 10.6.4.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to A825ReceiveMessage.

##### 10.6.4.7.1 daugwyarinc825-A825ReceiveMessage-LLR-001

Requirement ID: H398-LLD-GWY-FNC-969

This function shall call OsIntEnter function to tell uCOS that the function is entering the ISR.

##### 10.6.4.7.2 daugwyarinc825-A825ReceiveMessage-LLR-002

Requirement ID: H398-LLD-GWY-FNC-970

This function shall call CanReceive function with parameter (M\_CANX, M\_CAN\_FIFO0, reference of rx message) to retrieve the CAN message.

##### 10.6.4.7.3 daugwyarinc825-A825ReceiveMessage-LLR-003

Requirement ID: H398-LLD-GWY-FNC-971

The function shall

- Set lcc of a825 to (rx message extended identifier bit shifted to right by M\_LCC\_LOC) bitwise AND with M\_LCC\_MSK).

- Set rci of a825 to (rx message extended identifier bit shifted to right by M\_RCI\_LOC) bitwise AND with M\_RCI\_MSK).

##### 10.6.4.7.4 daugwyarinc825-A825ReceiveMessage-LLR-004

Requirement ID: H398-LLD-GWY-FNC-972

This function shall

- Call RxQueueInsert function and assign its return value to Result.

- Call OsSemPost with parameter (Sem\_a825\_task) only when the following conditions are met, else do nothing:

1. LCC of a825 is NOC.
2. Return value of RxQueueInsert function is A825\_RX\_OK.

##### 10.6.4.7.5 daugwyarinc825-A825ReceiveMessage-LLR-005

Requirement ID: H398-LLD-GWY-FNC-973

This function shall

- Call RxQueueInsert function and assign its return value to Result.

- Call OsSemPost with parameter (Sem\_a825\_task) only when the following conditions are met, else do nothing:

1. LCC of a825 is NSC.
2. Return value of RxQueueInsert function is A825\_RX\_OK.

##### 10.6.4.7.6 daugwyarinc825-A825ReceiveMessage-LLR-006

Requirement ID: H398-LLD-GWY-FNC-974

The function shall set Result value to A825\_RX\_BAD\_CHANNEL,when LCC of a825 is other than NSC and NOC.

##### 10.6.4.7.7 daugwyarinc825-A825ReceiveMessage-LLR-007

Requirement ID: H398-LLD-GWY-FNC-975

This function shall increment A825\_receiver\_error by one, when the return from the RxQueueInsert function is not equal to A825\_RX\_OK, otherwise do nothing.

##### 10.6.4.7.8 daugwyarinc825-A825ReceiveMessage-LLR-008

Requirement ID: H398-LLD-GWY-FNC-976

This function shall leave the ISR by calling OsIntExit function .

### 10.6.5 TxQueueInsert

Low Level Design Details about CSU TxQueueInsert will follow in the sub sections.

#### 10.6.5.1 Brief Description

The TxQueueInsert function inserts a CAN message into the selected queue.

#### 10.6.5.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.6.5.3 List of global variables accessed and modified

Accessed : U32\_critical\_sr

Modified : U32\_critical\_sr

#### 10.6.5.4 Parameter list (Input/Output)

Inputs : T\_CAN\_RX\_MSG \*ps\_new\_mess - The new message to insert

T\_CAN\_TX\_QUEUE \*ps\_which\_q - The queue to place the message in

Outputs : T\_CAN\_TX\_QUEUE \*ps\_which\_q - The queue to place the message in

#### 10.6.5.5 Return Value

T\_A825\_XMIT\_RESULT - A825 transmit routines result

#### 10.6.5.6 Other CSUs called by this CSU

CanItConfig

A825BufferCheck

RestoreStatusReg

SaveStatusReg

#### 10.6.5.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TxQueueInsert.

##### 10.6.5.7.1 daugwyarinc825-TxQueueInsert-LLR-001

Requirement ID: H398-LLD-GWY-FNC-985

This function shall call CanItConfig with parameters (M\_CANX, M\_CAN\_IT\_TME, DISABLE) which disables the can transmit Interrupt.

##### 10.6.5.7.2 daugwyarinc825-TxQueueInsert-LLR-002

Requirement ID: H398-LLD-GWY-FNC-986

This function shall call CanItConfig function with parameter (M\_CANX, M\_CAN\_IT\_TME, ENABLE) which enables the can transmit interrupt and return A825\_XM\_OVERFLOW,when the counter of message queue is equal to M\_CAN\_X\_Q\_SIZE otherwise do nothing.

##### 10.6.5.7.3 daugwyarinc825-TxQueueInsert-LLR-003

Requirement ID: H398-LLD-GWY-FNC-987

The function shall do the following:

a) Insert the new message to the next empty space of the queue(i.e set ps\_new\_mess to tail of ps\_which\_q).

b) The function calls ‘SaveStatusReg’ (M\_OS\_ENTER\_CRITICAL)’ to disable interrupts before loading the ISR into the interrupt vector table.

##### 10.6.5.7.4 daugwyarinc825-TxQueueInsert-LLR-004

Requirement ID: H398-LLD-GWY-FNC-988

The function shall do the following:

1. Increment the message queue counter by one.
2. Increment the message queue by one.

##### 10.6.5.7.5 daugwyarinc825-TxQueueInsert-LLR-005

Requirement ID: H398-LLD-GWY-FNC-989

The function shall

- Set next queue address to the starting of the buffer address (i.e tail of ps\_which\_q to buff of ps\_which\_q)when queue size is in range of buffer (i.e tail of ps\_which\_q is equal to tail of ps\_which\_q bitwise AND with buff of ps\_which\_q with size M\_CAN\_X\_Q\_SIZE) otherwise do nothing.

- Call RestoreStatusReg (M\_OS\_EXIT\_CRITICAL) to restore the interrupt.

##### 10.6.5.7.6 daugwyarinc825-TxQueueInsert-LLR-006

Requirement ID: H398-LLD-GWY-FNC-990

This function shall call CanItConfig function with parameter (M\_CANX, M\_CAN\_IT\_TME, ENABLE) which enables the can transmit inetrupt.

##### 10.6.5.7.7 daugwyarinc825-TxQueueInsert-LLR-007

Requirement ID: H398-LLD-GWY-FNC-991

This function shall call A825BufferCheck to check the various transmit buffers (in priority order) and fill the three transmit buffers and return A825\_XM\_OK.

### 10.6.6 BuildBroadcastExtId

Low Level Design Details about CSU BuildBroadcastExtId will follow in the sub sections.

#### 10.6.6.1 Brief Description

The BuildBroadcastExtId function builds the 29 bit CAN identifier for an ARINC 825 broadcast.

#### 10.6.6.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.6.6.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.6.6.4 Parameter list (Input/Output)

Inputs : T\_A825\_MSG \*ps\_xm - Pointer to an ARINC 825 message

Outputs : None

#### 10.6.6.5 Return Value

T\_UINT32 - 29 bit identifier

#### 10.6.6.6 Other CSUs called by this CSU

None

#### 10.6.6.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to BuildBroadcastExtId.

##### 10.6.6.7.1 daugwyarinc825-BuildBroadcastExtId-LLR-001

Requirement ID: H398-LLD-GWY-FNC-1000

The function shall build the 29 bit CAN identifier for an ARINC 825 broadcast type message by setting extended identifier as follows:

((lcl of bid of sid of ps\_xm bit shifted to left by M\_LCL\_LOC) bitwise OR

(pvt of bid of sid of ps\_xm bit shifted to left by M\_PVT\_LOC) bitwise OR

(doc of bid of sid of ps\_xm bit shifted to left by M\_DOC\_LOC) bitwise OR

(rci of bid of sid of ps\_xm bit shifted to left by M\_RCI\_LOC) bitwise OR

(rsd of bid of sid of ps\_xm bit shifted to left by M\_RSD\_LOC) bitwise OR

(sfid of bid of sid of ps\_xm bit shifted to left by M\_SRC\_FID\_LOC) bitwise OR

(lcc of ps\_xm bit shifted to left by M\_LCC\_LOC))

##### 10.6.6.7.2 daugwyarinc825-BuildBroadcastExtId-LLR-002

Requirement ID: H398-LLD-GWY-FNC-1001

The function shall return the extended identifier for the Broadcast msg.

### 10.6.7 BuildPeerToPeerExtId

Low Level Design Details about CSU BuildPeerToPeerExtId will follow in the sub sections.

#### 10.6.7.1 Brief Description

The BuildPeerToPeerExtId builds the 29 bit CAN identifier for an ARINC 825 peer to peer.

#### 10.6.7.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.6.7.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.6.7.4 Parameter list (Input/Output)

Inputs : T\_A825\_MSG \*ps\_xm - IN Pointer to an ARINC 825 message

Outputs : None

#### 10.6.7.5 Return Value

T\_UINT32 - 29 bit identifier

#### 10.6.7.6 Other CSUs called by this CSU

None

#### 10.6.7.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to BuildPeerToPeerExtId.

##### 10.6.7.7.1 daugwyarinc825-BuildPeerToPeerExtId-LLR-001

Requirement ID: H398-LLD-GWY-FNC-1010

The function shall build the 29 bit CAN identifier for an ARINC 825 peer to peer type message by setting extended identifier as follows:

((cfid of pid of sid of ps\_xm bit shifted to left by M\_CFID\_LOC) bitwise OR

(smt of pid of sid of ps\_xm bit shifted to left by M\_SMT\_LOC) bitwise OR

(lcl of pid of sid of ps\_xm bit shifted to left by M\_LCL\_LOC) bitwise OR

(pvt of pid of sid of ps\_xm bit shifted to left by M\_PVT\_LOC) bitwise OR

(sfid of pid of sid of ps\_xm bit shifted to left by M\_SER\_FID\_LOC) bitwise OR

(sid of pid of sid of ps\_xm bit shifted to left by M\_SID\_LOC) bitwise OR

(rci of pid of sid of ps\_xm bit shifted to left by M\_RCI\_LOC) bitwise OR

(lcc of ps\_xm bit shifted to left by M\_LCC\_LOC))

##### 10.6.7.7.2 daugwyarinc825-BuildPeerToPeerExtId-LLR-002

Requirement ID: H398-LLD-GWY-FNC-1011

The function shall return the extended identifier for the peer to peer msg.

### 10.6.8 A825Xmit

Low Level Design Details about CSU A825Xmit will follow in the sub sections.

#### 10.6.8.1 Brief Description

The A825Xmit function builds a CAN message from the user supplied ARINC 825 message. Place the newly built CAN message in the queue for that message type.

#### 10.6.8.2 List of HLRs allocated

#### 10.6.8.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.6.8.4 Parameter list (Input/Output)

Inputs : T\_A825\_MSG \*ps\_xm - Pointer to the ARINC 825 message to transmit

Outputs : None

#### 10.6.8.5 Return Value

T\_A825\_XMIT\_RESULT - A825 transmit routines result

#### 10.6.8.6 Other CSUs called by this CSU

TxQueueInsert

BuildBroadcastExtId

BuildPeerToPeerExtId

#### 10.6.8.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to A825Xmit.

##### 10.6.8.7.1 daugwyarinc825-A825Xmit-LLR-001

Requirement ID: H398-LLD-GWY-FNC-1020

The function shall set the reference of Normal Operation channel queue to p tx queue and store the return value of the function ‘BuildBroadcastExtId’ with parameter (ps\_xm) to ext id when selected lcc is NOC.

##### 10.6.8.7.2 daugwyarinc825-A825Xmit-LLR-002

Requirement ID: H398-LLD-GWY-FNC-1021

The function shall set the reference of Node Service Channel queue to p tx queue and store the return value of function ‘BuildPeerToPeerExtId’ with parameter (ps\_xm) to ext id when selected lcc is NSC.

##### 10.6.8.7.3 daugwyarinc825-A825Xmit-LLR-003

Requirement ID: H398-LLD-GWY-FNC-1022

The function shall return A825\_BAD\_CHANNEL when the selected lcc is other than Normal Operation channel or Node Service channel.

##### 10.6.8.7.4 daugwyarinc825-A825Xmit-LLR-004

Requirement ID: H398-LLD-GWY-FNC-1023

The function shall return A825\_BAD\_PAYLOAD\_SIZE when the payload size is greater than M\_BYTES\_OF\_PYLD otherwise do nothing.

##### 10.6.8.7.5 daugwyarinc825-A825Xmit-LLR-005

Requirement ID: H398-LLD-GWY-FNC-1024

The function shall set the remaining CAN message data.

1. Set the extended ID to, extended ID of message to be transmitted.
2. Set the identifier for the message to be transmitted to M\_CAN\_ID\_EXT
3. Set the not a remote frame of the message to be transmitted to M\_CAN\_RTR\_DATA
4. Set the dlc of message to be transmitted to payload size of pointer to the ARINC 825 message to transmit.

##### 10.6.8.7.6 daugwyarinc825-A825Xmit-LLR-006

Requirement ID: H398-LLD-GWY-FNC-1025

This function shall loop from M\_ZERO to payload size minus 1 and copy the payload to data of message to be transmitted.

##### 10.6.8.7.7 daugwyarinc825-A825Xmit-LLR-007

Requirement ID: H398-LLD-GWY-FNC-1026

This function shall call TxQueueInsert function with parameters (reference to tx\_message, p\_tx\_queue) and return the return value of the function TxQueueInsert.

### 10.6.9 XmitFromBuff

Low Level Design Details about CSU XmitFromBuff will follow in the sub sections.

#### 10.6.9.1 Brief Description

The XmitFromBuff function transmits a message from the supplied queue if a CAN mailbox is ready.

#### 10.6.9.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.6.9.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.6.9.4 Parameter list (Input/Output)

Inputs : T\_CAN\_TX\_QUEUE \*ps\_which\_q - The queue to place the message

Outputs : T\_CAN\_TX\_QUEUE \*ps\_which\_q - The queue to place the message

#### 10.6.9.5 Return Value

T\_UINT8 - The number of the mailbox that is used for transmission

#### 10.6.9.6 Other CSUs called by this CSU

CanTransmit

#### 10.6.9.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to XmitFromBuff.

##### 10.6.9.7.1 daugwyarinc825-XmitFromBuff-LLR-001

Requirement ID: H398-LLD-GWY-FNC-1035

This function shall call CanTransmit function with parameters (M\_CANX, head of ps\_which\_q), which transmits the can message.

##### 10.6.9.7.2 daugwyarinc825-XmitFromBuff-LLR-002

Requirement ID: H398-LLD-GWY-FNC-1036

This function shall decrement the counter of queue to place the message by one,and increment the header of queue to place the message by one, when return value of CanTransmit is not equal to M\_CAN\_TXSTATUS\_NOMAILBOX.

##### 10.6.9.7.3 daugwyarinc825-XmitFromBuff-LLR-003

Requirement ID: H398-LLD-GWY-FNC-1037

This function shall do the following when return value of CanTransmit is not equal to M\_CAN\_TXSTATUS\_NOMAILBOX otherwise do nothing.

a) function shall set head of queue to buffer of queue to place the message, when buffer of queue with parameters M\_CAN\_X\_Q\_SIZE to place the message is equal to head of queue to place the message in, otherwise do nothing.

##### 10.6.9.7.4 daugwyarinc825-XmitFromBuff-LLR-004

Requirement ID: H398-LLD-GWY-FNC-1038

This function shall return the return value of CanTransmit function.

### 10.6.10 A825BufferCheck

Low Level Design Details about CSU A825BufferCheck will follow in the sub sections.

#### 10.6.10.1 Brief Description

The A825BufferCheck function checks the various transmit buffers (in priority order) and fill the three transmit buffers.

#### 10.6.10.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.6.10.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.6.10.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.6.10.5 Return Value

None

#### 10.6.10.6 Other CSUs called by this CSU

XmitFromBuff

CanItConfig

#### 10.6.10.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to A825BufferCheck.

##### 10.6.10.7.1 daugwyarinc825-A825BufferCheck-LLR-001

Requirement ID: H398-LLD-GWY-FNC-1047

This function shall loop till Counter of Arinc 825 transmit Queue for Normal Operating Channel is not equal to zero and perform the following.

a) return from the function when M\_CAN\_TXSTATUS\_NOMAILBOX is equals to return value of the function 'XmitFromBuff' with parameter (reference of Arinc 825 transmit Queue for Normal Operating Channel), otherwise do nothing.

##### 10.6.10.7.2 daugwyarinc825-A825BufferCheck-LLR-002

Requirement ID: H398-LLD-GWY-FNC-1048

This function shall loop till Counter of Arinc 825 transmit Queue for Node Service channel is not equal to zero and perform the following.

a) return from the function when M\_CAN\_TXSTATUS\_NOMAILBOX is equals to return value of the function 'XmitFromBuff' with parameter (reference of Arinc 825 transmit Queue for Node Service channel), otherwise do nothing.

##### 10.6.10.7.3 daugwyarinc825-A825BufferCheck-LLR-003

Requirement ID: H398-LLD-GWY-FNC-1049

This function shall call CanItConfig with parameter (M\_CANX, M\_CAN\_IT\_TME, DISABLE).

### 10.6.11 CanConfig

Low Level Design Details about CSU CanConfig will follow in the sub sections.

#### 10.6.11.1 Brief Description

The CanConfig function performs low level initilization of the CAN Peripheral.

#### 10.6.11.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.6.11.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.6.11.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.6.11.5 Return Value

None

#### 10.6.11.6 Other CSUs called by this CSU

RccAhb1PeriphClockCmd

GpioPinAFConfig

GpioInit

RccApb1PeriphClockCmd

CanDeInit

CanInit

CanFilterInit

CanItConfig

#### 10.6.11.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to CanConfig.

##### 10.6.11.7.1 daugwyarinc825-CanConfig-LLR-001

Requirement ID: H398-LLD-GWY-FNC-1058

This function shall call RccAhb1PeriphClockCmd with parameters (M\_CAN\_GPIO\_CLK, ENABLE), which enables the GPIO clock.

##### 10.6.11.7.2 daugwyarinc825-CanConfig-LLR-002

Requirement ID: H398-LLD-GWY-FNC-1059

The function shall Configure pin function of PA11 by Calling function ‘GpioPinAFConfig’ with parameter (M\_CAN\_GPIO\_PORT, M\_CAN\_RX\_SOURCE, M\_CAN\_AF\_PORT).

##### 10.6.11.7.3 daugwyarinc825-CanConfig-LLR-003

Requirement ID: H398-LLD-GWY-FNC-1060

The function shall Configure pin function of PA12 by Calling function ‘GpioPinAFConfig’ with parameter (M\_CAN\_GPIO\_PORT, M\_CAN\_TX\_SOURCE, M\_CAN\_AF\_PORT).

##### 10.6.11.7.4 daugwyarinc825-CanConfig-LLR-004

Requirement ID: H398-LLD-GWY-FNC-1061

The function shall perform the following:

1. Set gpio\_pin of initial structure of GPIO to M\_CAN\_RX\_PIN bitwise OR with M\_CAN\_TX\_PIN.
2. Set gpio\_mode of initial structure of GPIO to GPIO\_MODE\_AF.
3. Set gpio\_speed of initial structure of GPIO to GPIO\_SPEED\_50MHZ.
4. Set gpio\_otype of initial structure of GPIO to GPIO\_OTYPE\_PP.
5. Set gpio\_pupd of initial structure of GPIO to GPIO\_PUPD\_UP.
6. Initializes the GPIOA peripheral by Calling function 'GpioInit' with parameter (M\_CAN\_GPIO\_PORT, reference to GPIO init structure).

##### 10.6.11.7.5 daugwyarinc825-CanConfig-LLR-005

Requirement ID: H398-LLD-GWY-FNC-1062

The function shall Enable CAN clock by Calling function 'RccApb1PeriphClockCmd' with parameter (M\_CAN\_CLK, ENABLE).

##### 10.6.11.7.6 daugwyarinc825-CanConfig-LLR-006

Requirement ID: H398-LLD-GWY-FNC-1063

The function shall call CanDeInit with parameters (M\_CANX) which initialises the CAN register.

##### 10.6.11.7.7 daugwyarinc825-CanConfig-LLR-007

Requirement ID: H398-LLD-GWY-FNC-1064

This function shall perform the following:

1. Set can\_ttcm of CAN initial structure to DISABLE.
2. Set can\_abom of CAN initial structure to ENABLE.
3. Set can\_ awum of CAN initial structure to DISABLE.
4. Set can\_ nart of CAN initial structure to DISABLE.
5. Set can\_ rflm of CAN initial structure to DISABLE.
6. Set can\_ txfp of CAN initial structure to DISABLE.
7. Set can\_ can\_mode of CAN initial structure to M\_CAN\_MODE\_NORMAL.
8. Set can\_ can\_sjw of CAN initial structure to M\_CAN\_SJW\_1TQ.
9. Set can\_ can\_bs1 of CAN initial structure to M\_CAN\_BS1\_15TQ.
10. Set can\_can\_bs2 of CAN initial structure to M\_CAN\_BS2\_5TQ.
11. Set can\_ can\_prescaler of CAN initial structure to M\_TWO.
12. Call CanInit with parameters (M\_CANX, reference to initial can structure) which initialises the CAN cells.

##### 10.6.11.7.8 daugwyarinc825-CanConfig-LLR-008

Requirement ID: H398-LLD-GWY-FNC-1065

The function shall perform the following:

Set can\_filter\_number of can filter initial structure to M\_ZERO

Set can\_filter\_mode of can filter initial structure to M\_CAN\_FILTERMODE\_IDMASK

Set can\_filter\_scale of can filter initial structure to M\_CAN\_FILTERSCALE\_32BIT

Set can\_filter\_id\_high of can filter initial structure to M\_HEX4\_ZERO

Set can\_filter\_id\_low of can filter initial structure to M\_HEX4\_ZERO

Set can\_filter\_mask\_id\_high of can filter initial structure to M\_HEX4\_ZERO

Set can\_filter\_mask\_id\_low of can filter initial structure to M\_HEX4\_ZERO

Set can\_filter\_fifo\_assignment of can filter initial structure to M\_ZERO

Set can\_filter\_activation of can filter initial structure to ENABLE

call CanFilterInit with parameter (reference to can filter initialise structure)

##### 10.6.11.7.9 daugwyarinc825-CanConfig-LLR-009

Requirement ID: H398-LLD-GWY-FNC-1066

The function shall call CanItConfig with parameter (M\_CANX, M\_CAN\_IT\_FMP0, ENABLE).

### 10.6.12 NvicConfig

Low Level Design Details about CSU NvicConfig will follow in the sub sections.

#### 10.6.12.1 Brief Description

The NvicConfig function sets up the nested vectored interrupt controler to respond to CAN receive and transmit interrupts.

#### 10.6.12.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.6.12.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.6.12.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.6.12.5 Return Value

None

#### 10.6.12.6 Other CSUs called by this CSU

IntrInstall

NvicInit

A825ReceiveMessage  
A825BufferCheck

#### 10.6.12.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to NvicConfig.

##### 10.6.12.7.1 daugwyarinc825-NvicConfig-LLR-001

Requirement ID: H398-LLD-GWY-FNC-1075

The function shall Call function ‘IntrInstall’ with parameter (INTR\_CAN\_1\_RX\_0, A825ReceiveMessage) to set up the CAN receive interrupt vector.

##### 10.6.12.7.2 daugwyarinc825-NvicConfig-LLR-002

Requirement ID: H398-LLD-GWY-FNC-1076

The function shall Call function ‘IntrInstall’ with parameter (INTR\_CAN\_1\_TX, A825BufferCheck) to set up the CAN receive interrupt vector.

##### 10.6.12.7.3 daugwyarinc825-NvicConfig-LLR-003

Requirement ID: H398-LLD-GWY-FNC-1077

The function shall perform the following:

1. Set nvic\_irq\_channel of nvic initial structure to CAN1\_RX0\_IRQN
2. Set nvic\_irq\_channel\_preempt\_pri of nvic initial structure to M\_ONE
3. Set nvic\_irq\_channel\_subpriority of nvic initial structure to M\_HEX\_ZERO
4. Set nvic\_irq\_channel\_cmd of nvic initial structure to ENABLE
5. Call function ‘NvicInit’ with parameter (Reference to NVIC init structure), to Initializes the NVIC peripheral (CAN 1 receive interrupt).

##### 10.6.12.7.4 daugwyarinc825-NvicConfig-LLR-004

Requirement ID: H398-LLD-GWY-FNC-1078

The function shall perform the following:

1. Set nvic\_irq\_channel of nvic initial structure to CAN1\_TX\_IRQN
2. Set nvic\_irq\_channel\_preempt\_pri of nvic initial structure to M\_ONE
3. Set nvic\_irq\_channel\_subpriority of nvic initial structure to M\_HEX\_ZERO
4. Set nvic\_irq\_channel\_cmd of nvic initial structure to ENABLE
5. Call function ‘NvicInit’ with parameter (Reference to NVIC init structure), to Initializes the NVIC peripheral (CAN 1 receive interrupt).

### 10.6.13 A825Init

Low Level Design Details about CSU A825Init will follow in the sub sections.

#### 10.6.13.1 Brief Description

The A825Init function performs initialization necessary for ARINC 825 communication.

#### 10.6.13.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.6.13.3 List of global variables accessed and modified

Accessed : None

Modified : Sem\_a825\_task - holds return value of OsSemCreate function

#### 10.6.13.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.6.13.5 Return Value

None

#### 10.6.13.6 Other CSUs called by this CSU

OsSemCreate

NvicConfig

CanConfig

#### 10.6.13.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to A825Init.

##### 10.6.13.7.1 daugwyarinc825-A825Init-LLR-001

Requirement ID: H398-LLD-GWY-FNC-1087

This function shall call OsSemCreate which Create a Sem to signaling the A825 receiver task and calls NvicConfig which sets up the nested vectored interrupt controler to respond to CAN receive and transmit interrupts.

##### 10.6.13.7.2 daugwyarinc825-A825Init-LLR-002

Requirement ID: H398-LLD-GWY-FNC-1088

This function shall call CanConfig which configures CAN message.

## 10.7 daugwyarinc8252

The daugwyarinc8252 CSC contains implementation of ARINC825.

### 10.7.1 RxQueueRemove2

Low Level Design Details about CSU RxQueueRemove2 will follow in the sub sections.

#### 10.7.1.1 Brief Description

The RxQueueRemove2 function retrieves a message from the specified receive queue and decompose the CAN message into an ARINC 825 message.

#### 10.7.1.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.7.1.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.7.1.4 Parameter list (Input/Output)

Inputs : T\_CAN\_RX\_QUEUE \*ps\_which\_q - The receive queue to retrieve from

Outputs : T\_CAN\_RX\_QUEUE \*ps\_which\_q - The receive queue to retrieve from

T\_A825\_MSG \*ps\_destination - Where to place the retrieved message

#### 10.7.1.5 Return Value

T\_A825\_REC\_RESULT - Return the status of the received message.

#### 10.7.1.6 Other CSUs called by this CSU

CanItConfig

#### 10.7.1.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to RxQueueRemove2.

##### 10.7.1.7.1 daugwyarinc8252-RxQueueRemove2-LLR-001

Requirement ID: H398-LLD-GWY-FNC-1098

The function shall return A825\_RX\_QUEUE\_EMPTY when cntr of ps\_which\_q is equal to M\_ZERO, otherwise do nothing.

##### 10.7.1.7.2 daugwyarinc8252-RxQueueRemove2-LLR-002

Requirement ID: H398-LLD-GWY-FNC-1099

The function shall disable the Can Receiver Transmit Interrupt by calling CanItConfig function with parameters M\_CANY, M\_CAN\_IT\_FMP0 and DISABLE.

##### 10.7.1.7.3 daugwyarinc8252-RxQueueRemove2-LLR-003

Requirement ID: H398-LLD-GWY-FNC-1100

The function shall retrieve the CAN message from the dereference of receive queue(head of ps\_which\_q) .

##### 10.7.1.7.4 daugwyarinc8252-RxQueueRemove2-LLR-004

Requirement ID: H398-LLD-GWY-FNC-1101

The function shall adjust the head and check for a wrap by decrementing number of messages in the Rx queue (cntr of ps\_which\_q) by M\_ONE and incrementing the next message to receive (head of ps\_which\_q) by M\_ONE.

##### 10.7.1.7.5 daugwyarinc8252-RxQueueRemove2-LLR-005

Requirement ID: H398-LLD-GWY-FNC-1102

The function shall set next message to receive (head of ps\_which\_q) to reference of the Rx buffer (buff of ps\_which\_q) when the next message to receive is in the message queue size range (buff [M\_CAN\_R\_Q\_SIZE] of ps\_which\_q) otherwise do nothing.

##### 10.7.1.7.6 daugwyarinc8252-RxQueueRemove2-LLR-006

Requirement ID: H398-LLD-GWY-FNC-1103

The function shall enable the Can Receiver Interrupt by calling CanItConfig function with parameters M\_CANY, M\_CAN\_IT\_FMP0 and ENABLE.

##### 10.7.1.7.7 daugwyarinc8252-RxQueueRemove2-LLR-007

Requirement ID: H398-LLD-GWY-FNC-1104

The function shall set the Logical communication channel to ((ext\_id of CAN message right shift with M\_LCC\_LOC) bitwise AND with M\_LCC\_MSK)

##### 10.7.1.7.8 daugwyarinc8252-RxQueueRemove2-LLR-008

Requirement ID: H398-LLD-GWY-FNC-1105

The function shall translate CAN MESSAGE to ARINC 825 Broadcast MESSAGE when message is a broadcast message i.e. lcc of ARINC 825 message is NOC.

(Refer the Arinc 825 design document for extracting the data from CAN message)

- rci[bit 0..1],

- doc[bit 2 .. 15],

- pvt[bit 16] ,

- lcl[bit 17],

- rsd[bit 18],

- sfid[bit 19 .. 25],

- lcc[bit 26 .. 28].

##### 10.7.1.7.9 daugwyarinc8252-RxQueueRemove2-LLR-009

Requirement ID: H398-LLD-GWY-FNC-1106

The function shall translate CAN MESSAGE to ARINC 825 Peer to Peer MESSAGE when message is a peer to peer message i.e. lcc of ARINC 825 message is NSC.

(Refer the Arinc 825 design document for extracting the data from CAN message)

- rci[bit 0..1],

- sid[bit 2 .. 8],

- sfid[bit 9 .. 15]

- pvt[bit 16] ,

- lcl[bit 17],

- smt[bit 18],

- cfid[bit 19 .. 25],

- lcc[bit 26 .. 28].

##### 10.7.1.7.10 daugwyarinc8252-RxQueueRemove2-LLR-010

Requirement ID: H398-LLD-GWY-FNC-1107

The function shall translate CAN message DLC to A825 paysize.

##### 10.7.1.7.11 daugwyarinc8252-RxQueueRemove2-LLR-011

Requirement ID: H398-LLD-GWY-FNC-1108

The function shall set number of bytes of payload (u8\_paysize) of ARINC 825 message to

M\_BYTES\_OF\_PYLD when number of bytes of payload (u8\_paysize) of ARINC 825 message is greater than M\_BYTES\_OF\_PYLD otherwise do nothing.

##### 10.7.1.7.12 daugwyarinc8252-RxQueueRemove2-LLR-012

Requirement ID: H398-LLD-GWY-FNC-1109

The function shall loop till the local counter is less than arinc 825 u8\_paysize and copy the data from CAN message to the u8\_payload of A825 payload.

##### 10.7.1.7.13 daugwyarinc8252-RxQueueRemove2-LLR-013

Requirement ID: H398-LLD-GWY-FNC-1110

The function shall copy A825 message to the destination and return A825\_RX\_OK.

### 10.7.2 A825GetMessage2

Low Level Design Details about CSU A825GetMessage2 will follow in the sub sections.

#### 10.7.2.1 Brief Description

The A825GetMessage2 function retrieves a message from the specified communication channel.

#### 10.7.2.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.7.2.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.7.2.4 Parameter list (Input/Output)

Inputs : T\_LCC\_TYPE echannel - The communication channel

Outputs : T\_A825\_MSG \*ps\_destination - The pointer to data to be extracted

#### 10.7.2.5 Return Value

T\_A825\_REC\_RESULT - A825 Receiver routines result

#### 10.7.2.6 Other CSUs called by this CSU

RxQueueRemove2

#### 10.7.2.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to A825GetMessage2.

##### 10.7.2.7.1 daugwyarinc8252-A825GetMessage2-LLR-001

Requirement ID: H398-LLD-GWY-FNC-1119

The function shall retrieve a message from the specified receive queue by calling RxQueueRemove2 function with parameters ps\_destination and reference to NOC receive queue when the channel is Normal Operating Channel.

##### 10.7.2.7.2 daugwyarinc8252-A825GetMessage2-LLR-002

Requirement ID: H398-LLD-GWY-FNC-1120

The function shall retrieve a message from the specified receive queue by calling RxQueueRemove2 function with parameters ps\_destination and reference to NSC receive queue when the channel is Node Service Channel.

##### 10.7.2.7.3 daugwyarinc8252-A825GetMessage2-LLR-003

Requirement ID: H398-LLD-GWY-FNC-1121

The function shall set the return value to A825\_RX\_BAD\_CHANNEL when the channel is other than NOC and NSC.

##### 10.7.2.7.4 daugwyarinc8252-A825GetMessage2-LLR-004

Requirement ID: H398-LLD-GWY-FNC-1122

The function shall return the computed result.

### 10.7.3 RxQueueInsert2

Low Level Design Details about CSU RxQueueInsert2 will follow in the sub sections.

#### 10.7.3.1 Brief Description

The RxQueueInsert2 function inserts a CAN message into the selected queue.

#### 10.7.3.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.7.3.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.7.3.4 Parameter list (Input/Output)

Inputs : T\_CAN\_RX\_MSG \*ps\_new\_mess - The new message to insert.

T\_CAN\_RX\_QUEUE \*Ptr\_toqueue - The queue to place the message in.

Outputs : T\_CAN\_RX\_QUEUE \*Ptr\_toqueue - The queue to place the message in.

#### 10.7.3.5 Return Value

T\_A825\_REC\_RESULT - A825 Receiver routines result

#### 10.7.3.6 Other CSUs called by this CSU

None

#### 10.7.3.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to RxQueueInsert2.

##### 10.7.3.7.1 daugwyarinc8252-RxQueueInsert2-LLR-001

Requirement ID: H398-LLD-GWY-FNC-1131

The function shall return A825\_RX\_OVERFLOW when number of messages in the Rx queue (cntr of ps\_which\_q) is M\_CAN\_R\_Q\_SIZE otherwise do nothing

##### 10.7.3.7.2 daugwyarinc8252-RxQueueInsert2-LLR-002

Requirement ID: H398-LLD-GWY-FNC-1132

The function shall insert the new message (ps\_new\_mess) into the next message in the queue (tail of ps\_which\_q)

##### 10.7.3.7.3 daugwyarinc8252-RxQueueInsert2-LLR-003

Requirement ID: H398-LLD-GWY-FNC-1133

The function shall increment number of messages in the Rx queue (cntr of ps\_which\_q) by M\_ONE and increment the next message to insert(tail of ps\_which\_q) by M\_ONE

##### 10.7.3.7.4 daugwyarinc8252-RxQueueInsert2-LLR-004

Requirement ID: H398-LLD-GWY-FNC-1134

The function shall set next message to insert (tail of ps\_which\_q) to the Rx buffer (buff of ps\_which\_q) when the next message to reference of (insert is in the message queue size range (buff [M\_CAN\_R\_Q\_SIZE] of ps\_which\_q)) otherwise do nothing.

##### 10.7.3.7.5 daugwyarinc8252-RxQueueInsert2-LLR-005

Requirement ID: H398-LLD-GWY-FNC-1135

The function shall return A825\_RX\_OK when the received message is successfully placed in the queue.

### 10.7.4 A825ReceiveMessage2

Low Level Design Details about CSU A825ReceiveMessage2 will follow in the sub sections.

#### 10.7.4.1 Brief Description

The A825ReceiveMessage2 function is called directly from the interrupt service routine to buffer incoming messages for later processing. The LCC must conform to one of the four main A825 channels or the message will be discarded.

#### 10.7.4.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.7.4.3 List of global variables accessed and modified

Accessed : Cm\_data

Sem\_a825\_task2

Modified : None

#### 10.7.4.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.7.4.5 Return Value

None

#### 10.7.4.6 Other CSUs called by this CSU

CanReceive

OsIntEnter

RxQueueInsert2

OsSemPost

OsIntExit

#### 10.7.4.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to A825ReceiveMessage2.

##### 10.7.4.7.1 daugwyarinc8252-A825ReceiveMessage2-LLR-001

Requirement ID: H398-LLD-GWY-FNC-1144

The function shall call OsIntEnter to inform uCOS that it is entering the ISR.

##### 10.7.4.7.2 daugwyarinc8252-A825ReceiveMessage2-LLR-002

Requirement ID: H398-LLD-GWY-FNC-1145

The function shall retrieve the CAN message by calling CanReceive function with parameters M\_CANY, M\_CAN\_FIFO0 and reference to CAN Rx message

##### 10.7.4.7.3 daugwyarinc8252-A825ReceiveMessage2-LLR-003

Requirement ID: H398-LLD-GWY-FNC-1146

The function shall set the logical communication channel to ((ext\_id of receive message) right shift with M\_LCC\_LOC) bitwise AND with M\_LCC\_MSK and set rci to ((ext\_id of receive message) right shift with M\_RCI\_LOC) bitwise AND with M\_RCI\_MSK.

##### 10.7.4.7.4 daugwyarinc8252-A825ReceiveMessage2-LLR-004

Requirement ID: H398-LLD-GWY-FNC-1147

The function shall perform the following when Received rci is equal to dau\_rci of Cm\_data OR received rci is M\_THREE and lcc channel is NOC.

a) store the return value of the function RxQueueInsert2 with parameter (reference of rx\_message, reference of Noc\_rx\_queue) to result.

b) call function OsSemPost with parameter Sem\_a825\_task2 when return value of the function RxQueueInsert2 with parameter (reference of rx\_message, reference of Noc\_rx\_queue) to A825\_RX\_OK otherwise do nothing.

##### 10.7.4.7.5 daugwyarinc8252-A825ReceiveMessage2-LLR-005

Requirement ID: H398-LLD-GWY-FNC-1148

The function shall perform the following when Received rci is equal to dau\_rci of Cm\_data OR received rci is M\_THREE and lcc channel is NSC.

a) store the return value of the function RxQueueInsert2 with parameter (reference of rx\_message, reference of Nsc\_rx\_queue) to result.

b) call function OsSemPost with parameter Sem\_a825\_task2 when return value of the function RxQueueInsert2 with parameter (reference of rx\_message, reference of Nsc\_rx\_queue) to A825\_RX\_OK otherwise do nothing.

##### 10.7.4.7.6 daugwyarinc8252-A825ReceiveMessage2-LLR-006

Requirement ID: H398-LLD-GWY-FNC-1149

The function shall set result to A825\_RX\_BAD\_CHANNEL when channel is other than NSC or NOC.

##### 10.7.4.7.7 daugwyarinc8252-A825ReceiveMessage2-LLR-007

Requirement ID: H398-LLD-GWY-FNC-1150

The function shall increment ARINC825 receiver error, when received CAN Rx message is from the A825\_RX\_BAD\_CHANNEL or the return value of function RxQueueInsert2 is other than A825\_RX\_OK, otherwise do nothing.

##### 10.7.4.7.8 daugwyarinc8252-A825ReceiveMessage2-LLR-008

Requirement ID: H398-LLD-GWY-FNC-1151

The function shall call OsIntExit to exit from ISR.

### 10.7.5 TxQueueInsert2

Low Level Design Details about CSU TxQueueInsert2 will follow in the sub sections.

#### 10.7.5.1 Brief Description

The TxQueueInsert2 function inserts a CAN message into the selected queue.

#### 10.7.5.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.7.5.3 List of global variables accessed and modified

Accessed : U32\_critical\_sr

Modified : U32\_critical\_sr

#### 10.7.5.4 Parameter list (Input/Output)

Inputs : T\_CAN\_TX\_MSG \*ps\_new\_mess - The new message to insert.

T\_CAN\_TX\_QUEUE \*ps\_which\_q - The queue to place the message in.

Outputs : T\_CAN\_TX\_QUEUE \*ps\_which\_q - The queue to place the message in.

#### 10.7.5.5 Return Value

T\_A825\_XMIT\_RESULT - A825 transmit routines result

#### 10.7.5.6 Other CSUs called by this CSU

CanItConfig

A825BufferCheck2

#### 10.7.5.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TxQueueInsert2.

##### 10.7.5.7.1 daugwyarinc8252-TxQueueInsert2-LLR-001

Requirement ID: H398-LLD-GWY-FNC-1160

The function shall disable the Can Transmit Interrupt by calling CanItConfig function with parameters M\_CANY, M\_CAN\_IT\_TME and DISABLE.

##### 10.7.5.7.2 daugwyarinc8252-TxQueueInsert2-LLR-002

Requirement ID: H398-LLD-GWY-FNC-1161

The function shall enable the Can Transmit Interrupt by calling CanItConfig function with parameters M\_CANY, M\_CAN\_IT\_TME and ENABLE and returns A825\_XM\_OVERFLOW when number of messages in the Tx queue (cntr of ps\_which\_q) is equal to M\_CAN\_X\_Q\_SIZE otherwise do nothing.

##### 10.7.5.7.3 daugwyarinc8252-TxQueueInsert2-LLR-003

Requirement ID: H398-LLD-GWY-FNC-1162

The function shall insert the new message(ps\_new\_mess) into the next message in the queue (tail of ps\_which\_q).

##### 10.7.5.7.4 daugwyarinc8252-TxQueueInsert2-LLR-004

Requirement ID: H398-LLD-GWY-FNC-1163

The function shall do the following

1.Call the function SaveStatusReg (M\_OS\_ENTER\_CRITICAL) to enter critical code section.

2.Adjust the tail and check for a wrap by incrementing number of messages in the Tx queue (cntr of ps\_which\_q) by M\_ONE and Next message to insert (tail of ps\_which\_q) by M\_ONE

##### 10.7.5.7.5 daugwyarinc8252-TxQueueInsert2-LLR-005

Requirement ID: H398-LLD-GWY-FNC-1164

The function shall set next message to insert(tail of ps\_which\_q) to the Tx buffer (buff of ps\_which\_q) when the next message to insert is in the Transmit message Queue Size range (buff[M\_CAN\_X\_Q\_SIZE] of ps\_which\_q) otherwise do nothing.

##### 10.7.5.7.6 daugwyarinc8252-TxQueueInsert2-LLR-006

Requirement ID: H398-LLD-GWY-FNC-1165

The function shall call the function RestoreStatusReg (M\_OS\_EXIT\_CRITICAL) to exit from the critical section code.

##### 10.7.5.7.7 daugwyarinc8252-TxQueueInsert2-LLR-007

Requirement ID: H398-LLD-GWY-FNC-1166

The function shall enable the Can Transmit Interrupt by calling CanItConfig function with parameters M\_CANY, M\_CAN\_IT\_TME and ENABLE.

##### 10.7.5.7.8 daugwyarinc8252-TxQueueInsert2-LLR-008

Requirement ID: H398-LLD-GWY-FNC-1167

The function shall call A825BufferCheck2 for immediate transmit and returns A825\_XM\_OK.

### 10.7.6 BuildBroadcastExtId2

Low Level Design Details about CSU BuildBroadcastExtId2 will follow in the sub sections.

#### 10.7.6.1 Brief Description

The BuildBroadcastExtId2 function builds the 29 bit CAN identifier for an ARINC 825 broadcast.

#### 10.7.6.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.7.6.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.7.6.4 Parameter list (Input/Output)

Inputs : T\_A825\_MSG \*ps\_xm - Pointer to an ARINC 825 message

Outputs : None

#### 10.7.6.5 Return Value

T\_UINT32 - 29 bit identifer

#### 10.7.6.6 Other CSUs called by this CSU

None

#### 10.7.6.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to BuildBroadcastExtId2.

##### 10.7.6.7.1 daugwyarinc8252-BuildBroadcastExtId2-LLR-001

Requirement ID: H398-LLD-GWY-FNC-1176

The function shall build the 29 bit CAN identifier for an ARINC 825 broadcast type message (Refer the Arinc 825 design document for extracting the data from the received Arinc 825 message )

- rci[bit 0..1],

- doc[bit 2 .. 15],

- pvt[bit 16] ,

- lcl[bit 17],

- rsd[bit 18],

- sfid[bit 19 .. 25]

- lcc[bit 26 .. 28].

##### 10.7.6.7.2 daugwyarinc8252-BuildBroadcastExtId2-LLR-002

Requirement ID: H398-LLD-GWY-FNC-1177

The function shall return 29 bit CAN identifier.

### 10.7.7 BuildPeerToPeerExtId2

Low Level Design Details about CSU BuildPeerToPeerExtId2 will follow in the sub sections.

#### 10.7.7.1 Brief Description

The BuildBroadcastExtId2 function builds the 29 bit CAN identifier for an ARINC 825 peer to peer.

#### 10.7.7.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.7.7.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.7.7.4 Parameter list (Input/Output)

Inputs : T\_A825\_MSG \*ps\_xm - Pointer to an ARINC 825 message

Outputs : None

#### 10.7.7.5 Return Value

T\_UINT32 - 29 bit identifer

#### 10.7.7.6 Other CSUs called by this CSU

None

#### 10.7.7.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to BuildPeerToPeerExtId2.

##### 10.7.7.7.1 daugwyarinc8252-BuildPeerToPeerExtId2-LLR-001

Requirement ID: H398-LLD-GWY-FNC-1186

The function shall build the 29 bit CAN identifier for an ARINC 825 peer to peer type message by setting extended identifier as follows:

((cfid of pid of sid of ps\_xm bit shifted to left by M\_CFID\_LOC) bitwise OR

(smt of pid of sid of ps\_xm bit shifted to left by M\_SMT\_LOC) bitwise OR

(lcl of pid of sid of ps\_xm bit shifted to left by M\_LCL\_LOC) bitwise OR

(pvt of pid of sid of ps\_xm bit shifted to left by M\_PVT\_LOC) bitwise OR

(sfid of pid of sid of ps\_xm bit shifted to left by M\_SER\_FID\_LOC) bitwise OR

(sid of pid of sid of ps\_xm bit shifted to left by M\_SID\_LOC) bitwise OR

(rci of pid of sid of ps\_xm bit shifted to left by M\_RCI\_LOC) bitwise OR

(lcc of ps\_xm bit shifted to left by M\_LCC\_LOC))

##### 10.7.7.7.2 daugwyarinc8252-BuildPeerToPeerExtId2-LLR-002

Requirement ID: H398-LLD-GWY-FNC-1187

The function shall return 29 bit CAN identifier.

### 10.7.8 A825Xmit2

Low Level Design Details about CSU A825Xmit2 will follow in the sub sections.

#### 10.7.8.1 Brief Description

The A825Xmit2 function builds a CAN message from the user supplied ARINC 825 message.

Place the newly built CAN message in the queue for that message type.

#### 10.7.8.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.7.8.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.7.8.4 Parameter list (Input/Output)

Inputs : T\_A825\_MSG \*ps\_xm - Pointer to the ARINC 825 message to transmit

Outputs : None

#### 10.7.8.5 Return Value

T\_A825\_XMIT\_RESULT - A825 transmit routines result

#### 10.7.8.6 Other CSUs called by this CSU

BuildBroadcastExtId2

BuildPeerToPeerExtId2

TxQueueInsert2

#### 10.7.8.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to A825Xmit2.

##### 10.7.8.7.1 daugwyarinc8252-A825Xmit2-LLR-001

Requirement ID: H398-LLD-GWY-FNC-1196

The function shall locate Queue(reference of NOC transmit queue) and builds the 29 bit ID by calling BuildBroadcastExtId2 with parameter pointer to the ARINC 825 message(ps\_xm) when

the selected channel is NOC.

##### 10.7.8.7.2 daugwyarinc8252-A825Xmit2-LLR-002

Requirement ID: H398-LLD-GWY-FNC-1197

The function shall locate Queue(reference of NSC transmit queue) and builds the 29 bit ID by calling BuildPeerToPeerExtId2 with parameter pointer to the ARINC 825 message(ps\_xm) when

a) selected channel is not NOC

b) selected channel is NSC.

##### 10.7.8.7.3 daugwyarinc8252-A825Xmit2-LLR-003

Requirement ID: H398-LLD-GWY-FNC-1198

The function shall return A825\_BAD\_CHANNEL when selected channel is other than NOC and NSC.

##### 10.7.8.7.4 daugwyarinc8252-A825Xmit2-LLR-004

Requirement ID: H398-LLD-GWY-FNC-1199

The function shall return A825\_BAD\_PAYLOAD\_SIZE when number of bytes of payload(u8\_paysize of ps\_xm) is greater than M\_BYTES\_OF\_PYLD, otherwise do nothing.

##### 10.7.8.7.5 daugwyarinc8252-A825Xmit2-LLR-005

Requirement ID: H398-LLD-GWY-FNC-1200

The function shall fill in the CAN TX message Data i.e extended identifier is set to local extended ID, type of identifier to M\_CAN\_ID\_EXT, type of frame for the received message to M\_CAN\_RTR\_DATA and length of the frame that will be received to the payload size (u8\_paysize of ps\_xm).

##### 10.7.8.7.6 daugwyarinc8252-A825Xmit2-LLR-006

Requirement ID: H398-LLD-GWY-FNC-1201

The function shall loop the local counter from M\_ZERO to the local counter is less than arinc 825 Paysize and copies the payload (u8\_payload of ps\_xm) to CAN data(data of CAN TX message).

##### 10.7.8.7.7 daugwyarinc8252-A825Xmit2-LLR-007

Requirement ID: H398-LLD-GWY-FNC-1202

The function shall insert a CAN message into the selected queue by calling TxQueueInsert2 function with parameters reference to Message to be transmitted and pointer to message queue and returns the result of the function TxQueueInsert2.

### 10.7.9 XmitFromBuff2

Low Level Design Details about CSU XmitFromBuff2 will follow in the sub sections.

#### 10.7.9.1 Brief Description

The XmitFromBuff2 transmits a message from the supplied queue if a CAN mailbox is ready.

#### 10.7.9.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.7.9.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.7.9.4 Parameter list (Input/Output)

Inputs : T\_CAN\_TX\_QUEUE \*ps\_which\_q - The queue to place the message in

Outputs : T\_CAN\_TX\_QUEUE \*ps\_which\_q - The queue to place the message in

#### 10.7.9.5 Return Value

T\_UINT8 - The number of the mailbox that is used for transmission

#### 10.7.9.6 Other CSUs called by this CSU

CanTransmit

#### 10.7.9.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to XmitFromBuff2.

##### 10.7.9.7.1 daugwyarinc8252-XmitFromBuff2-LLR-001

Requirement ID: H398-LLD-GWY-FNC-1211

The function shall transmit a CAN frame message by calling CanTransmit with parameters M\_CANY, next message to transmit (head of ps\_which\_q)

##### 10.7.9.7.2 daugwyarinc8252-XmitFromBuff2-LLR-002

Requirement ID: H398-LLD-GWY-FNC-1212

The function shall adjust the head and check for a wrap by decrementing number of messages in the Tx queue(cntr of ps\_which\_q) by M\_ONE and incrementing next message to transmit (head of ps\_which\_q) by M\_ONE when the result of the function CanTransmit is other than M\_CAN\_TXSTATUS\_NOMAILBOX otherwise, do nothing.

##### 10.7.9.7.3 daugwyarinc8252-XmitFromBuff2-LLR-003

Requirement ID: H398-LLD-GWY-FNC-1213

The function shall set next message to transmit (head of ps\_which\_q) to the Tx buffer(buff of ps\_which\_q) when

a) The next message to transmit (head of ps\_which\_q) is equal to address of (buff[M\_CAN\_X\_Q\_SIZE] of ps\_which\_q)

b) Result of the function CanTransmit is other than M\_CAN\_TXSTATUS\_NOMAILBOX.

Otherwise, do nothing.

##### 10.7.9.7.4 daugwyarinc8252-XmitFromBuff2-LLR-004

Requirement ID: H398-LLD-GWY-FNC-1214

The function shall return the number of the mailbox that is used for transmission.

### 10.7.10 A825BufferCheck2

Low Level Design Details about CSU A825BufferCheck2 will follow in the sub sections.

#### 10.7.10.1 Brief Description

The A825BufferCheck2 function checks the various transmit buffers (in priority order) and fills the transmit buffers.

#### 10.7.10.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.7.10.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.7.10.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.7.10.5 Return Value

T\_UINT8 - The number of the mailbox that is used for transmission

#### 10.7.10.6 Other CSUs called by this CSU

XmitFromBuff2

CanItConfig

#### 10.7.10.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to A825BufferCheck2.

##### 10.7.10.7.1 daugwyarinc8252-A825BufferCheck2-LLR-001

Requirement ID: H398-LLD-GWY-FNC-1223

The function shall loop till number of messages in the Tx queue (cntr of NOC transmit queue) is not equal to M\_ZERO.

##### 10.7.10.7.2 daugwyarinc8252-A825BufferCheck2-LLR-002

Requirement ID: H398-LLD-GWY-FNC-1224

The function shall loop untill cntr of NOC transmit queue is not equal to M\_ZERO and do the following:

- Return from the function when return value of the function XmitFromBuff2 with parameter reference to NOC transmit queue is M\_CAN\_TXSTATUS\_NOMAILBOX.

- Otherwise, do nothing.

##### 10.7.10.7.3 daugwyarinc8252-A825BufferCheck2-LLR-003

Requirement ID: H398-LLD-GWY-FNC-1225

The function shall loop till number of messages in the Tx queue (cntr of NSC transmit queue) is not equal to M\_ZERO.

##### 10.7.10.7.4 daugwyarinc8252-A825BufferCheck2-LLR-004

Requirement ID: H398-LLD-GWY-FNC-1226

The function shall loop till number of messages in the Tx queue (cntr of NSC transmit queue) is not equal to M\_ZERO and return when there is no mailbox free for transmission i.e. the return value of the function XmitFromBuff2 with parameter reference to NSC transmit queue is M\_CAN\_TXSTATUS\_NOMAILBOX. Otherwise, do nothing.

##### 10.7.10.7.5 daugwyarinc8252-A825BufferCheck2-LLR-005

Requirement ID: H398-LLD-GWY-FNC-1227

The function shall disable the Can Transmit Interrupt by calling CanItConfig function with parameters M\_CANY, M\_CAN\_IT\_TME and DISABLE.

### 10.7.11 CanConfig2

Low Level Design Details about CSU CanConfig2 will follow in the sub sections.

#### 10.7.11.1 Brief Description

The CanConfig2 function performs low level initialization of the CAN Peripheral.

#### 10.7.11.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.7.11.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.7.11.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.7.11.5 Return Value

None

#### 10.7.11.6 Other CSUs called by this CSU

RccAhb1PeriphClockCmd

GpioPinAFConfig

RccApb1PeriphClockCmd

CanDeInit

CanItConfig

CanInit

CanFilterInit  
GpioInit

#### 10.7.11.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to CanConfig2.

##### 10.7.11.7.1 daugwyarinc8252-CanConfig2-LLR-001

Requirement ID: H398-LLD-GWY-FNC-1236

The function shall enable GPIO clock by calling the function RccAhb1PeriphClockCmd with parameters M\_CAN2\_GPIO\_CLK and ENABLE.

##### 10.7.11.7.2 daugwyarinc8252-CanConfig2-LLR-002

Requirement ID: H398-LLD-GWY-FNC-1237

The function shall connect CAN pins to AF9(PA11) by calling the function GpioPinAFConfig with parameters M\_CAN2\_GPIO\_PORT, M\_CAN2\_RX\_SOURCE and M\_CAN2\_AF\_PORT.

##### 10.7.11.7.3 daugwyarinc8252-CanConfig2-LLR-003

Requirement ID: H398-LLD-GWY-FNC-1238

The function shall connect CAN pins to AF9 (PA12) by calling the function GpioPinAFConfig with parameters M\_CAN2\_GPIO\_PORT, M\_CAN2\_TX\_SOURCE and M\_CAN2\_AF\_PORT.

##### 10.7.11.7.4 daugwyarinc8252-CanConfig2-LLR-004

Requirement ID: H398-LLD-GWY-FNC-1239

The function shall

- set gpio\_pin of GPIO Init structure to M\_CAN2\_RX\_PIN bitwise OR M\_CAN2\_TX\_PIN

- set operating mode for the selected pin of GPIO Init structure to GPIO\_MODE\_AF

- set speed for the selected pin of GPIO Init structure to GPIO\_SPEED\_50MHZ

- set operating output type for the selected pin of GPIO Init structure to GPIO\_OTYPE\_PP

- set operating Pull-up/Pull down for the selected pin of GPIO Init structure to GPIO\_PUPD\_UP

- initializes the GPIOA peripheral by calling GpioInit function with parameters M\_CAN2\_GPIO\_PORT and reference to GPIO Init structure.

##### 10.7.11.7.5 daugwyarinc8252-CanConfig2-LLR-005

Requirement ID: H398-LLD-GWY-FNC-1240

The function shall enable CAN clock by calling the function RccApb1PeriphClockCmd with parameters M\_CAN2\_CLK and ENABLE.

##### 10.7.11.7.6 daugwyarinc8252-CanConfig2-LLR-006

Requirement ID: H398-LLD-GWY-FNC-1241

The function shall de-initialize the CAN peripheral registers by calling the function CanDeInit with parameter M\_CANY.

##### 10.7.11.7.7 daugwyarinc8252-CanConfig2-LLR-007

Requirement ID: H398-LLD-GWY-FNC-1242

The function shall

- disable time triggered communication of CAN Init structure

- enable automatic bus-off management of CAN Init structure

- disable the automatic wake-up mode of CAN Init structure

- disable the non-automatic retransmission mode of CAN Init structure

- disable the Receive FIFO Locked mode of CAN Init structure

- disable the transmit FIFO priority of CAN Init structure

- set CAN operating mode of CAN Init structure to M\_CAN\_MODE\_NORMAL

- set CAN synchronization jump width of CAN Init structure to M\_CAN\_SJW\_1TQ

- set time quanta in Bit Segment 1 of CAN Init structure to M\_CAN\_BS1\_15TQ

- set time quanta in Bit Segment 2 of CAN Init structure to M\_CAN\_BS2\_5TQ

- set length of a time quantum of CAN Init structure to M\_TWO

- initialize the CAN peripheral registers by calling the function CanInit with parameters M\_CANY and reference to CAN init structure.

##### 10.7.11.7.8 daugwyarinc8252-CanConfig2-LLR-008

Requirement ID: H398-LLD-GWY-FNC-1243

The function shall

- set filter which will be initialized of CAN filter Init structure to M\_FOURTEEN

- set filter mode to be initialized of CAN filter

Init structure to M\_CAN\_FILTERMODE\_IDMASK

- set filter scale of CAN filter Init structure to M\_CAN\_FILTERSCALE\_32BIT

- set filter identification number high, filter identification number low, filter mask number high, filter mask number low of CAN filter Init structure to M\_HEX4\_ZERO

- set FIFO assigned to the filter of CAN filter Init structure to M\_ZERO

- set can\_filter activation of CAN filter Init structure to ENABLE

- configures the CAN reception filter by calling the function CanFilterInit with parameter reference to CAN filter init structure.

##### 10.7.11.7.9 daugwyarinc8252-CanConfig2-LLR-009

Requirement ID: H398-LLD-GWY-FNC-1244

The function shall enable FIFO 0 message pending Interrupt by calling CanItConfig function with parameters M\_CANY, M\_CAN\_IT\_FMP0 and ENABLE.

### 10.7.12 NvicConfig2

Low Level Design Details about CSU NvicConfig2 will follow in the sub sections.

#### 10.7.12.1 Brief Description

The NvicConfig2 function sets up the nested vectored interrupt controller to respond to CAN receive and transmit interrupts.

#### 10.7.12.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.7.12.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.7.12.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.7.12.5 Return Value

None

#### 10.7.12.6 Other CSUs called by this CSU

IntrInstall

NvicInit

A825BufferCheck2  
A825ReceiveMessage2

#### 10.7.12.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to NvicConfig2.

##### 10.7.12.7.1 daugwyarinc8252-NvicConfig2-LLR-001

Requirement ID: H398-LLD-GWY-FNC-1253

The function shall set up the CAN receive interrupt vector by calling the function IntrInstall with parameters INTR\_CAN\_2\_RX\_0 and pointer to function A825ReceiveMessage2.

##### 10.7.12.7.2 daugwyarinc8252-NvicConfig2-LLR-002

Requirement ID: H398-LLD-GWY-FNC-1254

The function shall set up the CAN transmit interrupt vector by calling the function IntrInstall with parameters INTR\_CAN\_2\_TX and pointer to function A825BufferCheck2.

##### 10.7.12.7.3 daugwyarinc8252-NvicConfig2-LLR-003

Requirement ID: H398-LLD-GWY-FNC-1255

The function shall

- set irq channel of NVIC init structure to CAN2\_RX0\_IRQN

- set pre-emption priority for the irq channel CAN2\_RX0\_IRQN of NVIC init structure to M\_ONE

- set sub-priority level for the irq channel CAN2\_RX0\_IRQN of NVIC init structure to M\_HEX\_ZERO

- set irq channel CAN2\_RX0\_IRQN defined of NVIC init structure to ENABLE

- initialize the NVIC peripheral for the irq channel CAN2\_RX0\_IRQN by calling the function NvicInit with parameter reference to NVIC init structure.

##### 10.7.12.7.4 daugwyarinc8252-NvicConfig2-LLR-004

Requirement ID: H398-LLD-GWY-FNC-1256

The function shall

- set irq channel of NVIC init structure to CAN2\_TX\_IRQN

- set pre-emption priority for the irq channel CAN2\_TX\_IRQN of NVIC init structure to M\_ONE

- set sub-priority level for the irq channel CAN2\_TX\_IRQN of NVIC init structure to M\_HEX\_ZERO

- set irq channel CAN2\_TX\_IRQN defined of NVIC init structure to ENABLE

- initialize the NVIC peripheral for the irq channel CAN2\_TX\_IRQN by calling the function NvicInit with parameter reference to NVIC init structure.

### 10.7.13 A825Init2

Low Level Design Details about CSU A825Init2 will follow in the sub sections.

#### 10.7.13.1 Brief Description

The A825Init2 function performs initialization necessary for A825 communication.

#### 10.7.13.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.7.13.3 List of global variables accessed and modified

Accessed : Sem\_a825\_task2

Modified : None

#### 10.7.13.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.7.13.5 Return Value

None

#### 10.7.13.6 Other CSUs called by this CSU

OsSemCreate

NvicConfig2

CanConfig2

#### 10.7.13.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to A825Init2.

##### 10.7.13.7.1 daugwyarinc8252-A825Init2-LLR-001

Requirement ID: H398-LLD-GWY-FNC-1265

The function OsSemCreate shall be called to create semaphore for signaling Sem\_a825\_task2 by passing M\_ZERO as a counter value.

##### 10.7.13.7.2 daugwyarinc8252-A825Init2-LLR-002

Requirement ID: H398-LLD-GWY-FNC-1266

The function shall perform NVIC configuration by calling NvicConfig2 function.

##### 10.7.13.7.3 daugwyarinc8252-A825Init2-LLR-003

Requirement ID: H398-LLD-GWY-FNC-1267

The function shall perform CAN configuration by calling CanConfig2 function.

## 10.8 daugwybittest

The daugwybittest defines the functions for PBIT.

### 10.8.1 RamTest

Low Level Design Details about CSU RamTest will follow in the sub sections.

#### 10.8.1.1 Brief Description

The RamTest function performs the SRAM test.

#### 10.8.1.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.8.1.3 List of global variables accessed and modified

Accessed : Sram\_buffer\_check

Modified : Sram\_buffer\_check

#### 10.8.1.4 Parameter list (Input/Output)

Inputs: None

Outputs: None

#### 10.8.1.5 Return Value

None

#### 10.8.1.6 Other CSUs called by this CSU

RterrError

RterrForever

#### 10.8.1.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to RamTest

##### 10.8.1.7.1 daugwybittest-RamTest-LLR-001

Requirement ID: H398-LLD-GWY-FNC-1277

The function shall set ram\_test\_fail of Error status to FALSE and perform the SRAM test by looping for each Sram\_buffer\_check location from range M\_ZERO to M\_RAM\_SIZE minus 1

a) Set Sram\_buffer\_check location to M\_CHECK\_5A5A.

b) Do the following when Sram\_buffer\_check location is equal to M\_CHECK\_5A5A

(i) Set Sram\_buffer\_check location to M\_CHECK\_A5A5.

ii) set ram\_test\_fail of Error status to TRUE and call 'RterrError' with parameters MEMMORY\_ERR, M\_TWO and reference of function pointer RterrForever when Sram\_buffer\_check location is not eqaul to M\_CHECK\_A5A5,

Otherwise do nothing

c) Do the following when Sram\_buffer\_check location is not equal to M\_CHECK\_5A5A

- set ram\_test\_fail of Error status to TRUE

- call 'RterrError' with parameters MEMMORY\_ERR, M\_TWO and reference of function pointer RterrForever

### 10.8.2 ProcessorTest

Low Level Design Details about CSU ProcessorTest will follow in the sub sections.

#### 10.8.2.1 Brief Description

The ProcessorTest function performs the CPU test.

#### 10.8.2.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.8.2.3 List of global variables accessed and modified

Accessed : Proc\_test\_res

Modified : Proc\_test\_res

#### 10.8.2.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.8.2.5 Return Value

None

#### 10.8.2.6 Other CSUs called by this CSU

AluTest

RterrError

RterrForever

#### 10.8.2.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to ProcessorTest

##### 10.8.2.7.1 daugwybittest-ProcessorTest-LLR-001

Requirement ID: H398-LLD-GWY-FNC-1286

The function shall do the following:

* Set 'Proc\_test\_res' to FAILED
* Perform the arithmetical and logical operations on registers by calling function AluTest.
* Call the function RterrError with parameters BIT\_ERR,M\_ONE and reference of function pointer RterrForever to flash code accordingly when value of 'Proc\_test\_res' is equal to FAILED, otherwise do nothing

### 10.8.3 CrcCheck

Low Level Design Details about CSU CrcCheck will follow in the sub sections.

#### 10.8.3.1 Brief Description

The CrcCheck function performs the CRC check.

#### 10.8.3.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.8.3.3 List of global variables accessed and modified

Accessed :None

Modified : None

#### 10.8.3.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.8.3.5 Return Value

None

#### 10.8.3.6 Other CSUs called by this CSU

CrcResetDr

CrcCalcBlockCrc

RterrError

RterrForever

GpioSetBits

#### 10.8.3.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to CrcCheck

##### 10.8.3.7.1 daugwybittest-CrcCheck-LLR-001

Requirement ID: H398-LLD-GWY-FNC-1311

The function shall Resets the CRC data Register by calling function CrcResetDr.

##### 10.8.3.7.2 daugwybittest-CrcCheck-LLR-002

Requirement ID: H398-LLD-GWY-FNC-1312

The function shall Calculates application data check sum by calling CrcCalcBlockCrc

with parameters M\_MEMMAP\_DATA\_1\_ADDR and M\_MEMMAP\_DATA\_1\_CRC\_CNT.

##### 10.8.3.7.3 daugwybittest-CrcCheck-LLR-003

Requirement ID: H398-LLD-GWY-FNC-1313

The function shall do the following when the calculated Application data check sum is not equal to actual CRC 'M\_MEMMAP\_DATA\_1\_CRC\_ADDR', otherwise do nothing

a) call GpioSetBits with reference to GPIO port B (M\_GPIOB) and GPIO pin 2 (M\_GPIO\_PIN\_2)

as parameters using macro M\_HW\_LED\_HB\_ON to turn on the heartbeat LED.

b) read Boot CRC error and flash code accordingly by calling RterrError with parameters

MCD\_ERR,M\_ZERO and reference of function pointer RterrForever

##### 10.8.3.7.4 daugwybittest-CrcCheck-LLR-004

Requirement ID: H398-LLD-GWY-FNC-1314

The function shall do the following:

-Set Application data check sum to M\_ZERO

-Resets the CRC data Register by calling function CrcResetDr

##### 10.8.3.7.5 daugwybittest-CrcCheck-LLR-005

Requirement ID: H398-LLD-GWY-FNC-1315

The function shall calculate Flight Application check sum by calling CrcCalcBlockCrc

with parameters M\_MEMMAP\_FLIGHT\_ADDR and M\_MEMMAP\_FLIGHT\_CRC\_CNT.

##### 10.8.3.7.6 daugwybittest-CrcCheck-LLR-006

Requirement ID: H398-LLD-GWY-FNC-1316

The function shall do the following when the calculated Flight Application check sum is not equal to actual

CRC 'M\_MEMMAP\_FLIGHT\_CRC\_ADDR', otherwise do nothing

a) call GpioSetBits with reference to GPIO port B (M\_GPIOB) and GPIO pin 2 (M\_GPIO\_PIN\_2)

as parameters using macro M\_HW\_LED\_HB\_ON to turn on the heartbeat LED.

b) read Boot CRC error and flash code accordingly by calling RterrError with parameters

FLIGHT\_APP\_CRC\_ERR,M\_ZERO and reference of function pointer RterrForever.

### 10.8.4 XRamTest

Low Level Design Details about CSU XRamTest will follow in the sub sections.

#### 10.8.4.1 Brief Description

The XRamTest function performs the XRAM test.

#### 10.8.4.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.8.4.3 List of global variables accessed and modified

Accessed : Xram\_test\_buffer

Modified : Xram\_test\_buffer, Cm\_data

#### 10.8.4.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.8.4.5 Return Value

None

#### 10.8.4.6 Other CSUs called by this CSU

RterrError

RterrForever

#### 10.8.4.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to XRamTest

##### 10.8.4.7.1 daugwybittest-XRamTest-LLR-001

Requirement ID: H398-LLD-GWY-FNC-1325

The function shall set xram\_error of error of Cm\_data to FALSE

##### 10.8.4.7.2 daugwybittest-XRamTest-LLR-002

Requirement ID: H398-LLD-GWY-FNC-1326

The function shall loop from xram equal to M\_XRAM\_TEST\_START until M\_XRAM\_NVRAM\_TEST minus 1 and set Xram\_test\_buffer of index xram buffer with M\_5A\_XRAM\_TEST

##### 10.8.4.7.3 daugwybittest-XRamTest-LLR-003

Requirement ID: H398-LLD-GWY-FNC-1327

The function shall loop from xram equal to M\_XRAM\_TEST\_START until M\_XRAM\_NVRAM\_TEST and set Xram\_test\_buffer of index xram buffer with M\_A5\_XRAM\_TEST when Xram\_test\_buffer of index xram buffer is equal to M\_5A\_XRAM\_TEST

##### 10.8.4.7.4 daugwybittest-XRamTest-LLR-004

Requirement ID: H398-LLD-GWY-FNC-1328

The function shall loop from xram equal to M\_XRAM\_TEST\_START until M\_XRAM\_NVRAM\_TEST and do the following when Xram\_test\_buffer of index xram buffer is equal to M\_5A\_XRAM\_TEST and Xram\_test\_buffer of index xram buffer is not equal to M\_A5\_XRAM\_TEST

* Set xram\_error of error of Cm\_data to TRUE
* Call RterrError with parameters MEMMORY\_ERR,M\_ONE and reference of function pointer RterrForever

##### 10.8.4.7.5 daugwybittest-XRamTest-LLR-005

Requirement ID: H398-LLD-GWY-FNC-1329

The function shall loop from xram equal to M\_XRAM\_TEST\_START until M\_XRAM\_NVRAM\_TEST and do the nothing when Xram\_test\_buffer of index xram buffer is equal to M\_5A\_XRAM\_TEST and Xram\_test\_buffer of index xram buffer is equal to M\_A5\_XRAM\_TEST

##### 10.8.4.7.6 daugwybittest-XRamTest-LLR-006

Requirement ID: H398-LLD-GWY-FNC-1330

The function shall loop from xram equal to M\_XRAM\_TEST\_START until M\_XRAM\_NVRAM\_TEST and do the following when Xram\_test\_buffer of index xram buffer is not equal to M\_5A\_XRAM\_TEST

* Set xram\_error of error of Cm\_data to TRUE
* Call RterrError with parameters MEMMORY\_ERR,M\_ONE and reference of function pointer RterrForever

### 10.8.5 PbitCheck

Low Level Design Details about CSU PbitCheck will follow in the sub sections.

#### 10.8.5.1 Brief Description

The PbitCheck function performs the Power-on Built in test.

#### 10.8.5.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.8.5.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.8.5.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.8.5.5 Return Value

None

#### 10.8.5.6 Other CSUs called by this CSU

ProcessorTest

RamTest

#### 10.8.5.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to PbitCheck

##### 10.8.5.7.1 daugwybittest-PbitCheck-LLR-001

Requirement ID: H398-LLD-GWY-FNC-1339

The function shall perform Power-on Built in test by calling functions ProcessorTest and RamTest

## 10.9 daugwybittesta

The daugwybittesta CSC defines function for arithmetical and logical operations on registers during PBIT.

### 10.9.1 AluTest

Low Level Design Details about CSU AluTest will follow in the sub sections.

#### 10.9.1.1 Brief Description

The AluTest function performs the arithmetical and logical operations on registers.

#### 10.9.1.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD(H398-003-012-GWY).

#### 10.9.1.3 List of global variables accessed and modified

Accessed: Proc\_test\_res

Modified: None

#### 10.9.1.4 Parameter list (Input/Output)

Inputs: None

Outputs: None

#### 10.9.1.5 Return Value

None

#### 10.9.1.6 Other CSUs called by this CSU

None

#### 10.9.1.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to AluTest.

##### 10.9.1.7.1 daugwybittesta-AluTest-LLR-001

Requirement ID: H398-LLD-GWY-FNC-1349

The function shall perform the addition operation on registers (R1, R2) with operands M\_CONST\_SIX and M\_CONST\_FIVE. when the obtained result (R2) matches with the value M\_CONST\_ELEVEN, then perform carry flag test otherwise exit from the subroutine.

##### 10.9.1.7.2 daugwybittesta-AluTest-LLR-002

Requirement ID: H398-LLD-GWY-FNC-1350

The function shall perform the addition operation on registers (R1, R2) with operands M\_CONST\_SIX and M\_CONST\_FIVE and store the status of the carry flag from APSR into the register (R4). When the obtained result (R4) matches with the value M\_CONST\_CARRY\_FLAG, then perform zero flag test otherwise exit from the subroutine.

##### 10.9.1.7.3 daugwybittesta-AluTest-LLR-003

Requirement ID: H398-LLD-GWY-FNC-1351

The function shall perform the addition operation on registers (R1, R2) with operands M\_CONST\_PATTERN and M\_CONST\_PATTERN and store the status of the zero flag from APSR into the register (R4). When the obtained result (R4) matches with the value M\_CONST\_OVERFLOW\_FLAG perform subtract test otherwise exit from the subroutine.

##### 10.9.1.7.4 daugwybittesta-AluTest-LLR-004

Requirement ID: H398-LLD-GWY-FNC-1352

The function shall perform the addition operation on registers (R1, R2) with operands M\_CONST\_SIX and M\_CONST\_FIVE, compare the result (R2) with M\_CONST\_ELEVEN and store the status of the zero flag from APSR into the register (R4). When the obtained result (R4) matches with the value M\_CONST\_ZERO\_FLAG, then perform overflow flag test otherwise exit from the subroutine.

##### 10.9.1.7.5 daugwybittesta-AluTest-LLR-005

Requirement ID: H398-LLD-GWY-FNC-1353

The function shall perform the subtract operation on registers (R1, R2) with operands M\_CONST\_SIX and M\_CONST\_FIVE and when the obtained result(R1) matches with the value M\_CONST\_ONE then perform multiply test otherwise exit from the subroutine.

##### 10.9.1.7.6 daugwybittesta-AluTest-LLR-006

Requirement ID: H398-LLD-GWY-FNC-1354

The function shall perform the multiply operation on registers (R1, R2) with operands M\_CONST\_THREE and M\_CONST\_TWO and when the obtained result (R1) matches with the value M\_CONST\_SIX, then perform division test otherwise exit from the subroutine.

##### 10.9.1.7.7 daugwybittesta-AluTest-LLR-007

Requirement ID: H398-LLD-GWY-FNC-1355

The function shall perform the unsigned division operation on registers (R1, R2) with operands M\_CONST\_FOUR and M\_CONST\_TWO and when the obtained result (R1) matches with the value M\_CONST\_TWO then perform logical AND test otherwise exit from the subroutine.

##### 10.9.1.7.8 daugwybittesta-AluTest-LLR-008

Requirement ID: H398-LLD-GWY-FNC-1356

The function shall perform the logical AND operation on registers (R1, R2) with operands M\_CONST\_HEX\_ONE and M\_CONST\_SEVENTEEN and when the obtained result (R3) matches with the value M\_CONST\_ONE then perform logical OR test otherwise exit from the subroutine.

##### 10.9.1.7.9 daugwybittesta-AluTest-LLR-009

Requirement ID: H398-LLD-GWY-FNC-1357

The function shall perform the logical OR operation on registers (R1, R2) with operands M\_CONST\_SIXTEEN and M\_CONST\_HEX\_ONE and when the obtained result (R3) matches with the value M\_CONST\_SEVENTEEN then perform exclusive OR test otherwise exit from the subroutine.

##### 10.9.1.7.10 daugwybittesta-AluTest-LLR-010

Requirement ID: H398-LLD-GWY-FNC-1358

The function shall perform the logical XOR operation on registers (R1, R2) with operands M\_CONST\_SEVENTEEN and M\_CONST\_HEX\_ONE and when the obtained result (R3) matches with the value M\_CONST\_SIXTEEN then branch to Test result otherwise exit from the subroutine.

##### 10.9.1.7.11 daugwybittesta-AluTest-LLR-011

Requirement ID: H398-LLD-GWY-FNC-1359

The function shall initialize R4 with M\_CONST\_ONE and load the address of Proc\_test\_res to the R2 register, set the global variable Proc\_test\_res to TRUE by writing R4 to content of R2 address.

## 10.10 daugwycbit

The module implements CBIT functionality.

### 10.10.1 StackTest

Low Level Design Details about CSU StackTest will follow in the sub sections.

#### 10.10.1.1 Brief Description

The function performs Stack Test.

#### 10.10.1.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.10.1.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 10.10.1.4 Parameter list (Input/Output)

Inputs: T\_UINT32 \*buffer - Pointer to the top-of-stack address

T\_UINT32 length - length of stack.

Outputs: T\_UINT32 \*buffer - Pointer to the top-of-stack address

#### 10.10.1.5 Return Value

None

#### 10.10.1.6 Other CSUs called by this CSU

None

#### 10.10.1.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to StackTest.

##### 10.10.1.7.1 daugwycbit-StackTest-LLR-001

Requirement ID: H398-LLD-GWY-FNC-1369

The function shall do the stack test for all M\_TEN contiguous memory as follows:

1. Decrement buffer value up to when index less than or equal to size (i.e size is equal to M\_GET\_70PERC\_STACK(length)) by incrementing index counter.
2. Decrement buffer value from index M\_ZERO to index less than or equal to M\_FOURTEEN minus 1 by incrementing index counter.
3. Call RterrError function with parameters (BIT\_ERR,2, RterrForever). When each buffer content is not equal to M\_CBIT\_TASK\_STK\_VAL

Other wise do nothing.

### 10.10.2 ContinuousBitTask

Low Level Design Details about CSU ContinuousBitTask will follow in the sub sections.

#### 10.10.2.1 Brief Description

The function increments system tick for signaling semaphore for CBIT.

#### 10.10.2.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.10.2.3 List of global variables accessed and modified

Accessed: Appdata\_appdata

A429\_task\_stk

A429\_out\_task\_stk

Logic\_task\_stk

Logic\_com4\_task\_stk

L\_app\_task\_stk

A825comm\_task\_stk

A825comm\_task\_stk2

Init\_task\_stk

Os\_task\_idle\_stk

Eti\_taskstk

Soundtaskstk

Cbit\_task\_stack

Modified: None

#### 10.10.2.4 Parameter list (Input/Output)

Inputs: Ptr\_data: Pointer to NULL data

Outputs: Ptr\_data: Pointer to NULL data

#### 10.10.2.5 Return Value

None

#### 10.10.2.6 Other CSUs called by this CSU

OsSemPend

StackTest

#### 10.10.2.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to ContinuousBitTask.

##### 10.10.2.7.1 daugwycbit-ContinuousBitTask-LLR-001

Requirement ID: H398-LLD-GWY-FNC-1378

The function shall set Ptr\_data with addition of (Ptr\_data and M\_ZERO) and loop forever to perform following

- call 'OsSemPend' for waiting on a semaphore with parameters

a) pointer to the semaphore of CBIT task,

b) timeout value of M\_ZERO, and

c) address of error code variable.

- Do the following when func\_state of a429chan with index M\_ZERO of Appdata\_appdata is ENABLED

1. Call StackTest with parameters reference of A429\_task\_stk with index as M\_A429\_TASK\_STK\_SIZE and M\_A429\_TASK\_STK\_SIZE
2. Call StackTest with parameters reference of A429\_out\_task\_stk with index as M\_A429\_TASK\_STK\_SIZE and M\_A429\_OUT\_TASK\_STK\_SIZE

- call 'StackTest' with parameters Reference of A429\_task\_stk with index as M\_A429\_TASK\_STK\_SIZE and M\_A429\_TASK\_STK\_SIZE.

- call 'StackTest' with parameters Reference of A429\_out\_task\_stk with index as M\_A429\_OUT\_TASK\_STK\_SIZE and M\_A429\_OUT\_TASK\_STK\_SIZE.

- call 'StackTest' with parameters Reference of Logic\_task\_stk with index as M\_LOGIC\_TASK\_STK\_SIZE and M\_LOGIC\_TASK\_STK\_SIZE.

- call 'StackTest' with parameters Reference of Logic\_com4\_task\_stk with index as M\_COM4\_TASK\_STK\_SIZE and M\_COM4\_TASK\_STK\_SIZE.

- call 'StackTest' with parameters Reference of L\_app\_task\_stk with index as M\_APP\_TASK\_STK\_SIZE and M\_APP\_TASK\_STK\_SIZE.

- call 'StackTest' with parameters Reference of A825comm\_task\_stk with index as M\_A825\_TASK\_STK\_SIZE and M\_A825\_TASK\_STK\_SIZE.

- call 'StackTest' with parameters Reference of A825comm\_task\_stk2 with index as M\_A8252\_TASK\_STK\_SIZE and M\_A8252\_TASK\_STK\_SIZE.

- call 'StackTest' with parameters Reference of Init\_task\_stk with index as M\_INIT\_TASK\_STK\_SIZE and M\_INIT\_TASK\_STK\_SIZE.

- call 'StackTest' with parameters Reference of Os\_task\_idle\_stk with index as M\_OS\_IDLE\_TASK\_STK\_SIZE and M\_OS\_IDLE\_TASK\_STK\_SIZE.

- call 'StackTest' with parameters Reference of Eti\_taskstk with index as M\_ETI\_TASK\_STK\_SIZE and M\_ETI\_TASK\_STK\_SIZE.

- call 'StackTest' with parameters Reference of Soundtaskstk with index as M\_SOUND\_TASK\_STK\_SIZE and M\_SOUND\_TASK\_STK\_SIZE.

- call 'StackTest' with parameters Reference of Cbit\_task\_stack with index as M\_CBIT\_TASK\_STK\_SIZE and M\_CBIT\_TASK\_STK\_SIZE.

### 10.10.3 ContinuousBitTaskInit

Low Level Design Details about CSU ContinuousBitTaskInit will follow in the sub sections.

#### 10.10.3.1 Brief Description

The function initializes stack and creates Task to implement CBIT.

#### 10.10.3.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.10.3.3 List of global variables accessed and modified

Accessed: Cbit\_task\_stack

Modified: Cbit\_task\_stack

#### 10.10.3.4 Parameter list (Input/Output)

Inputs: None

Outputs: None

#### 10.10.3.5 Return Value

None

#### 10.10.3.6 Other CSUs called by this CSU

TbaseTaskSignaling

OsTaskCreate

OsSemCreate

RterrError

ContinuousBitTask

RterrForever

#### 10.10.3.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to ContinuousBitTaskInit.

##### 10.10.3.7.1 daugwycbit-ContinuousBitTaskInit-LLR-001

Requirement ID: H398-LLD-GWY-FNC-1387

The function shall initialize task signaling parameters by calling 'TbaseTaskSignaling' with parameters

a) M\_CBIT\_TASK\_TICKS and

b) Return value of function OsSemCreate with parameter M\_ZERO.

##### 10.10.3.7.2 daugwycbit-ContinuousBitTaskInit-LLR-002

Requirement ID: H398-LLD-GWY-FNC-1388

The function shall initialize all the elements of Cbit\_task\_stack with M\_CBIT\_TASK\_STK\_VAL up to when stack counter value is less than M\_CBIT\_TASK\_STK\_SIZE.

##### 10.10.3.7.3 daugwycbit-ContinuousBitTaskInit-LLR-003

Requirement ID: H398-LLD-GWY-FNC-1389

The function 'OsTaskCreate' shall be called to create ContinuousBitTask by passing following arguments.

- Pointer to the ContinuousBitTask's address.

- M\_HW\_NULL pointer passed as argument to the ContinuousBitTask.

- ContinuousBitTask's top-of-stack i.e Cbit\_task\_stack [M\_CBIT\_TASK\_STK\_SIZE].

- Task priority as M\_CBIT\_TASK\_PRIO.

When the return value of the 'OsTaskCreate' is not equal to M\_OS\_NO\_ERR then call RterrError with parameters BIT\_ERR, M\_THREE and reference to function pointer RterrForever otherwise do nothing.

## 10.11 daugwycomm

The daugwycomm CSC contains the Communication control.

### 10.11.1 CommPutChars

Low Level Design Details about CSU CommPutChars will follow in the sub sections.

#### 10.11.1.1 Brief Description

The CommPutChars function transmits Characters/values via rs232.

#### 10.11.1.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.11.1.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.11.1.4 Parameter list (Input/Output)

Inputs : void \*p\_src - pointer to buffer which contains data to be transmitted

T\_SINT16 i16\_timeout - Maximum time given for specified length of data

transmission

T\_UINT16 u16\_len - length of data to be transmitted

T\_UINT8 u8\_comport - Comport number

Outputs : T\_SINT16 i16\_timeout - Maximum time given for specified length of data

transmission

#### 10.11.1.5 Return Value

T\_SINT16 Return timeout value.

#### 10.11.1.6 Other CSUs called by this CSU

Rs232TxBlock,

OsTimeDly

#### 10.11.1.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to CommPutChars

##### 10.11.1.7.1 daugwycomm-CommPutChars-LLR-001

Requirement ID: H398-LLD-GWY-FNC-1399

The function shall loop infinitely and perform the following in each loop otherwise return M\_ZERO

a) load the data to be transmitted into transmitting buffer by calling 'Rs232TxBlock' with parameters data to be transmitted (pu8\_src), data length(u16\_len) and comport number(u8\_comport).

b) check status after transmission of data and do the following

When status is RS232\_OK return i16\_timeout value.

When status is RS232\_WAITING and i16\_timeout is less than or equal to M\_ZERO, return i16\_timeout-M\_ONE.

When status is RS232\_WAITING and i16\_timeout is greater than M\_ZERO, then decrement the i16\_timeout by M\_ONE and call OsTimeDly with M\_ONE as parameter.

otherwise return M\_MINUS\_ONE.

### 10.11.2 CommGetChars

Low Level Design Details about CSU CommGetChars will follow in the sub sections.

#### 10.11.2.1 Brief Description

The CommGetChars function gets Characters/Values from Rs232.

#### 10.11.2.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.11.2.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.11.2.4 Parameter list (Input/Output)

Inputs : T\_UINT8 \*pu8\_dest - pointer to receive buffer,,

T\_SINT16 i16\_timeout - Maximum time given for specified length of data

to be received

T\_UINT16 u16\_len - length of data to be copied

T\_UINT8 u8\_comport - Comport number

Outputs : T\_SINT16 i16\_timeout - Maximum time given for specified length of data

to be received

#### 10.11.2.5 Return Value

T\_SINT16 Return timeout value.

#### 10.11.2.6 Other CSUs called by this CSU

Rs232RxBlock

OsTimeDly

ResetCOM

#### 10.11.2.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to CommGetChars

##### 10.11.2.7.1 daugwycomm-CommGetChars-LLR-001

Requirement ID: H398-LLD-GWY-FNC-1408

The function shall return i16\_timeout value When i16\_timeout value is less than M\_ZERO otherwise do nothing.

##### 10.11.2.7.2 daugwycomm-CommGetChars-LLR-002

Requirement ID: H398-LLD-GWY-FNC-1409

The function shall loop infinitely When i16\_timeout value is greater than or equal to M\_ZERO and perform the following in each loop otherwise return M\_ZERO

a) store the data into receive buffer from R232 by calling 'Rs232RxBlock' with parameters receive buffer(pu8\_dest), data length(u16\_len) and comport number(u8\_comport).

b) check status after receive of data and do the following

- When status is RS232\_OK return i16\_timeout value.

- When status is RS232\_WAITING and i16\_timeout is less than or equal to M\_ZERO, return i16\_timeout decrement by M\_ONE,

- When status is RS232\_WAITING and i16\_timeout is greater than M\_ZERO, then increment the i16\_timeout by M\_ONE and call OsTimeDly with M\_ONE as parameter,

- otherwise call ResetCOM with parameter u8\_comport and return M\_MINUS\_ONE.

## 10.12 daugwycrc

The daugwycrc CSC contains the implementation of the crc protocol.

### 10.12.1 UpdCrc32

Low Level Design Details about CSU UpdCrc32 will follow in the sub sections.

#### 10.12.1.1 Brief Description

The UpdCrc32 function returns the updated CRC

#### 10.12.1.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.12.1.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.12.1.4 Parameter list (Input/Output)

Inputs : const T\_UINT8 b\_data Next byte

const T\_CRC32 crc Accumulated CRC

Outputs : None

#### 10.12.1.5 Return Value

T\_CRC32 returns the updated CRC

#### 10.12.1.6 Other CSUs called by this CSU

None

#### 10.12.1.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to UpdCrc32.

##### 10.12.1.7.1 daugwycrc-UpdCrc32-LLR-001

Requirement ID: H398-LLD-GWY-FNC-1419

The function shall return computed CRC value

NOTE: a) The CRC value is computed as follows

(crc right shifted by M\_SHIFT\_BY\_8, bitwise AND with M\_CRC\_MASK) XOR with (value from CRC table for corresponding index)

b) index for CRC table is computed as follows

(crc XOR b\_data, bitwise AND with M\_EXTRT\_LBYTE).

## 10.13 daugwycrt0

The daugwycrt0 CSC copies and resets the vector table for STM32F40x. After Reset the Cortex-M4 processor is in Thread mode, priority is Privileged, and the Stack is set to Main.

### 10.13.1 Crt0TransferData

Low Level Design Details about CSU Crt0TransferData will follow in the sub sections.

#### 10.13.1.1 Brief Description

The Crt0TransferData function is reset handler for reset condition assign to the interrupt section.

#### 10.13.1.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.13.1.3 List of global variables accessed and modified

Accessed : \_data\_rom

\_data\_start

\_data\_end

\_bss\_end

\_bss\_start

Modified : \_data\_start

\_bss\_start

#### 10.13.1.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.13.1.5 Return Value

None

#### 10.13.1.6 Other CSUs called by this CSU

MainFunc

10.13.1.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to Crt0TransferData.

##### 10.13.1.7.1 daugwycrt0-Crt0TransferData-LLR-001

Requirement ID: H398-LLD-GWY-FNC-1429

The function shall move Data Memory Section from ROM to RAM when the range of the data memory is less than \_data\_end.

##### 10.13.1.7.2 daugwycrt0-Crt0TransferData-LLR-002

Requirement ID: H398-LLD-GWY-FNC-1430

The function shall fill the bss Memory Section with zero when the range of the bss memory is less than \_bss\_end.

##### 10.13.1.7.3 daugwycrt0-Crt0TransferData-LLR-003

Requirement ID: H398-LLD-GWY-FNC-1431

The function shall call MainFunc function to branch to main function.

## 10.14 daugwydtoa

The daugwydtoa CSC provides interface to internal DAC.

### 10.14.1 DtoaHwInit

Low Level Design Details about CSU DtoaHwInit will follow in the sub sections.

#### 10.14.1.1 Brief Description

The DtoaHwInit function initialize DAC and GPIO port A pin 4 for DAC.

#### 10.14.1.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.14.1.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.14.1.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.14.1.5 Return Value

None

#### 10.14.1.6 Other CSUs called by this CSU

DacInit

DacStructInit

DmaStructInit,

RccAhb1PeriphClockCmd

RccApb1PeriphClockCmd

GpioInit

TimPrescalerConfig

TimSetAutoReload

TimSelectOutputTrigger

Timcmd

TmrInitDac

DacCmd

#### 10.14.1.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to DtoaHwInit

##### 10.14.1.7.1 daugwydtoa-DtoaHwInit-LLR-001

Requirement ID: H398-LLD-GWY-FNC-1441

The function shall do the following:

-call TmrInitDac

-initialize the DAC peripheral to their default values by calling 'DacStructInit' with parameter dac init structure.

##### 10.14.1.7.2 daugwydtoa-DtoaHwInit-LLR-002

Requirement ID: H398-LLD-GWY-FNC-1442

The function shall initialize the DMA to their default values by calling 'DmaStructInit' with parameter dma init structure with members pre-initialized as follows

a) set dma\_channel to M\_DMA\_CHANNEL\_0,

b) set dma\_peripheral\_baseaddr to M\_ZERO,

c) set dma\_memory0\_baseaddr to M\_ZERO,

d) set dma\_dir to M\_DMA\_DIR\_PERIPHERAL\_TO\_MEMORY,

e) set dma\_buffersize to M\_ZERO,

f) set dma\_peripheral\_inc to M\_DMA\_PERIPHERALINC\_DISABLE,

g) set dma\_memory\_inc to M\_DMA\_MEMORYINC\_DISABLE,

h) set dma\_peripheral\_datasize to M\_DMA\_PERIPHERAL\_DATASIZE\_BYTE,

i) set dma\_memory\_datasize to M\_ZERO,

j) set dma\_mode to M\_DMA\_MODE\_NORMAL,

k) set dma\_priority to M\_DMA\_PRIORITY\_LOW,

l) set dma\_fifo\_mode to M\_DMA\_FIFOMODE\_DISABLE,

m) set dma\_fifo\_threshold to M\_ZERO,

n) set dma\_memory\_burst to M\_DMA\_MEMORYBURST\_SINGLE,

o) set dma\_peripheral\_burst to M\_DMA\_PERIPHERALBURST\_SINGLE.

##### 10.14.1.7.3 daugwydtoa-DtoaHwInit-LLR-003

Requirement ID: H398-LLD-GWY-FNC-1443

The function shall enable DMA 1 and GPIO port A clock by calling 'RccAhb1PeriphClockCmd' with parameters M\_RCC\_AHB1PERIPH\_DMA1 | M\_RCC\_AHB1PERIPH\_GPIOA and ENABLE.

##### 10.14.1.7.4 daugwydtoa-DtoaHwInit-LLR-004

Requirement ID: H398-LLD-GWY-FNC-1444

The function shall enable DAC peripheral clock by calling 'RccApb1PeriphClockCmd' with parameters M\_RCC\_APB1PERIPH\_DAC and ENABLE.

##### 10.14.1.7.5 daugwydtoa-DtoaHwInit-LLR-005

Requirement ID: H398-LLD-GWY-FNC-1445

The function shall configure pin 4 of GPIO port A by calling 'GpioInit' with parameters M\_GPIOA and gpio init structure with the members initialized as follows

a) set GPIO port output speed register(gpio\_speed) with GPIO\_SPEED\_50MHZ,

b) set gpio\_pin to M\_GPIO\_PIN\_4(gpio pin 4),

c) set GPIO port mode register(gpio\_mode) to GPIO\_MODE\_AN,

d) set GPIO port output type register(gpio\_otype) to GPIO\_OTYPE\_PP and

e) set GPIO port pull-up/pull-down register(gpio\_pupd) to GPIO\_PUPD\_NOPULL.

##### 10.14.1.7.6 daugwydtoa-DtoaHwInit-LLR-006

Requirement ID: H398-LLD-GWY-FNC-1446

The function shall initialize the DAC channel 1 with reset values by calling 'DacInit' with parameters M\_DAC\_CHANNEL\_1 and dac init structure with the members initialized as follows

a) set dac\_trigger to M\_DAC\_TRIGGER\_NONE,

b) set dac\_wavegeneration to M\_DAC\_WAVEGENERATION\_NONE, and

c) set dac\_outputbuffer to M\_DAC\_OUTPUTBUFFER\_ENABLE

##### 10.14.1.7.7 daugwydtoa-DtoaHwInit-LLR-007

Requirement ID: H398-LLD-GWY-FNC-1447

The function shall enable the DAC channel 1 by calling 'DacCmd' with parameters M\_DAC\_CHANNEL\_1 and ENABLE.

##### 10.14.1.7.8 daugwydtoa-DtoaHwInit-LLR-008

Requirement ID: H398-LLD-GWY-FNC-1448

The function shall call the RccApb1PeriphClockCmd with parameters M\_RCC\_APB1PERIPH\_TIM7 and ENABLE to enable TIM6 Periph clock.

##### 10.14.1.7.9 daugwydtoa-DtoaHwInit-LLR-009

Requirement ID: H398-LLD-GWY-FNC-1449

The function shall call the TimPrescalerConfig to Configure prescaler for timer 6.with parameters

M\_TIM7, M\_TMR\_7\_SCALE - M\_ONE, M\_TIM\_PSCRELOADMODE\_UPDATE.

##### 10.14.1.7.10 daugwydtoa-DtoaHwInit-LLR-010

Requirement ID: H398-LLD-GWY-FNC-1450

The function shall call the TimSetAutoReload to Set timer 6 to counter reset value with parameters

M\_TIM7, M\_TMR\_7\_PERIOD – M\_ONE.

##### 10.14.1.7.11 daugwydtoa-DtoaHwInit-LLR-011

Requirement ID: H398-LLD-GWY-FNC-1451

The function shall call the TimSelectOutputTrigger to TIM6 TRGO selection with parameters

M\_TIM7, M\_TIM\_TRGOSOURCE\_UPDATE.

##### 10.14.1.7.12 daugwydtoa-DtoaHwInit-LLR-012

Requirement ID: H398-LLD-GWY-FNC-1452

The function shall call the TimCmd to TIM6 counter enable with parameters

M\_TIM7 and ENABLE.

## 10.15 daugwyetitimer

The daugwyetitimer CSC contains implementation of timer routines.

### 10.15.1 ETIinit

Low Level Design Details about CSU ETIinit will follow in the sub sections.

#### 10.15.1.1 Brief Description

ETIinit installs Semaphore into timebase and creates Timer OS task.

#### 10.15.1.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.15.1.3 List of global variables accessed and modified

Accessed : Eti\_taskstk

Modified : Eti\_taskstk

#### 10.15.1.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.15.1.5 Return Value

None

#### 10.15.1.6 Other CSUs called by this CSU

TbaseTaskSignaling

OsSemCreate

OsTaskCreate

TimerTask

RterrError

RterrForever

#### 10.15.1.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to ETIinit.

##### 10.15.1.7.1 daugwyetitimer-ETIinit-LLR-001

Requirement ID: H398-LLD-GWY-FNC-1462

The function shall install task signaling parameters for the ETIinit task by calling 'TbaseTaskSignaling' with parameters

a) M\_ETI\_TASK\_TICKS as Task Ticks and

b) return value of function OsSemCreate with parameter M\_ZERO to semaphore timers stack.

##### 10.15.1.7.2 daugwyetitimer-ETIinit-LLR-002

Requirement ID: H398-LLD-GWY-FNC-1463

The function shall loop from stack counter is M\_ZERO to M\_ETI\_TASK\_STK\_SIZE and initialize Eti\_taskstk of index stack counter with M\_CBIT\_TASK\_STK\_VAL.

##### 10.15.1.7.3 daugwyetitimer-ETIinit-LLR-003

Requirement ID: H398-LLD-GWY-FNC-1464

The function shall do the following:

-create the Init Task by calling 'OsTaskCreate' with parameters (reference of function pointer TimerTask, M\_HW\_NULL, base address of array Eti\_taskstk with index as M\_ETI\_TASK\_STK\_SIZE and M\_ETI\_TASK\_PRIO).

-call the function RterrError with parameter (BIT\_ERR, M\_THREE, reference of function pointer RterrForever) when the function OsTaskCreate returns other then M\_OS\_NO\_ERR, otherwise do nothing.

### 10.15.2 TimerTask

Low Level Design Details about CSU TimerTask will follow in the sub sections.

#### 10.15.2.1 Brief Description

The TimerTask is the routine to driver timer task.

#### 10.15.2.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.15.2.3 List of global variables accessed and modified

Accessed : Xram\_data\_buffer

Modified : Xram\_data\_buffer

#### 10.15.2.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.15.2.5 Return Value

None

#### 10.15.2.6 Other CSUs called by this CSU

OsSemPend

#### 10.15.2.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TimerTask.

##### 10.15.2.7.1 daugwyetitimer-TimerTask-LLR-002

Requirement ID: H398-LLD-GWY-FNC-1474

The function shall loop infinitely and perform the following operations within loop.

1. Call OsSemPend with parameters semaphore timers task, M\_ZERO and address of u8err.
2. Increment the seconds of Xram\_data\_buffer.

## 10.16 daugwyexmon

The daugwyexmon CSC contains routines for defining monitor routines.

### 10.16.1 CheckCondition

Low Level Design Details about CSU CheckCondition will follow in the sub sections.

#### 10.16.1.1 Brief Description

The CheckCondition function checks the condition of the aircraft or engine.

#### 10.16.1.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.16.1.3 List of global variables accessed and modified

Accessed : Op\_mode

Modified : None

#### 10.16.1.4 Parameter list (Input/Output)

Inputs : T\_UINT16 cond - Aircraft or Engine condition

Outputs : None

#### 10.16.1.5 Return Value

T\_BOOL - Returns status

#### 10.16.1.6 Other CSUs called by this CSU

None

#### 10.16.1.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to CheckCondition.

##### 10.16.1.7.1 daugwyexmon-CheckCondition-LLR-001

Requirement ID: H398-LLD-GWY-FNC-1486

The function shall return aircraft or engine status as TRUE, when the current aircraft or engine condition (cond) is STATE\_ALWAYS.

##### 10.16.1.7.2 daugwyexmon-CheckCondition-LLR-002

Requirement ID: H398-LLD-GWY-FNC-1487

The function shall return aircraft or engine status as FALSE, when the current aircraft or engine condition (cond) is STATE\_SHUTDOWN.

##### 10.16.1.7.3 daugwyexmon-CheckCondition-LLR-003

Requirement ID: H398-LLD-GWY-FNC-1488

The function shall return aircraft or engine status as startup\_mode of Op\_mode, when the current aircraft or engine condition(cond) is STATE\_START.

##### 10.16.1.7.4 daugwyexmon-CheckCondition-LLR-004

Requirement ID: H398-LLD-GWY-FNC-1489

The function shall return aircraft or engine status as AEO of Op\_mode, when the current aircraft or engine condition(cond) is STATE\_AEO.

##### 10.16.1.7.5 daugwyexmon-CheckCondition-LLR-005

Requirement ID: H398-LLD-GWY-FNC-1490

The function shall return aircraft or engine status as takeoff of Op\_mode, when the current aircraft or engine condition (cond) is STATE\_TAKEOFF.

##### 10.16.1.7.6 daugwyexmon-CheckCondition-LLR-006

Requirement ID: H398-LLD-GWY-FNC-1491

The function shall return aircraft or engine status as TRUE, when the current aircraft or engine condition (cond) is STATE\_TAKEOFF\_AEO, and AEO of Op\_mode AND takeoff of Op\_mode is TRUE.

##### 10.16.1.7.7 daugwyexmon-CheckCondition-LLR-007

Requirement ID: H398-LLD-GWY-FNC-1492

The function shall return aircraft or engine status as FALSE, when the current aircraft or engine condition (cond) is STATE\_TAKEOFF\_AEO, and AEO of Op\_mode AND takeoff of Op\_mode is FALSE.

##### 10.16.1.7.8 daugwyexmon-CheckCondition-LLR-008

Requirement ID: H398-LLD-GWY-FNC-1493

The function shall return aircraft or engine status as TRUE, when the current aircraft or engine condition (cond) is STATE\_OEI1, and OEI\_renc of Op\_mode AND NOT of OEI\_lenc of Op\_mode is TRUE.

##### 10.16.1.7.9 daugwyexmon-CheckCondition-LLR-009

Requirement ID: H398-LLD-GWY-FNC-1494

The function shall return aircraft or engine status as FALSE, when the current aircraft or engine condition (cond) is STATE\_OEI1, and OEI\_renc of Op\_mode AND NOT of OEI\_lenc of Op\_mode is FALSE.

##### 10.16.1.7.10 daugwyexmon-CheckCondition-LLR-010

Requirement ID: H398-LLD-GWY-FNC-1495

The function shall return aircraft or engine status as TRUE, when the current aircraft or engine condition (cond) is STATE\_OEI2, and NOT of OEI\_renc of Op\_mode AND OEI\_lenc of Op\_mode is TRUE.

##### 10.16.1.7.11 daugwyexmon-CheckCondition-LLR-011

Requirement ID: H398-LLD-GWY-FNC-1496

The function shall return aircraft or engine status as FALSE, when the current aircraft or engine condition (cond) is STATE\_OEI2, and NOT of OEI\_renc of Op\_mode AND OEI\_lenc of Op\_mode is FALSE.

##### 10.16.1.7.12 daugwyexmon-CheckCondition-LLR-012

Requirement ID: H398-LLD-GWY-FNC-1497

The function shall do nothing

When the current aircraft or engine condition (cond) is other than STATE\_ALWAYS, STATE\_SHUTDOWN, STATE\_START, STATE\_AEO, STATE\_TAKEOFF, STATE\_TAKEOFF\_AEO, STATE\_OEI1, STATE\_OEI2.

##### 10.16.1.7.13 daugwyexmon-CheckCondition-LLR-013

Requirement ID: H398-LLD-GWY-FNC-6147

Returns status to main functionality

### 10.16.2 CheckRange

Low Level Design Details about CSU CheckRange will follow in the sub sections.

#### 10.16.2.1 Brief Description

The CheckRange function checks the range secondary parameters.

#### 10.16.2.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.16.2.3 List of global variables accessed and modified

Accessed : Ex\_data

Modified : None

#### 10.16.2.4 Parameter list (Input/Output)

Inputs : T\_FLOAT32 low - Lower boundary range

T\_FLOAT32 high - Higher boundary range

T\_UINT16 param - secondary parameter value to check range

Outputs : None

#### 10.16.2.5 Return Value

T\_BOOL - Returns the status of range check.

#### 10.16.2.6 Other CSUs called by this CSU

None

#### 10.16.2.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to CheckRange.

##### 10.16.2.7.1 daugwyexmon-CheckRange-LLR-001

Requirement ID: H398-LLD-GWY-FNC-1506

The function shall return TRUE when secondary parameter (param) is BLANK.

##### 10.16.2.7.2 daugwyexmon-CheckRange-LLR-002

Requirement ID: H398-LLD-GWY-FNC-1507

The function shall do the following:

- Set Ex\_data with index param to temporary value

- Set return variable to TRUE, when (((low is equal to M\_FP\_ZERO by calling the macro M\_FP\_COMPARE)) OR (low is less than M\_FP\_ZERO)) AND ((high is equal to M\_FP\_ZERO by calling the macro M\_FP\_COMPARE)) OR (high is less than M\_FP\_ZERO))

##### 10.16.2.7.3 daugwyexmon-CheckRange-LLR-003

Requirement ID: H398-LLD-GWY-FNC-1508

The function shall set the return variable to TRUE when,

- (((low is equal to M\_FP\_ZERO by calling the macro M\_FP\_COMPARE)) OR (low is less than M\_FP\_ZERO)) AND ((high is equal to M\_FP\_ZERO by calling the macro M\_FP\_COMPARE) OR (high is less than M\_FP\_ZERO)) returns FALSE.

- (((temporary value is equal to low by calling the macro M\_FP\_COMPARE)) OR (temporary value is greater than low)) AND (temporary value is less than high)).

##### 10.16.2.7.4 daugwyexmon-CheckRange-LLR-004

Requirement ID: H398-LLD-GWY-FNC-1509

The function shall return the return variable.

### 10.16.3 Chklimit

Low Level Design Details about CSU Chklimit will follow in the sub sections.

#### 10.16.3.1 Brief Description

The Chklimit function checks limit for being exceeded.

#### 10.16.3.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.16.3.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.16.3.4 Parameter list (Input/Output)

Inputs : T\_LIMIT\_DATA \*plim - Parameter limit

T\_FLOAT32 parval - Parameter value

T\_LIM\_OTH\_ENTRY\_DATA \*pothlim - Parameter limit

Outputs : T\_LIMIT\_DATA \*plim - Parameter limit

#### 10.16.3.5 Return Value

T\_BOOL - Returns status

#### 10.16.3.6 Other CSUs called by this CSU

CheckCondition

CheckRange

#### 10.16.3.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to Chklimit.

##### 10.16.3.7.1 daugwyexmon-Chklimit-LLR-001

Requirement ID: H398-LLD-GWY-FNC-1518

The function shall do nothing when

-(CheckCondition function with parameter (spcond of plim) AND CheckRange function with parameters(low of plim, high of plim and var of plim)returns TRUE.

-lim\_type of plim is LIMIT\_HI\_LOW.

##### 10.16.3.7.2 daugwyexmon-Chklimit-LLR-011

Requirement ID: H398-LLD-GWY-FNC-5806

The function shall do the following when Lim\_oth\_type of pothlim is LIMIT\_FIXED:

1. Set limit to lim\_oth\_val of pothlim and limit\_found to TRUE when xtimer of plim is greater than equal to multiplication of (start\_oth\_t of pothlim) and M\_TICS\_PER\_SECOND and end\_oth\_t of pothlim equal to M\_ZERO. Otherwise do nothing.

2. -(CheckCondition function with parameter (spcond of plim) AND CheckRange function with parameters(low of plim, high of plim and var of plim)returns TRUE.

-lim\_type of plim is LIMIT\_OTHERS.

-value is greater than lim\_val of plim.

-The function shall loop from Zero until Lim\_oth\_type of pothlim is NOT equal to LIMIT\_NIL AND limit\_found equal to False.

##### 10.16.3.7.3 daugwyexmon-Chklimit-LLR-012

Requirement ID: H398-LLD-GWY-FNC-5807

The function shall do the following when Lim\_oth\_type of pothlim is LIMIT\_FIXED:

1. Set limit to lim\_oth\_val of pothlim and limit\_found to TRUE when xtimer of plim is lesser than multiplication of (end\_oth\_t of pothlim) and M\_TICS\_PER\_SECOND and if start\_oth\_t of pothlim equal to M\_ZERO. Otherwise do nothing.

2. xtimer of plim incremented by M\_ONE.

when

-(CheckCondition function with parameter (spcond of plim) AND CheckRange function with parameters(low of plim, high of plim and var of plim)returns TRUE.

-lim\_type of plim is LIMIT\_OTHERS.

-value is greater than lim\_val of plim.

-The function shall loop from Zero until Lim\_oth\_type of pothlim is NOT equal to LIMIT\_NIL AND limit\_found equal to False.

##### 10.16.3.7.4 daugwyexmon-Chklimit-LLR-013

Requirement ID: H398-LLD-GWY-FNC-5808

The function shall do the following when Lim\_oth\_type of pothlim is LIMIT\_FIXED:

1. Set limit to lim\_oth\_val of pothlim and limit\_found to TRUE when (xtimer of plim is greater than equal to multiplication of (start\_oth\_t of pothlim) and M\_TICS\_PER\_SECOND) AND (xtimer of plim is less than multiplication of (end\_oth\_t of pothlim) and M\_TICS\_PER\_SECOND) is True. Otherwise do nothing.

2. xtimer of plim incremented by M\_ONE.

when

-(CheckCondition function with parameter (spcond of plim) AND CheckRange function with parameters(low of plim, high of plim and var of plim)returns TRUE.

-lim\_type of plim is LIMIT\_OTHERS.

-value is greater than lim\_val of plim.

-The function shall loop from Zero until Lim\_oth\_type of pothlim is NOT equal to LIMIT\_NIL AND limit\_found equal to False.

##### 10.16.3.7.5 daugwyexmon-Chklimit-LLR-014

Requirement ID: H398-LLD-GWY-FNC-5809

The function shall do the following when Lim\_oth\_type of pothlim is LIMIT\_RANGE:

1. Do nothing when end\_oth\_t of pothlim is equal to M\_ZERO and also start\_oth\_t of pothlim is equal to M\_ZERO.

2. Set limit to multiplication of ((lim\_oth\_conv\_val of pothlim), xtimer of plim) is addition with lim\_oth\_val of pothlim and also sets limit\_found to True, when ((xtimer of plim is greater than equal to multiplication of (start\_oth\_t of pothlim) and M\_TICS\_PER\_SECOND) AND (xtimer of plim is less than multiplication of (end\_oth\_t of pothlim) and M\_TICS\_PER\_SECOND)) is True. Otherwise do nothing.

3. xtimer of plim incremented by M\_ONE.

when

-(CheckCondition function with parameter (spcond of plim) AND CheckRange function with parameters(low of plim, high of plim and var of plim)returns TRUE.

-lim\_type of plim is LIMIT\_OTHERS.

-value is greater than lim\_val of plim.

-The function shall loop from Zero until Lim\_oth\_type of pothlim is NOT equal to LIMIT\_NIL AND limit\_found equal to False.

##### 10.16.3.7.6 daugwyexmon-Chklimit-LLR-015

Requirement ID: H398-LLD-GWY-FNC-6192

The function shall do nothing and come out of loop when Lim\_oth\_type of pothlim is other than LIMIT\_FIXED and LIMIT\_RANGE.

- xtimer of plim incremented by M\_ONE.

when

-(CheckCondition function with parameter (spcond of plim) AND CheckRange function with parameters(low of plim, high of plim and var of plim)returns TRUE.

-lim\_type of plim is LIMIT\_OTHERS.

-value is greater than lim\_val of plim.

-The function shall loop from Zero until Lim\_oth\_type of pothlim is NOT equal to LIMIT\_NIL AND limit\_found equal to False.

-Set plim of xtimer to M\_ZERO

Set Value to M\_FP\_ZERO when Value is not greater than plim of lim\_val.

##### 10.16.3.7.7 daugwyexmon-Chklimit-LLR-002

Requirement ID: H398-LLD-GWY-FNC-1519

The function shall do nothing when

-(CheckCondition function with parameter (spcond of plim) AND CheckRange function with parameters(low of plim, high of plim and var of plim)returns TRUE.

-lim\_type of plim is other than LIMIT\_HI\_LOW, LIMIT\_OTHERS.

##### 10.16.3.7.8 daugwyexmon-Chklimit-LLR-004

Requirement ID: H398-LLD-GWY-FNC-1521

This function shall do the following when (CheckCondition function with parameter (spcond of plim) AND CheckRange function with parameters(low of plim, high of plim and var of plim)returns TRUE

-set limit value to Value and parval to limit value when parameter limiting above or below (hilo\_lim of plim) is equal to RANGE\_BELOW\_OR\_EQUAL OR parameter limiting above or below(hilo\_lim of plim) is equal to RANGE\_BELOW.

##### 10.16.3.7.9 daugwyexmon-Chklimit-LLR-005

Requirement ID: H398-LLD-GWY-FNC-1522

This function shall do the following when (CheckCondition function with parameter (spcond of plim) AND CheckRange function with parameters(low of plim, high of plim and var of plim)returns TRUE

- Set limit exceeded to TRUE when parameter limiting above or below (hilo\_lim of plim) is equal to RANGE\_BELOW\_OR\_EQUAL OR parameter limiting above or below(hilo\_lim of plim) is equal to RANGE\_ABOVE\_OR\_EQUAL and

((value is equal to limit value by calling the macro M\_FP\_COMPARE) OR (value is greater than limit value))

##### 10.16.3.7.10 daugwyexmon-Chklimit-LLR-006

Requirement ID: H398-LLD-GWY-FNC-1523

This function shall do the following when (CheckCondition function with parameter (spcond of plim) AND CheckRange function with parameters(low of plim, high of plim and var of plim)returns TRUE

- Set limit exceeded to TRUE, when parameter limiting above or below (hilo\_lim of plim) is other than RANGE\_BELOW\_OR\_EQUAL OR parameter limiting above or below(hilo\_lim of plim) is other than RANGE\_ABOVE\_OR\_EQUAL and Value is greater than limit value

##### 10.16.3.7.11 daugwyexmon-Chklimit-LLR-007

Requirement ID: H398-LLD-GWY-FNC-1524

This function shall do the following when (CheckCondition function with parameter (spcond of plim) AND CheckRange function with parameters(low of plim, high of plim and var of plim)returns TRUE and limit exceeded is TRUE.

a) Set result variable to TRUE when Parameter timer (ptimer of plim) is greater than or equal to product of parameter maximum time (maxtime of plim) and M\_TICS\_PER\_SECOND.

b) Set exceed upper of parameter limits (exceed\_upper of plim) to TRUE when parameter limit of limit high is greater than M\_FP\_ZERO and ((value is equal to lim\_high of plim by calling the macro M\_FP\_COMPARE) OR (value is greater than lim\_high of plim))

c) Increment the Parameter timer (ptimer of plim) by M\_ONE

##### 10.16.3.7.12 daugwyexmon-Chklimit-LLR-008

Requirement ID: H398-LLD-GWY-FNC-1525

This function shall set the parameter timer (ptimer of plim) to M\_ZERO when (CheckCondition function with parameter (spcond of plim) AND CheckRange function with parameters(low of plim, high of plim and var of plim)returns TRUE and limit exceeded is FALSE.

##### 10.16.3.7.13 daugwyexmon-Chklimit-LLR-009

Requirement ID: H398-LLD-GWY-FNC-1526

The function shall set the parameter time (ptimer of plim) to M\_ZERO, when (CheckCondition function with parameter (spcond of plim) AND CheckRange function with parameters(low of plim, high of plim and var of plim)returns FALSE.

##### 10.16.3.7.14 daugwyexmon-Chklimit-LLR-010

Requirement ID: H398-LLD-GWY-FNC-1527

This function shall return the result variable.

### 10.16.4 ValidateParam

Low Level Design Details about CSU ValidateParam will follow in the sub sections.

#### 10.16.4.1 Brief Description

The ValidateParam function validates parameters limit .

#### 10.16.4.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.16.4.3 List of global variables accessed and modified

Accessed : Tx\_a429\_bnr

Modified : Ex\_data

#### 10.16.4.4 Parameter list (Input/Output)

Inputs : T\_UINT32 param - Function parameter

T\_UINT32 label - Label value

Outputs : None

#### 10.16.4.5 Return Value

None

#### 10.16.4.6 Other CSUs called by this CSU

#### 10.16.4.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to ValidateParam.

##### 10.16.4.7.1 daugwyexmon-ValidateParam-LLR-001

Requirement ID: H398-LLD-GWY-FNC-1536

This function shall do the following when status of Tx\_a429\_bnr with index label is OK

- Set data of Tx\_a429\_bnr with index label to Ex\_data with index param

- Set Ex counter with index as param to M\_ZERO

##### 10.16.4.7.2 daugwyexmon-ValidateParam-LLR-002

Requirement ID: H398-LLD-GWY-FNC-1537

This function shall set Ex\_data with index param to M\_FP\_ZERO when, status of Tx\_a429\_bnr with index label is not equal to OK and Ex counter with index param is greater than product of M\_ONE and M\_TICS\_PER\_SECOND.

##### 10.16.4.7.3 daugwyexmon-ValidateParam-LLR-003

Requirement ID: H398-LLD-GWY-FNC-1538

This function shall increment the Ex counter with index param by one when, status of Tx\_a429\_bnr with index label is not equal to OK and Ex counter with index param is less than or equal to product of M\_ONE and M\_TICS\_PER\_SECOND.

### 10.16.5 ExceedancePacket

Low Level Design Details about CSU ExceedancePacket will follow in the sub sections.

#### 10.16.5.1 Brief Description

The ExceedancePacket function frames exceedance packet in ARINC 429 Format.

#### 10.16.5.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.16.5.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.16.5.4 Parameter list (Input/Output)

Inputs : T\_UINT8 u8\_label - 8 bits label data

T\_SINT32 i32\_data - 32 bits data

T\_UINT8 u8\_sigbits - 8 bits significant bits data

T\_UINT16 id - data ID

T\_SINT32 a429\_sdi - a429 SDI

Outputs : T\_SINT32 i32\_data - 32 bits data

T\_UINT8 u8\_sigbits - 8 bits significant bits data

#### 10.16.5.5 Return Value

T\_SINT32 - Returns 32 bit arinc word

#### 10.16.5.6 Other CSUs called by this CSU

HwAbs32

#### 10.16.5.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to ExceedancePacket.

##### 10.16.5.7.1 daugwyexmon-ExceedancePacket-LLR-001

Requirement ID: H398-LLD-GWY-FNC-1547

The function shall do the following:

- Set i32\_data with i32\_data left shifted by the return value of the function HwAbs32 with parameter (M\_TWENTY subtracted with u8\_sigbits) Bitwise OR id left shifted by M\_TWO Bitwise OR a429\_sdi.

- Update u8\_sigbits with addition of u8\_sigbits and (subtraction of M\_TWENTY with u8\_sigbits)

- Set 32 bit arinc word with M\_EXCEEDANCE\_PACKET\_MASK Bitwise OR u8\_label.

- Set 32 bit arinc word with 32 bit arinc word Bitwise OR i32\_data left shifed by (subtraction of M\_THIRTYONE with u8\_sigbits).

- Return 32 bit arinc word

### 10.16.6 ExceedanceToCMU

Low Level Design Details about CSU ExceedanceToCMU will follow in the sub sections.

#### 10.16.6.1 Brief Description

The function ExceedanceToCMU sends exceedance packet to the configuration management unit.

#### 10.16.6.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.16.6.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.16.6.4 Parameter list (Input/Output)

Inputs : T\_LIMIT\_ENTRY\_DATA \*tbl\_ex-- Exceedance data table

Outputs : None

#### 10.16.6.5 Return Value

None

#### 10.16.6.6 Other CSUs called by this CSU

HwCopy

A825Xmit

ExceedancePacket

OsTimeDly

#### 10.16.6.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to ExceedanceToCMU.

##### 10.16.6.7.1 daugwyexmon-ExceedanceToCMU-LLR-001

Requirement ID: H398-LLD-GWY-FNC-1556

This function shall do the following:

1. Set lcc of exceedance message structure to NOC.
2. Set sfid of bid of sid of exceedance message structure to M\_HOWELL\_DAU\_FID.
3. Set rsd of bid of sid of exceedance message structure to M\_ZERO.
4. Set lcl of bid of sid of exceedance message structure to M\_ONE.
5. Set pvt of bid of sid of exceedance message structure to M\_ONE.
6. Set doc of bid of sid of exceedance message structure to DOC\_A429\_EXC.
7. Set u8\_paysize of exceedance message structure to M\_EIGHT.

##### 10.16.6.7.2 daugwyexmon-ExceedanceToCMU-LLR-002

Requirement ID: H398-LLD-GWY-FNC-1557

This function shall

1. Set the exceedance type data to E\_Sys of tbl\_ex is bitwise AND with M\_LABEL\_301\_TAGSYSIDS\_MASK then left shift by M\_SIX times Bitwise OR operation with E\_Param of tbl\_ex is bitwise AND with M\_LABEL\_301\_TAGSYSIDS\_MASK then left shift by M\_FOUR times Bitwise OR operation with cEventId of tbl\_ex is bitwise AND with M\_BYTE\_LSB\_MASK.
2. Call ExceedancePacket with parameters M\_EXCEEDANCE\_LABEL\_0301, exceedance type data,M\_EIGHT, M\_ZERO, M\_ZERO and return value ExceedancePacket is stored in arinc word.
3. Call HwCopy with parameters (reference to u8\_payload with zeroth index of exceedance message structure, address of arinc word ,M\_FOUR.

##### 10.16.6.7.3 daugwyexmon-ExceedanceToCMU-LLR-003

Requirement ID: H398-LLD-GWY-FNC-1558

This function shall set exceedance type data to divided value of peak of tbl\_ex with M\_PEAK\_WHEN\_NOT\_IN\_MGT\_VAL when, E\_Param of tbl\_ex is not equal to MGT\_P.

##### 10.16.6.7.4 daugwyexmon-ExceedanceToCMU-LLR-004

Requirement ID: H398-LLD-GWY-FNC-1559

This function shall set exceedance type data to divided value of peak of tbl\_ex with M\_PEAK\_WHEN\_IN\_MGT\_VAL when, E\_Param of tbl\_ex is equal to MGT\_P.

##### 10.16.6.7.5 daugwyexmon-ExceedanceToCMU-LLR-005

Requirement ID: H398-LLD-GWY-FNC-1560

This function shall call ExceedancePacket is called with the parameters M\_EXCEEDANCE\_LABEL\_0302, exceedance type data,M\_DEC\_FIFTEEN, M\_ZERO, M\_ZERO and then assigned the returning value of that function arinc word variable.

Function HwCopy is called with parameters (u8\_payload of M\_FOUR index of exceedance message structure, address of arinc word(return value of ExceedancePacket function), Value M\_FOUR) ,to copy the exceedance type data to payload.

Function A825Xmit is called with the address of exceedance message structure in order to transmit exceedance type and peak to CMU for logging.

Function OsTimeDly is called with parameter as M\_ONE.

Exceedance duration is calculated by dividing duration of tbl\_ex with M\_APP\_RATE and M\_PEAK\_WHEN\_IN\_MGT\_VAL.

##### 10.16.6.7.6 daugwyexmon-ExceedanceToCMU-LLR-006

Requirement ID: H398-LLD-GWY-FNC-1561

This function shall call ExceedancePacket is called with the parameters M\_EXCEEDANCE\_LABEL\_0303, exceedance type data,M\_SIXTEEN, M\_ZERO, M\_ZERO and then assigned the returning value of that function arinc word variable.

Function HwCopy is called with parameters (u8\_payload of M\_ZERO index of exceedance message structure, address of arinc word(return value of ExceedancePacket function), Value M\_FOUR) ,to copy the exceedance type data to payload.

Exceedance start timer start\_t of tbl\_ex is set to exceedance type data.

##### 10.16.6.7.7 daugwyexmon-ExceedanceToCMU-LLR-007

Requirement ID: H398-LLD-GWY-FNC-1562

This function shall call ExceedancePacket is called with the parameters M\_EXCEEDANCE\_LABEL\_0306, exceedance type data, M\_SEVENTEEN , M\_ZERO, M\_ZERO and then assigned the returning value of that function arinc word variable.

Function HwCopy is called with parameters (u8\_payload of M\_FOUR index of reference of exceedance message structure, address of arinc word(return value of ExceedancePacket function), Value M\_FOUR) ,to copy the exceedance type data to payload.

A825Xmit function is called with the address of exceedance message structure.

Exceedance date time date\_t of tbl\_ex is set to exceedance type data.

##### 10.16.6.7.8 daugwyexmon-ExceedanceToCMU-LLR-008

Requirement ID: H398-LLD-GWY-FNC-1563

This function shall call ExceedancePacket is called with the parameterS M\_EXCEEDANCE\_LABEL\_0307, exceedance type data, M\_EIGHTEEN , M\_ZERO, M\_ZERO and then assigned the returning value of that function arinc word variable.

Function HwCopy is called with parameters (u8\_payload of M\_ZERO index of exceedance message structure, address of arinc word(return value of ExceedancePacket function), Value M\_FOUR) ,to copy the exceedance type data to payload.

Arinc word is reset to M\_ZERO.

Function HwCopy is called with parameters (u8\_payload of M\_FOUR index of exceedance message structure, address of arinc word(return value of ExceedancePacket function), Value M\_FOUR) ,to copy the exceedance type data to payload.

A825Xmit function is called with the address of exceedance message structure.

### 10.16.7 Limmon

Low Level Design Details about CSU Limmon will follow in the sub sections.

#### 10.16.7.1 Brief Description

The function Limmon is for limiting moinitoring.

#### 10.16.7.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.16.7.3 List of global variables accessed and modified

Accessed : Tx\_a429\_bnr

Ex\_event

Ex\_cas

Cmu\_date

Cmu\_time

Ex\_data

Modified : None

#### 10.16.7.4 Parameter list (Input/Output)

Inputs : struct lim\_table \*limits-Limits Monitor Table

Outputs : struct lim\_table \*limits-Limits Monitor Table

#### 10.16.7.5 Return Value

None

#### 10.16.7.6 Other CSUs called by this CSU

ValidateParam

Chklimit

ExceedanceToCMU

HwCopy

#### 10.16.7.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to Limmon.

##### 10.16.7.7.1 daugwyexmon-Limmon-LLR-001

Requirement ID: H398-LLD-GWY-FNC-1572

This function shall call ValidateParam function to validate with parameters NR and LABEL\_040.

##### 10.16.7.7.2 daugwyexmon-Limmon-LLR-003

Requirement ID: H398-LLD-GWY-FNC-1574

This function shall call ValidateParam function to validate with parameters NP1 and LABEL\_0344\_1.

##### 10.16.7.7.3 daugwyexmon-Limmon-LLR-004

Requirement ID: H398-LLD-GWY-FNC-1575

This function shall call ValidateParam function to validate with parameters NP2 and LABEL\_0344\_2.

##### 10.16.7.7.4 daugwyexmon-Limmon-LLR-005

Requirement ID: H398-LLD-GWY-FNC-1576

This function shall call ValidateParam function to validate with parameters NG1 and LABEL\_0346\_1.

##### 10.16.7.7.5 daugwyexmon-Limmon-LLR-006

Requirement ID: H398-LLD-GWY-FNC-1577

This function shall call ValidateParam function to validate with parameters NG2 and LABEL\_0346\_2.

##### 10.16.7.7.6 daugwyexmon-Limmon-LLR-007

Requirement ID: H398-LLD-GWY-FNC-1578

This function shall call ValidateParam function to validate with parameters MGT1 and LABEL\_0345\_1.

##### 10.16.7.7.7 daugwyexmon-Limmon-LLR-008

Requirement ID: H398-LLD-GWY-FNC-1579

This function shall do the following

1. Set Ex\_data with index QE to M\_FP\_ZERO.

2. Call ValidateParam function to validate with parameters MGT2 and LABEL\_0345\_2.

##### 10.16.7.7.8 daugwyexmon-Limmon-LLR-009

Requirement ID: H398-LLD-GWY-FNC-1580

This function shall do the following

1. Call ValidateParam function to validate with parameters QE1 and LABEL\_0336\_1.

2. Set Ex\_data with indx QE to Ex\_data with index QE1 if Ex\_data with index QE1 is greater than Ex\_data with index QE.

##### 10.16.7.7.9 daugwyexmon-Limmon-LLR-010

Requirement ID: H398-LLD-GWY-FNC-1581

This function shall do the following

1. Call ValidateParam function to validate with parameters QE2 and LABEL\_0336\_2.

2. Set Ex\_data with indx QE to Ex\_data with index QE2 if Ex\_data with index QE2 is greater than Ex\_data with index QE.

##### 10.16.7.7.10 daugwyexmon-Limmon-LLR-002

Requirement ID: H398-LLD-GWY-FNC-1573

This function shall do the following

1. Call ValidateParam function to validate with parameters QM and LABEL\_0340.

2. Set Ex\_data with index QM to addition of (Ex\_data with index QE1 and Ex\_data with index QE2) when status of Tx\_a429\_bnr with index LABEL\_0340 is not equal to OK.

3. Set Ex\_data with index QM to M\_FP\_ONE\_TWENTY\_SEVEN\_PT\_NINE when Ex\_data with index QM is greaterthan M\_FP\_ONE\_TWENTY\_SEVEN\_PT\_NINE.

##### 10.16.7.7.11 daugwyexmon-Limmon-LLR-011

Requirement ID: H398-LLD-GWY-FNC-1582

The function shall loop from Zero until parm of limits is not equal to M\_NULL AND with entry of limits is not equal to M\_NULL do the following

a) set parameter value to dereference of parm of limits

b) set the parameter exceedance to false.

##### 10.16.7.7.12 daugwyexmon-Limmon-LLR-012

Requirement ID: H398-LLD-GWY-FNC-1583

This function shall loop from Zero until parm of limits is not equal to M\_NULL AND with entry of limits is not equal to M\_NULL and loop from entry of limits to CAS less than EOT minus 1 and perform the following:

a) Call Chklimit function with parameters (reference to limit of table, lim\_others of table and parameter value) and return value is assigned to exceed of table.

b) Set temporary cas event to temporary cas event Bitwise OR with exceed of table.

##### 10.16.7.7.13 daugwyexmon-Limmon-LLR-013

Requirement ID: H398-LLD-GWY-FNC-1584

The function shall do the following when loop from Zero until parm of limits is not equal to M\_NULL AND with entry of limits is not equal to M\_NULL and loop from entry of limits to CAS less than EOT minus 1 and NOT of prev\_exceed of table AND exceed of table is True:

1. set start\_t of table to Cmu\_time.
2. set date\_t of table to Cmu\_date.
3. set end\_t of table to start\_t of table.
4. set the duration of table to M\_ZERO.
5. set the peak of table to M\_FP\_ZERO.
6. set the exceed upper of limit of table to FALSE

##### 10.16.7.7.14 daugwyexmon-Limmon-LLR-014

Requirement ID: H398-LLD-GWY-FNC-1585

This function shall do the following when loop from Zero until parm of limits is not equal to M\_NULL AND with entry of limits is not equal to M\_NULL and loop from entry of limits to CAS less than EOT minus 1 and prev\_exceed of table AND NOT of exceed of table is True:

1. Set end\_t of table to data of Tx\_a429\_bnr with LABEL\_0150 index.
2. call the function ExceedanceToCMU with parameter as table when exceed upper of limit of table is equal to FALSE. otherwise set the exceed upper of limit of table to FALSE.

##### 10.16.7.7.15 daugwyexmon-Limmon-LLR-015

Requirement ID: H398-LLD-GWY-FNC-1586

This function shall loop from Zero until parm of limits is not equal to M\_NULL AND with entry of limits is not equal to M\_NULL and loop from entry of limits to CAS less than EOT minus 1 Increment the duration of table by one, when exceed of table is set to TRUE and duration of table is less than or equal to the value M\_MAX\_SECONDS\_VALUE substracted by M\_TWO.

##### 10.16.7.7.16 daugwyexmon-Limmon-LLR-016

Requirement ID: H398-LLD-GWY-FNC-1587

The function shall Set peak of table to parameter value, when parameter value is greater than peak of table and exceed of table is set to TRUE and Set exceed of table to prev\_exceed of table.

##### 10.16.7.7.17 daugwyexmon-Limmon-LLR-017

Requirement ID: H398-LLD-GWY-FNC-1588

This function shall loop from Zero until parm of limits is not equal to M\_NULL AND with entry of limits is not equal to M\_NULL and loop from entry of limits to CAS less than EOT minus 1 and perform exceed of limits bitwise OR with exceed of table and assigned to exceed of limits.

a) Call HwCopy function with parameters (Ex\_cas, temporary cas event , NUM\_OF\_CAS) to copy over the temporary CAS status items.

### 10.16.8 SetMonLimits

Low Level Design Details about SetMonLimits will follow in the sub sections.

#### 10.16.8.1 Brief Description

This function is to setMonlimits for monitoring from ACD

#### 10.16.8.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.16.8.3 List of global variables accessed and modified

Accessed: Cmdownload\_lp\_cmd\_data

Modified: Limits

#### 10.16.8.4 Parameter list (Input/Output)

Input: None

Output: None

#### 10.16.8.5 Return Value

None

#### 10.16.8.6 Other CSUs called by this CSU

None

#### 10.16.8.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to SetMonLimits

##### 10.16.8.7.1 daugwyexmon-SetMonLimits -LLR-001

Requirement ID: H398-LLD-GWY-FNC-6182

The function shall loop from M\_ZERO to num Of exc\_mon\_data of Cmdownload\_lp\_cmd\_data AND with entry of exc\_mon\_data with index as loop count of Cmdownload\_lp\_cmd\_data is not equal to M\_NULL and increment loop count perform the following:

The loop M\_ZERO to M\_MAX\_NO\_OF\_EXCEED\_ENTRY minus 1 AND CAS of entry with index as loop count of exc\_mon\_data with index as loop counter of Cmdownload\_lp\_cmd\_data is not equal to EOT and increment the loop counter.

1) CAS of entry with index as loop count of Limits of loop count to CAS with index as loop count of entry with index as loop count of exc\_mon\_data with index as loop count of Cmdownload\_lp\_cmd\_data.

2) E\_Param of entry with index as loop count of Limits of loop count to E\_Param of entry with index as loop count of exc\_mon\_data with index as loop count of Cmdownload\_lp\_cmd\_data.

3) E\_Sys of entry with index as loop count of Limits of loop count to E\_Sys of entry with index as loop count of exc\_mon\_data with index as loop count of Cmdownload\_lp\_cmd\_data.

4) cEventId of entry with index as loop count of Limits of loop count to cEventId of entry with index as loop count of exc\_mon\_data with index as loop count of Cmdownload\_lp\_cmd\_data.

5) date\_t of entry with index as loop count of Limits of loop count to date\_t of entry with index as loop count of exc\_mon\_data with index as loop count of Cmdownload\_lp\_cmd\_data.

6) duration of entry with index as loop count of Limits of loop count to duration of entry with index as loop count of exc\_mon\_data with index as loop count of Cmdownload\_lp\_cmd\_data.

7) end\_t of entry with index as loop count of Limits of loop count to end\_t of entry with index as loop count of exc\_mon\_data with index as loop count of Cmdownload\_lp\_cmd\_data.

8) exceed of entry with index as loop count of Limits of loop count to exceed of entry with index as loop count of exc\_mon\_data with index as loop count of Cmdownload\_lp\_cmd\_data.

9) limit.exceed\_upper of entry with index as loop count of Limits of loop count to limit.exceed\_upper of entry with index as loop count of exc\_mon\_data with index as loop count of Cmdownload\_lp\_cmd\_data.

10) limit.high of entry with index as loop count of Limits of loop count to limit.high of entry with index as loop count of exc\_mon\_data with index as loop count of Cmdownload\_lp\_cmd\_data.

11) limit.hilo\_lim of entry with index as loop count of Limits of loop count to limit.hilo\_lim of entry with index as loop count of exc\_mon\_data with index as loop count of Cmdownload\_lp\_cmd\_data.

12) limit.lim\_high of entry with index as loop count of Limits of loop count to limit.lim\_high of entry with index as loop count of exc\_mon\_data with index as loop count of Cmdownload\_lp\_cmd\_data.

13) limit.lim\_type of entry with index as loop count of Limits of loop count to limit.lim\_type of entry with index as loop count of exc\_mon\_data with index as loop count of Cmdownload\_lp\_cmd\_data.

14) limit.lim\_val of entry with index as loop count of Limits of loop count to limit.lim\_val of entry with index as loop count of exc\_mon\_data with index as loop count of Cmdownload\_lp\_cmd\_data.

15) limit.low of entry with index as loop count of Limits of loop count to limit.low of entry with index as loop count of exc\_mon\_data with index as loop count of Cmdownload\_lp\_cmd\_data.

16) limit.maxtime of entry with index as loop count of Limits of loop count to limit.maxtime of entry with index as loop count of exc\_mon\_data with index as loop count of Cmdownload\_lp\_cmd\_data.

17) limit.ptimer of entry with index as loop count of Limits of loop count to limit.ptimer of entry with index as loop count of exc\_mon\_data with index as loop count of Cmdownload\_lp\_cmd\_data.

18) limit.spcond of entry with index as loop count of Limits of loop count to limit.spcond of entry with index as loop count of exc\_mon\_data with index as loop count of Cmdownload\_lp\_cmd\_data.

19) limit.var of entry with index as loop count of Limits of loop count to limit.var of entry with index as loop count of exc\_mon\_data with index as loop count of Cmdownload\_lp\_cmd\_data.

20) limit.xtimer of entry with index as loop count of Limits of loop count to limit.xtimer of entry with index as loop count of exc\_mon\_data with index as loop count of Cmdownload\_lp\_cmd\_data.

21) peak of entry with index as loop count of Limits of loop count to peak of entry with index as loop count of exc\_mon\_data with index as loop count of Cmdownload\_lp\_cmd\_data.

22) prev\_exceed of entry with index as loop count of Limits of loop count to prev\_exceed of entry with index as loop count of exc\_mon\_data with index as loop count of Cmdownload\_lp\_cmd\_data.

23) start\_t of entry with index as loop count of Limits of loop count to start\_t of entry with index as loop count of exc\_mon\_data with index as loop count of Cmdownload\_lp\_cmd\_data.

##### 10.16.8.7.2 daugwyexmon-SetMonLimits-LLR-002

Requirement ID: H398-LLD-GWY-FNC-6387

The function shall loop from M\_ZERO to numOf\_exc\_mon\_data of Cmdownload\_lp\_cmd\_data AND with entry of exc\_mon\_data with index as loop count of Cmdownload\_lp\_cmd\_data is not equal to M\_NULL and increment loop count perform the following:

The loop M\_ZERO to M\_MAX\_NO\_OF\_EXCEED\_ENTRY minus 1 AND CAS of entry with index as loop count of exc\_mon\_data with index as loop counter of Cmdownload\_lp\_cmd\_data is not equal to EOT and increment the loop counter.

1) Set lim\_others of entry with index as loop count of Limits with index as loop count to lim\_others of entry with index as loop count exc\_mon\_data with index as loop count Cmdownload\_lp\_cmd\_data, when limit.lim\_type of entry with index as loop count of exc\_mon\_data with index as loop count of Cmdownload\_lp\_cmd\_data is equal to LIMIT\_OTHERS AND M\_NULL is not equal to lim\_others of entry with index as loop count of exc\_mon\_data with index as loop count of Cmdownload\_lp\_cmd\_data , Otherwise Set lim\_others of entry with index loop count of Limits with index loop count to M\_NULL.

##### 10.16.8.7.3 daugwyexmon-SetMonLimits-LLR-003

Requirement ID: H398-LLD-GWY-FNC-6388

The function shall loop from M\_ZERO to numOf\_exc\_mon\_data of Cmdownload\_lp\_cmd\_data AND with entry of exc\_mon\_data with index as loop count of Cmdownload\_lp\_cmd\_data is not equal to M\_NULL and increment loop count perform the following:

1) Set CAS of entry with index as loop count of Limits with index as loop count to CAS OF entry with index as loop count of exc\_mon\_data with index as loop count of Cmdownload\_lp\_cmd\_data.

2) Set E\_Param of entry with index as loop count of Limits with index as loop count to E\_Param of entry with index as loop count of exc\_mon\_data with index as loop count of Cmdownload\_lp\_cmd\_data.

3) Set E\_Sys of entry with index as loop count of Limits with index as loop count to E\_Sys of entry with index as loop count of exc\_mon\_data with index as loop count of Cmdownload\_lp\_cmd\_data.

4) Set cEventId of entry with index as loop count of Limits with index as loop count to cEventId of entry with index as loop count of exc\_mon\_data with index as loop count of Cmdownload\_lp\_cmd\_data.

5) Set date\_t of entry with index as loop count of Limits with index as loop count to date\_t of entry with index as loop count of exc\_mon\_data with index as loop count of Cmdownload\_lp\_cmd\_data.

6) Set duration of entry with index as loop count of Limits with index as loop count to duration of entry with index as loop count of exc\_mon\_data with index as loop count of Cmdownload\_lp\_cmd\_data.

7) Set end\_t of entry with index as loop count of Limits with index as loop count to end\_t of entry with index as loop count of exc\_mon\_data with index as loop count of Cmdownload\_lp\_cmd\_data.

8) Set exceed of entry with index as loop count of Limits with index as loop count to exceed of entry with index as loop count of exc\_mon\_data with index as loop count of Cmdownload\_lp\_cmd\_data.

9) Set limit.exceed\_upper of entry with index as loop count of Limits with index as loop count to limit.exceed\_upper of entry with index as loop count of exc\_mon\_data with index as loop count of Cmdownload\_lp\_cmd\_data.

10) Set limit.high of entry with index as loop count of Limits with index as loop count to limit.high of entry with index as loop count of exc\_mon\_data with index as loop count of Cmdownload\_lp\_cmd\_data.

11) Set limit.hilo\_lim of entry with index as loop count of Limits with index as loop count to limit.hilo\_lim of entry with index as loop count of exc\_mon\_data with index as loop count of Cmdownload\_lp\_cmd\_data.

12) Set limit.lim\_high of entry with index as loop count of Limits with index as loop count to limit.lim\_high of entry with index as loop count of exc\_mon\_data with index as loop count of Cmdownload\_lp\_cmd\_data.

13) Set limit.lim\_type of entry with index as loop count of Limits with index as loop count to limit.lim\_type of entry with index as loop count of exc\_mon\_data with index as loop count of Cmdownload\_lp\_cmd\_data.

14) Set limit.lim\_val of entry with index as loop count of Limits with index as loop count to limit.lim\_val of entry with index as loop count of exc\_mon\_data with index as loop count of Cmdownload\_lp\_cmd\_data.

15) Set limit.low of entry with index as loop count of Limits with index as loop count to limit.low of entry with index as loop count of exc\_mon\_data with index as loop count of Cmdownload\_lp\_cmd\_data.

16) Set limit.maxtime of entry with index as loop count of Limits with index as loop count to limit.maxtime of entry with index as loop count of exc\_mon\_data with index as loop count of Cmdownload\_lp\_cmd\_data.

17) Set limit.ptimer of entry with index as loop count of Limits with index as loop count to limit.ptimer of entry with index as loop count of exc\_mon\_data with index as loop count of Cmdownload\_lp\_cmd\_data.

18) Set limit.spcond of entry with index as loop count of Limits with index as loop count to limit.spcond of entry with index as loop count of exc\_mon\_data with index as loop count of Cmdownload\_lp\_cmd\_data.

19) Set limit.var of entry with index as loop count of Limits with index as loop count to limit.var of entry with index as loop count of exc\_mon\_data with index as loop count of Cmdownload\_lp\_cmd\_data.

20) Set limit.xtimer of entry with index as loop count of Limits with index as loop count to limit.xtimer of entry with index as loop count of exc\_mon\_data with index as loop count of Cmdownload\_lp\_cmd\_data.

21) Set peak of entry with index as loop count of Limits with index as loop count to peak of entry with index as loop count of exc\_mon\_data with index as loop count of Cmdownload\_lp\_cmd\_data.

22) Set prev\_exceed of entry with index as loop count of Limits with index as loop count to prev\_exceed of entry with index as loop count of exc\_mon\_data with index as loop count of Cmdownload\_lp\_cmd\_data.

23) Set start\_t of entry with index as loop count of Limits with index as loop count to start\_t of entry with index as loop count of exc\_mon\_data with index as loop count of Cmdownload\_lp\_cmd\_data.

24) Set lim\_others of entry with index as loop count of Limits with index as loop count to M\_NULL.

## 10.17 daugwyhw

The daugwyhw CSC provides definition of some common routines and processor startup routine.

### 10.17.1 InitIO

Low Level Design Details about CSU InitIO will follow in the sub sections.

#### 10.17.1.1 Brief Description

The InitIO function initializes GPIO pins.

#### 10.17.1.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.17.1.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.17.1.4 Parameter list (Input/Output)

Inputs: None

Outputs: None

#### 10.17.1.5 Return Value

None

#### 10.17.1.6 Other CSUs called by this CSU

RccAhb3PeriphClockCmd

RccApb2PeriphClockCmd

RccAhb1PeriphClockCmd

GpioInit

GpioPinAFConfig

#### 10.17.1.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to InitIO

##### 10.17.1.7.1 daugwyhw-InitIO-LLR-001

Requirement ID: H398-LLD-GWY-FNC-1600

The function RccAhb3PeriphClockCmd shall be called to Enable FSMC Clock with parameters M\_RCC\_AHB3PERIPH\_FSMC and ENABLE

##### 10.17.1.7.2 daugwyhw-InitIO-LLR-002

Requirement ID: H398-LLD-GWY-FNC-1601

The function shall do the following:

-Call RccApb2PeriphClockCmd to Enable system configuration Clock with parameters M\_RCC\_APB2PERIPH\_SYSCFG and ENABLE.

-Configure 50 MHZ Clock for all ports by setting gpio\_speed of gpio init structure to GPIO\_SPEED\_50MHZ.

##### 10.17.1.7.3 daugwyhw-InitIO-LLR-003

Requirement ID: H398-LLD-GWY-FNC-1602

The function RccAhb1PeriphClockCmd shall be called to Enable GPIOA Clock with parameters M\_RCC\_AHB1PERIPH\_GPIOA and ENABLE

##### 10.17.1.7.4 daugwyhw-InitIO-LLR-004

Requirement ID: H398-LLD-GWY-FNC-1603

The function shall do the following

Members of gpio init structure is set as follows.

-gpio\_pin to M\_GPIOA\_TONE\_OUT

-gpio\_mode to GPIO\_MODE\_AN

-gpio\_pupd to GPIO\_PUPD\_NOPULL

-Call GpioInit with M\_GPIOA and reference to gpio init structure

##### 10.17.1.7.5 daugwyhw-InitIO-LLR-005

Requirement ID: H398-LLD-GWY-FNC-1604

The function RccAhb1PeriphClockCmd shall be called to Enable GPIOB Clock with parameters M\_RCC\_AHB1PERIPH\_GPIOB and ENABLE

##### 10.17.1.7.6 daugwyhw-InitIO-LLR-006

Requirement ID: H398-LLD-GWY-FNC-1605

The function shall do the following

Members of gpio init structure is set as follows.

-gpio\_pin to M\_GPIOB\_LED\_HB

-gpio\_mode to GPIO\_MODE\_OUT

- gpio\_otype to GPIO\_OTYPE\_PP

- gpio\_pupd to GPIO\_PUPD\_NOPULL

-Call GpioInit with M\_GPIOB and reference to gpio init structure

##### 10.17.1.7.7 daugwyhw-InitIO-LLR-007

Requirement ID: H398-LLD-GWY-FNC-1606

The function RccAhb1PeriphClockCmd shall be called to Enable GPIOC Clock with parameters M\_RCC\_AHB1PERIPH\_GPIOC and ENABLE

##### 10.17.1.7.8 daugwyhw-InitIO-LLR-008

Requirement ID: H398-LLD-GWY-FNC-1607

The function shall do the following

Members of gpio init structure is set as follows.

-gpio\_pin to M\_GPIOC\_ADC\_IN1 Bitwise OR with M\_GPIOC\_ADC\_IN2

-gpio\_mode to GPIO\_MODE\_AN

-gpio\_pupd to GPIO\_PUPD\_NOPULL

-Call GpioInit with M\_GPIOC and reference to gpio init structure

##### 10.17.1.7.9 daugwyhw-InitIO-LLR-009

Requirement ID: H398-LLD-GWY-FNC-1608

The function shall do the following.

Members of gpio init structure is set as follows.

-gpio\_pin to M\_GPIOC\_RS232\_MODE Bitwise OR M\_GPIOC\_CHARGE\_CNTRL Bitwise OR M\_GPIOC\_GREEN\_LED Bitwise OR M\_GPIOC\_RED\_LED

-gpio\_mode to GPIO\_MODE\_OUT

-gpio\_otype to GPIO\_OTYPE\_PP

-gpio\_pupd to GPIO\_PUPD\_NOPULL

-Call GpioInit with M\_GPIOC and reference to gpio init structure

##### 10.17.1.7.10 daugwyhw-InitIO-LLR-010

Requirement ID: H398-LLD-GWY-FNC-1609

The function shall do the following

Members of gpio init structure is set as follows.

-gpio\_pin to M\_GPIOC\_SYS\_SEL Bitwise OR with M\_GPIOC\_SYS\_MODE

-gpio\_mode to GPIO\_MODE\_IN

-gpio\_pupd to GPIO\_PUPD\_UP

-Call GpioInit with M\_GPIOC and reference to gpio init structure

##### 10.17.1.7.11 daugwyhw-InitIO-LLR-011

Requirement ID: H398-LLD-GWY-FNC-1610

The function RccAhb1PeriphClockCmd shall be called to Enable GPIOD clock with parameters M\_RCC\_AHB1PERIPH\_GPIOD and ENABLE

##### 10.17.1.7.12 daugwyhw-InitIO-LLR-012

Requirement ID: H398-LLD-GWY-FNC-1611

The function GpioPinAFConfig shall be called each time to Configure FSMC pins of port d with 1st and 3rd parameters as M\_GPIOD and M\_GPIO\_AF\_FSMC and 2nd parameter as follows for each call of the function.

- M\_GPIO\_PINSOURCE0

- M\_GPIO\_PINSOURCE1

- M\_GPIO\_PINSOURCE4

- M\_GPIO\_PINSOURCE5

- M\_GPIO\_PINSOURCE7

- M\_GPIO\_PINSOURCE8

- M\_GPIO\_PINSOURCE9

- M\_GPIO\_PINSOURCE10

- M\_GPIO\_PINSOURCE11

- M\_GPIO\_PINSOURCE12

- M\_GPIO\_PINSOURCE13

- M\_GPIO\_PINSOURCE14

- M\_GPIO\_PINSOURCE15

##### 10.17.1.7.13 daugwyhw-InitIO-LLR-013

Requirement ID: H398-LLD-GWY-FNC-1612

The function shall do the following.

Members of gpio init structure is set as follows.

-gpio\_pin to M\_GPIOD\_D2 Bitwise OR M\_GPIOD\_D3 Bitwise OR M\_GPIOD\_OE Bitwise OR M\_GPIOD\_WE Bitwise OR M\_GPIOD\_NE1 Bitwise OR M\_GPIOD\_D13 Bitwise OR M\_GPIOD\_D14 Bitwise OR M\_GPIOD\_D15 Bitwise OR M\_GPIOD\_A16 Bitwise OR M\_GPIOD\_A17 Bitwise OR M\_GPIOD\_A18 Bitwise OR M\_GPIOD\_D0 Bitwise OR M\_GPIOD\_D1

-gpio\_mode to GPIO\_MODE\_AF

-gpio\_otype to GPIO\_OTYPE\_PP

-gpio\_pupd to GPIO\_PUPD\_NOPULL

-Call GpioInit with M\_GPIOD and reference to gpio init structure

##### 10.17.1.7.14 daugwyhw-InitIO-LLR-014

Requirement ID: H398-LLD-GWY-FNC-1613

The function RccAhb1PeriphClockCmd shall be called to Enable GPIOE clock with parameters M\_RCC\_AHB1PERIPH\_GPIOE and ENABLE

##### 10.17.1.7.15 daugwyhw-InitIO-LLR-015

Requirement ID: H398-LLD-GWY-FNC-1614

The function GpioPinAFConfig shall be called each time to Configure FSMC pins of port E with 1st and 3rd parameters as M\_GPIOE and M\_GPIO\_AF\_FSMC and 2nd parameter as follows for each call of the function.

- M\_GPIO\_PINSOURCE0

- M\_GPIO\_PINSOURCE1

- M\_GPIO\_PINSOURCE7

- M\_GPIO\_PINSOURCE8

- M\_GPIO\_PINSOURCE9

- M\_GPIO\_PINSOURCE10

- M\_GPIO\_PINSOURCE11

- M\_GPIO\_PINSOURCE12

- M\_GPIO\_PINSOURCE13

- M\_GPIO\_PINSOURCE14

- M\_GPIO\_PINSOURCE15

##### 10.17.1.7.16 daugwyhw-InitIO-LLR-016

Requirement ID: H398-LLD-GWY-FNC-1615

The function shall do the following

Members of gpio init structure is set as follows.

-gpio\_pin to M\_GPIOE\_BLE Bitwise OR M\_GPIOE\_BHE Bitwise OR M\_GPIOE\_D4 Bitwise OR M\_GPIOE\_D5 Bitwise OR M\_GPIOE\_D6 Bitwise OR M\_GPIOE\_D7 Bitwise OR M\_GPIOE\_D8 Bitwise OR

M\_GPIOE\_D9 Bitwise OR M\_GPIOE\_D10 Bitwise OR M\_GPIOE\_D11 Bitwise OR M\_GPIOE\_D12

-gpio\_mode to GPIO\_MODE\_AF

- gpio\_otype to GPIO\_OTYPE\_PP

-gpio\_pupd to GPIO\_PUPD\_NOPULL

-Call GpioInit with M\_GPIOE and reference to gpio init structure

##### 10.17.1.7.17 daugwyhw-InitIO-LLR-017

Requirement ID: H398-LLD-GWY-FNC-1616

The function RccAhb1PeriphClockCmd shall be called to Enable GPIOF clock with parameters M\_RCC\_AHB1PERIPH\_GPIOF and ENABLE

##### 10.17.1.7.18 daugwyhw-InitIO-LLR-018

Requirement ID: H398-LLD-GWY-FNC-1617

The function GpioPinAFConfig shall be called each time to Configure FSMC pins of port F with 1st and 3rd parameters as M\_GPIOF and M\_GPIO\_AF\_FSMC and 2nd parameter as follows for each call of the function.

- M\_GPIO\_PINSOURCE0

- M\_GPIO\_PINSOURCE1

- M\_GPIO\_PINSOURCE2

- M\_GPIO\_PINSOURCE3

- M\_GPIO\_PINSOURCE4

- M\_GPIO\_PINSOURCE5

- M\_GPIO\_PINSOURCE12

- M\_GPIO\_PINSOURCE13

- M\_GPIO\_PINSOURCE14

- M\_GPIO\_PINSOURCE15

##### 10.17.1.7.19 daugwyhw-InitIO-LLR-019

Requirement ID: H398-LLD-GWY-FNC-1618

The function shall do the following

Members of gpio init structure is set as follows.

-gpio\_pin to M\_GPIOF\_A0 Bitwise OR M\_GPIOF\_A1 Bitwise OR M\_GPIOF\_A2 Bitwise OR M\_GPIOF\_A3 Bitwise OR M\_GPIOF\_A4 Bitwise OR M\_GPIOF\_A5 Bitwise OR M\_GPIOF\_A6 Bitwise OR M\_GPIOF\_A7 Bitwise OR M\_GPIOF\_A8 Bitwise OR M\_GPIOF\_A9

-gpio\_mode to GPIO\_MODE\_AF

- gpio\_otype to GPIO\_OTYPE\_PP

-gpio\_pupd to GPIO\_PUPD\_NOPULL

-Call GpioInit with M\_GPIOF and reference to gpio init structure

##### 10.17.1.7.20 daugwyhw-InitIO-LLR-020

Requirement ID: H398-LLD-GWY-FNC-1619

The function RccAhb1PeriphClockCmd shall be called to Enable M\_GPIOG Clock with parameters M\_RCC\_AHB1PERIPH\_GPIOG and ENABLE

##### 10.17.1.7.21 daugwyhw-InitIO-LLR-021

Requirement ID: H398-LLD-GWY-FNC-1620

The function GpioPinAFConfig shall be called each time to Configure FSMC pins of port G with 1st and 3rd parameters as M\_GPIOG and M\_GPIO\_AF\_FSMC and 2nd parameter as follows for each call of the function.

- M\_GPIO\_PINSOURCE0

- M\_GPIO\_PINSOURCE1

- M\_GPIO\_PINSOURCE2

- M\_GPIO\_PINSOURCE3

- M\_GPIO\_PINSOURCE4

- M\_GPIO\_PINSOURCE5

- M\_GPIO\_PINSOURCE9

##### 10.17.1.7.22 daugwyhw-InitIO-LLR-022

Requirement ID: H398-LLD-GWY-FNC-1621

The function shall do the following,

Members of gpio init structure is set as follows.

-gpio\_pin to M\_GPIOG\_A10 Bitwise OR M\_GPIOG\_A11 Bitwise OR M\_GPIOG\_A12 Bitwise OR

M\_GPIOG\_A13 Bitwise OR M\_GPIOG\_A14 Bitwise OR M\_GPIOG\_A15 Bitwise OR M\_GPIOG\_NE2

-gpio\_mode to GPIO\_MODE\_AF

-gpio\_otype to GPIO\_OTYPE\_PP

-gpio\_pupd to GPIO\_PUPD\_NOPULL

-Call GpioInit with M\_GPIOG and reference to gpio init structure

### 10.17.2 HwInit

Low Level Design Details about CSU HwInit will follow in the sub sections.

#### 10.17.2.1 Brief Description

The HwInit function initialize hardware related components.

#### 10.17.2.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.17.2.3 List of global variables accessed and modified

Accessed :None

Modified : None

#### 10.17.2.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.17.2.5 Return Value

None

#### 10.17.2.6 Other CSUs called by this CSU

RccDeInit

RccHseConfig

RterrError

RterrForever

RccApb1PeriphClockCmd

PwrMainRegulatorModeConfig

RccHclkConfig

RccPclk1Config

RccPclk2Config

FlashSetLatency

FlashInstructionCacheCmd

FlashDataCacheCmd

FlashPrefetchBufferCmd

RccPllConfig

RccPllCmd

RccGetFlagStatus

RccSysClkConfig

RccGetSysClkSource

IntrInit

NvicSetVectorTable

NvicPriorityGroupConfig

IntrInstall

NvicSetPriority

InitIO

RccWaitForHseStartUp

RccAhb1PeriphClockCmd

PendSvHandler

TbaseIntrHandler

#### 10.17.2.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to HwInit

##### 10.17.2.7.1 daugwyhw-HwInit-LLR-001

Requirement ID: H398-LLD-GWY-FNC-1630

The function shall start with the clocks in their expected state by calling function RccDeInit.

##### 10.17.2.7.2 daugwyhw-HwInit-LLR-002

Requirement ID: H398-LLD-GWY-FNC-1631

The function shall enable HSE (High Speed External Clock) by calling function RccHseConfig with parameter M\_RCC\_HSE\_ON

##### 10.17.2.7.3 daugwyhw-HwInit-LLR-003

Requirement ID: H398-LLD-GWY-FNC-1632

The function shall turn on error code by calling function RterrError with parameters BIT\_ERR,one and RterrForever to check if wait for HSE is ready when return value of function call RccWaitForHseStartUp is equal to ERROR, otherwise do nothing.

##### 10.17.2.7.4 daugwyhw-HwInit-LLR-004

Requirement ID: H398-LLD-GWY-FNC-1633

The function shall Enable power interface clock by calling function RccApb1PeriphClockCmd with parameters M\_RCC\_APB1PERIPH\_PWR and ENABLE

##### 10.17.2.7.5 daugwyhw-HwInit-LLR-005

Requirement ID: H398-LLD-GWY-FNC-1634

The function shall Configure power regulator voltage scale 1 by calling function PwrMainRegulatorModeConfig with parameter M\_PWR\_REGULATOR\_VOLTAGE\_SCALE1

##### 10.17.2.7.6 daugwyhw-HwInit-LLR-006

Requirement ID: H398-LLD-GWY-FNC-1635

The function shall configure hclk as sysclk by calling function RccHclkConfig with parameter M\_RCC\_SYSCLK\_DIV1

##### 10.17.2.7.7 daugwyhw-HwInit-LLR-007

Requirement ID: H398-LLD-GWY-FNC-1636

The function shall configure pclk1 as hclk divide by four by calling function RccPclk1Config with parameter M\_RCC\_HCLK\_DIV4

##### 10.17.2.7.8 daugwyhw-HwInit-LLR-008

Requirement ID: H398-LLD-GWY-FNC-1637

The function shall configure pclk2 as hclk divide by two by calling function RccPclk2Config with parameter M\_RCC\_HCLK\_DIV2

##### 10.17.2.7.9 daugwyhw-HwInit-LLR-009

Requirement ID: H398-LLD-GWY-FNC-1638

The function shall set wait states required on the flash by calling function FlashSetLatency with parameter M\_FLASH\_LATENCY\_5

##### 10.17.2.7.10 daugwyhw-HwInit-LLR-010

Requirement ID: H398-LLD-GWY-FNC-1639

The function shall Enable instruction cache by calling function FlashInstructionCacheCmd with parameter ENABLE

##### 10.17.2.7.11 daugwyhw-HwInit-LLR-011

Requirement ID: H398-LLD-GWY-FNC-1640

The function shall Enable data cache by calling function FlashDataCacheCmd with parameter ENABLE

##### 10.17.2.7.12 daugwyhw-HwInit-LLR-012

Requirement ID: H398-LLD-GWY-FNC-1641

The function shall Enable flash prefetch buffer by calling function FlashPrefetchBufferCmd with parameter ENABLE

##### 10.17.2.7.13 daugwyhw-HwInit-LLR-013

Requirement ID: H398-LLD-GWY-FNC-1642

The function shall configure PLL CLK by calling function RccPllConfig with parameters M\_RCC\_PLLSOURCE\_HSE, M\_HW\_PLL\_M, M\_HW\_PLL\_N, M\_HW\_PLL\_P and M\_HW\_PLL\_Q

##### 10.17.2.7.14 daugwyhw-HwInit-LLR-014

Requirement ID: H398-LLD-GWY-FNC-1643

The function shall Enable PLL by calling function RccPllCmd with parameter ENABLE

##### 10.17.2.7.15 daugwyhw-HwInit-LLR-015

Requirement ID: H398-LLD-GWY-FNC-1644

The function shall wait till PLL is ready by calling function RccGetFlagStatus with parameter M\_RCC\_FLAG\_PLLRDY and iterate till return value of function RccGetFlagStatus is equal to RESET.

##### 10.17.2.7.16 daugwyhw-HwInit-LLR-016

Requirement ID: H398-LLD-GWY-FNC-1645

The function shall select PLL as System Clock source by calling function RccSysClkConfig with parameter M\_RCC\_SYSCLKSOURCE\_PLLCLK

##### 10.17.2.7.17 daugwyhw-HwInit-LLR-017

Requirement ID: H398-LLD-GWY-FNC-1646

The function shall Wait till PLL is used as System Clock source by calling function RccGetSysClkSource and iterate till return value of function RccGetSysClkSource is not equal to M\_PLL\_USD\_AS\_SYSCLK.

##### 10.17.2.7.18 daugwyhw-HwInit-LLR-018

Requirement ID: H398-LLD-GWY-FNC-1647

The function shall Initialize the interrupt vector table in RAM by calling function IntrInit.

##### 10.17.2.7.19 daugwyhw-HwInit-LLR-019

Requirement ID: H398-LLD-GWY-FNC-1648

The function shall Set the Vector Table base address at RAM by calling function NvicSetVectorTable with parameters M\_NVIC\_VECTTAB\_RAM and M\_VT\_BSE\_ADDR\_OFFSET

##### 10.17.2.7.20 daugwyhw-HwInit-LLR-020

Requirement ID: H398-LLD-GWY-FNC-1649

The function shall Configure the group priority equal to sixteen and sub-priority is zero by calling function NvicPriorityGroupConfig with parameter M\_NVIC\_PRIORITYGROUP\_4

##### 10.17.2.7.21 daugwyhw-HwInit-LLR-021

Requirement ID: H398-LLD-GWY-FNC-1650

The function shall Setting up the system tick interrupt into vector table in RAM by calling function IntrInstall with parameters INTR\_SYS\_TICK and TbaseIntrHandler

##### 10.17.2.7.22 daugwyhw-HwInit-LLR-022

Requirement ID: H398-LLD-GWY-FNC-1651

The function shall set the priority of the system tick interrupt by calling function NvicSetPriority with parameters SYSTICK\_IRQN and M\_SYS\_TIC\_INTR\_PRIO

##### 10.17.2.7.23 daugwyhw-HwInit-LLR-023

Requirement ID: H398-LLD-GWY-FNC-1652

The function shall set up the PendSV interrupt into vector table in RAM by calling function IntrInstall with parameters INTR\_PEND\_SV and PendSvHandler

##### 10.17.2.7.24 daugwyhw-HwInit-LLR-024

Requirement ID: H398-LLD-GWY-FNC-1653

The function shall set the priority of the PendSV interrupt by calling function NvicSetPriority with parameters PENDSV\_IRQN and M\_PENDSV\_INTR\_PRIO

##### 10.17.2.7.25 daugwyhw-HwInit-LLR-025

Requirement ID: H398-LLD-GWY-FNC-1654

The function shall enable CRC Peripheral Clock by calling function RccAhb1PeriphClockCmd with parameters M\_RCC\_AHB1PERIPH\_CRC and ENABLE

##### 10.17.2.7.26 daugwyhw-HwInit-LLR-026

Requirement ID: H398-LLD-GWY-FNC-1655

The function shall Initializing the FPU context save not in lazy mode by setting fpccr of M\_FPU to M\_FPU\_CNTXT\_NTLAZY\_MODE

##### 10.17.2.7.27 daugwyhw-HwInit-LLR-027

Requirement ID: H398-LLD-GWY-FNC-1656

The function shall enable floating point coprocessor by setting cpacr of M\_SCB to cpacr of M\_SCB Bitwise OR with M\_FP\_COPROC\_EN

##### 10.17.2.7.28 daugwyhw-HwInit-LLR-029

Requirement ID: H398-LLD-GWY-FNC-6404

The function shall set shcsr of M\_SCB to shcsr of M\_SCB bitwise OR with M\_SCB\_SHCSR\_USGFAULTENA\_MSK bitwise OR with M\_SCB\_SHCSR\_BUSFAULTENA\_MSK and bitwise OR with M\_SCB\_SHCSR\_MEMFAULTENA\_MSK.

##### 10.17.2.7.29 daugwyhw-HwInit-LLR-030

Requirement ID: H398-LLD-GWY-FNC-6405

The function shall set ccr of M\_SCB to ccr of M\_SCB bitwise OR with M\_DIVIDE\_ZERO.

##### 10.17.2.7.30 daugwyhw-HwInit-LLR-028

Requirement ID: H398-LLD-GWY-FNC-1657

The function shall initialize IO pins by calling function InitIO

### 10.17.3 HwCopy

Low Level Design Details about CSU HwCopy will follow in the sub sections.

#### 10.17.3.1 Brief Description

The HwCopy function copies data from source buffer to destination buffer.

#### 10.17.3.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.17.3.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.17.3.4 Parameter list (Input/Output)

Inputs : T\_UINT8 \*p\_dest - pointer to destination buffer,

T\_UINT8 \*p\_src - pointer to source buffer,

T\_UINT32 count - number of bytes to be copied.

Outputs : T\_UINT8 \*p\_dest - pointer to destination buffer

#### 10.17.3.5 Return Value

None

#### 10.17.3.6 Other CSUs called by this CSU

None

#### 10.17.3.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to HwCopy

##### 10.17.3.7.1 daugwyhw-HwCopy-LLR-001

Requirement ID: H398-LLD-GWY-DRQ-1666

The function shall copy the data from p\_src buffer into p\_dest buffer for number of bytes specified by count by performing the following

-Loop till post decrement of count is not zero and copy the data from p\_src buffer into p\_dest buffer.

### 10.17.4 HwMemset

Low Level Design Details about CSU HwMemset will follow in the sub sections.

#### 10.17.4.1 Brief Description

The HwMemset function copies value to destination buffer.

#### 10.17.4.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.17.4.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.17.4.4 Parameter list (Input/Output)

Inputs : T\_UINT8 value - value to be copied into the destination

buffer

T\_UINT16 count - number of bytes to be copied

Outputs : T\_UINT8 \* p\_dest - pointer to destination buffer

#### 10.17.4.5 Return Value

None

#### 10.17.4.6 Other CSUs called by this CSU

None

#### 10.17.4.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to HwMemset

##### 10.17.4.7.1 daugwyhw-HwMemset-LLR-001

Requirement ID: H398-LLD-GWY-DRQ-1675

The function shall loop from range M\_ZERO to count minus 1 and copy the value into p\_dest buffer for every index of loop.

### 10.17.5 HwAbs32

Low Level Design Details about CSU HwAbs32 will follow in the sub sections.

#### 10.17.5.1 Brief Description

The HwAbs32 function provides absolute value of given 32 bit input.

#### 10.17.5.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.17.5.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.17.5.4 Parameter list (Input/Output)

Inputs : T\_SINT32 input -given input

Outputs : None

#### 10.17.5.5 Return Value

T\_SINT32 - absolute value of input

#### 10.17.5.6 Other CSUs called by this CSU

None

#### 10.17.5.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to HwAbs32

##### 10.17.5.7.1 daugwyhw-HwAbs32-LLR-001

Requirement ID: H398-LLD-GWY-DRQ-1684

The function shall do the following

* Set input with subtraction of M\_DEC\_ZERO and input when input is less than M\_DEC\_ZERO, otherwise do nothing
* Return input

## 10.18 daugwyiadc

The daugwyiadc CSC defines implementation of routines and APIs for internal ADC.

### 10.18.1 IadcRead

Low Level Design Details about CSU IadcRead will follow in the sub sections.

#### 10.18.1.1 Brief Description

The IadcRead function returns the average reading of the corresponding channel.

#### 10.18.1.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.18.1.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.18.1.4 Parameter list (Input/Output)

Inputs: T\_UINT8 u8\_chan -ADC channel number.

Outputs: None

#### 10.18.1.5 Return Value

T\_SINT16 - Returns the average reading of the corresponding channel.

#### 10.18.1.6 Other CSUs called by this CSU

None

#### 10.18.1.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to IadcRead

##### 10.18.1.7.1 daugwyiadc-IadcRead-LLR-001

Requirement ID: H398-LLD-GWY-FNC-1694

The function shall return bit shift of i32\_average of iadc data of index u8\_chan to right by M\_IADC\_POWER.

### 10.18.2 TransferCompleteIntr

Low Level Design Details about CSU TransferCompleteIntr will follow in the sub sections.

#### 10.18.2.1 Brief Description

The TransferCompleteIntr function applies averaging to reading from DMA.

#### 10.18.2.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.18.2.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.18.2.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.18.2.5 Return Value

None

#### 10.18.2.6 Other CSUs called by this CSU

DmaClearItPendingBit

#### 10.18.2.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TransferCompleteIntr

##### 10.18.2.7.1 daugwyiadc-TransferCompleteIntr-LLR-001

Requirement ID: H398-LLD-GWY-FNC-1703

The function shall Clear DMA2 stream 0 transfer complete interrupt bit by calling function DmaClearItPendingBit with parameters M\_DMA2\_STREAM0 and M\_DMA\_IT\_TCIF0.

##### 10.18.2.7.2 daugwyiadc-TransferCompleteIntr-LLR-002

Requirement ID: H398-LLD-GWY-FNC-1704

The function shall do the following:

-Loop from M\_ZERO to IADC\_CHANS minus 1

-Set i32\_average of iadc data for every loop index with i32\_average of iadc data added to (subtraction of Ai16 readings and (i32\_average of iadc data of loop index right shifted by M\_IADC\_POWER))

### 10.18.3 ConfigureDma

Low Level Design Details about CSU ConfigureDma will follow in the sub sections.

#### 10.18.3.1 Brief Description

The ConfigureDma function Initialize DMA for ADC 1.

#### 10.18.3.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.18.3.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.18.3.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.18.3.5 Return Value

None

#### 10.18.3.6 Other CSUs called by this CSU

RccAhb1PeriphClockCmd

IntrInstall

NvicInit

DmaItConfig

DmaInit

DmaCmd

TransferCompleteIntr

#### 10.18.3.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to ConfigureDma

##### 10.18.3.7.1 daugwyiadc-ConfigureDma-LLR-001

Requirement ID: H398-LLD-GWY-FNC-1713

The function shall Enable DMA2 clock by calling function RccAhb1PeriphClockCmd with parameters M\_RCC\_AHB1PERIPH\_DMA2 and ENABLE.

##### 10.18.3.7.2 daugwyiadc-ConfigureDma-LLR-002

Requirement ID: H398-LLD-GWY-FNC-1714

The function shall Install DMA2 stream 0 interrupt into vector table in RAM by calling function IntrInstall with parameters INTR\_DMA\_2\_STREAM\_0 and TransferCompleteIntr.

##### 10.18.3.7.3 daugwyiadc-ConfigureDma-LLR-003

Requirement ID: H398-LLD-GWY-FNC-1715

The function shall do the follwoing.

- nvic\_irq\_channel to DMA2\_STREAM0\_IRQN

- nvic\_irq\_channel\_preempt\_pri to M\_NVIC\_IRQ\_CHANNEL\_PREEMPT\_PRI8

- nvic\_irq\_channel\_subpriority to M\_NVIC\_IRQ\_CHANNEL\_SUBPRIORITY0

- nvic\_irq\_channel\_cmd to ENABLE

-Call ' NvicInit' with parameter as reference of nvic init structure.

##### 10.18.3.7.4 daugwyiadc-ConfigureDma-LLR-004

Requirement ID: H398-LLD-GWY-FNC-1716

The function shall Enable DMA2 stream 0 transfer complete interrupt by calling function DmaItConfig with parameters M\_DMA2\_STREAM0, M\_DMA\_IT\_TC and ENABLE

##### 10.18.3.7.5 daugwyiadc-ConfigureDma-LLR-005

Requirement ID: H398-LLD-GWY-FNC-1717

The function DmaInit shall be called with parameters M\_DMA2\_STREAM0 and reference of dma init structure to Initialize DMA2 channel 0 for ADC.

Members of dma init structure is set as follows.

- dma\_channel to M\_DMA\_CHANNEL\_0

-dma\_memory0\_baseaddr to Ai16 readings

-dma\_peripheral\_baseaddr to reference of dr of M\_ADC1

-dma\_dir to M\_DMA\_DIR\_PERIPHERAL\_TO\_MEMORY

-dma\_buffersize to IADC\_CHANS

-dma\_peripheral\_inc to M\_DMA\_PERIPHERALINC\_DISABLE

-dma\_memory\_inc to M\_DMA\_MEMORYINC\_ENABLE

-dma\_peripheral\_datasize to M\_DMA\_PERIPH\_DATASIZE\_HALFWORD

-dma\_memory\_datasize to M\_DMA\_MEMORY\_DATASIZE\_HALFWORD

-dma\_mode to M\_DMA\_MODE\_CIRCULAR

-dma\_priority to M\_DMA\_PRIORITY\_MEDIUM

-dma\_fifo\_mode to M\_DMA\_FIFOMODE\_DISABLE

-dma\_fifo\_threshold to M\_DMA\_FIFOTHRESHOLD\_FULL

-dma\_memory\_burst to M\_DMA\_MEMORYBURST\_SINGLE

-dma\_peripheral\_burst to M\_DMA\_PERIPHERALBURST\_SINGLE

##### 10.18.3.7.6 daugwyiadc-ConfigureDma-LLR-006

Requirement ID: H398-LLD-GWY-FNC-1718

The function shall Enable DMA2\_Stream 0 by calling function DmaCmd with parameters M\_DMA2\_STREAM0 and ENABLE.

### 10.18.4 TriggerInternalAdc

Low Level Design Details about CSU TriggerInternalAdc will follow in the sub sections.

#### 10.18.4.1 Brief Description

The TriggerInternalAdc function triggers the ADC 1 for software conversion.

#### 10.18.4.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.18.4.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.18.4.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.18.4.5 Return Value

None

#### 10.18.4.6 Other CSUs called by this CSU

TimClearITPendingBit

AdcSoftwareStartConv

#### 10.18.4.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TriggerInternalAdc

##### 10.18.4.7.1 daugwyiadc-TriggerInternalAdc-LLR-001

Requirement ID: H398-LLD-GWY-FNC-1727

The function ‘TimClearITPendingBit’ shall be called to Clear TIM6 update interrupt pending bit with parameters M\_TIM6 and M\_TIM\_IT\_UPDATE.

##### 10.18.4.7.2 daugwyiadc-TriggerInternalAdc-LLR-002

Requirement ID: H398-LLD-GWY-FNC-1728

The function ‘AdcSoftwareStartConv’ shall be called to Start ADC1 Software Conversion with parameter M\_ADC1.

### 10.18.5 IadcInit

Low Level Design Details about CSU IadcInit will follow in the sub sections.

#### 10.18.5.1 Brief Description

The IadcInit function initializes ADC 1 and DMA 2 for internal ADC.

#### 10.18.5.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.18.5.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.18.5.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.18.5.5 Return Value

None

#### 10.18.5.6 Other CSUs called by this CSU

RccApb2PeriphClockCmd

AdcCommonInit

ConfigureDma

AdcInit

AdcRegularChannelConfig

AdcDmaReqAfterLastTransferCmd

AdcDmaCmd

AdcCmd

TmrInitTim6

TriggerInternalAdc

#### 10.18.5.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to IadcInit

##### 10.18.5.7.1 daugwyiadc-IadcInit-LLR-001

Requirement ID: H398-LLD-GWY-FNC-1737

The function 'RccApb2PeriphClockCmd' shall be called to Enable ADC1 clock with parameters M\_RCC\_APB2PERIPH\_ADC1 and ENABLE

##### 10.18.5.7.2 daugwyiadc-IadcInit-LLR-002

Requirement ID: H398-LLD-GWY-FNC-1738

The function shall set members of adc commoninit structure as follows.

-adc\_mode to M\_ADC\_MODE\_INDEPENDENT

-adc\_prescaler to M\_ADC\_PRESCALER\_DIV8

-adc\_dma\_access\_mode to M\_ADC\_DMAACCESSMODE\_DISABLED

-adc\_two\_sampling\_delay to M\_ADC\_TWOSAMPLINGDELAY\_5CYCLES

##### 10.18.5.7.3 daugwyiadc-IadcInit-LLR-003

Requirement ID: H398-LLD-GWY-FNC-1739

The function AdcCommonInit shall be called to Common initialization for two ADCs with parameter as address of adc commoninit structure

##### 10.18.5.7.4 daugwyiadc-IadcInit-LLR-004

Requirement ID: H398-LLD-GWY-FNC-1740

The function ConfigureDma shall be called to Configure DMA for internal ADC.

##### 10.18.5.7.5 daugwyiadc-IadcInit-LLR-005

Requirement ID: H398-LLD-GWY-FNC-1741

The function shall set members of adc init structure as follows.

- adc\_resolution to M\_ADC\_RESOLUTION\_12B

- adc\_scan\_convmode to ENABLE

##### 10.18.5.7.6 daugwyiadc-IadcInit-LLR-006

Requirement ID: H398-LLD-GWY-FNC-1742

The function shall set members of adc init structure as follows.

- adc\_continuous\_convmode to DISABLE

- adc\_external\_trig\_convedge to M\_ADC\_EXTN\_TRIG\_CONV\_EDGE\_NONE

- adc\_external\_trigconv to M\_ADC\_EXTERNALTRIGCONV\_T1\_CC1

- adc\_dataalign to M\_ADC\_DATAALIGN\_RIGHT

##### 10.18.5.7.7 daugwyiadc-IadcInit-LLR-007

Requirement ID: H398-LLD-GWY-FNC-1743

The function shall set members of adc init structure as follows.

- adc\_nbr\_of\_conversion to IADC\_CHANS

##### 10.18.5.7.8 daugwyiadc-IadcInit-LLR-008

Requirement ID: H398-LLD-GWY-FNC-1744

The function AdcInit shall be called to configure ADC1 with parameters M\_ADC1 and address of adc init structure.

##### 10.18.5.7.9 daugwyiadc-IadcInit-LLR-009

Requirement ID: H398-LLD-GWY-FNC-1745

The function AdcRegularChannelConfig shall be called to ADC1 regular channels 10 configuration with parameters M\_ADC1, M\_ADC\_CHANNEL\_10, M\_DMA\_REG\_GRPSEQ\_RANK1,

M\_ADC\_SAMPLETIME\_56CYCLES

##### 10.18.5.7.10 daugwyiadc-IadcInit-LLR-010

Requirement ID: H398-LLD-GWY-FNC-1746

The function AdcRegularChannelConfig shall be called to M\_ADC1 regular channels 11 configuration with parameters M\_ADC1, M\_ADC\_CHANNEL\_11, M\_DMA\_REG\_GRPSEQ\_RANK2,

M\_ADC\_SAMPLETIME\_56CYCLES

##### 10.18.5.7.11 daugwyiadc-IadcInit-LLR-011

Requirement ID: H398-LLD-GWY-FNC-1747

The function AdcDmaReqAfterLastTransferCmd shall be called to Enable DMA request after last transfer (Single-ADC mode) with parameters M\_ADC1 and ENABLE

##### 10.18.5.7.12 daugwyiadc-IadcInit-LLR-012

Requirement ID: H398-LLD-GWY-FNC-1748

The function AdcDmaCmd shall be called to Enable ADC1 DMA with parameters M\_ADC1 and ENABLE

##### 10.18.5.7.13 daugwyiadc-IadcInit-LLR-013

Requirement ID: H398-LLD-GWY-FNC-1749

The function AdcCmd shall be called to Enable ADC1 with parameters M\_ADC1 and ENABLE

##### 10.18.5.7.14 daugwyiadc-IadcInit-LLR-014

Requirement ID: H398-LLD-GWY-FNC-1750

The function TmrInitTim6 shall be called to Configure timer 6 to trigger internal ADC converting rate with parameter TriggerInternalAdc

## 10.19 daugwyinit

The daugwyinit CSC initializes all the tasks for the OS to handle.

### 10.19.1 InitTask

Low Level Design Details about CSU InitTask will follow in the sub sections.

#### 10.19.1.1 Brief Description

This function is the TASK that is never exit, used by the OS to create the thread/task for this module.

#### 10.19.1.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWYs).

#### 10.19.1.3 List of global variables accessed and modified

Accessed : Appdata\_appdata

Modified : None

#### 10.19.1.4 Parameter list (Input/Output)

Inputs : void \* p\_data Kill compiler warning

Outputs : void \* p\_data Kill compiler warning

#### 10.19.1.5 Return Value

None

#### 10.19.1.6 Other CSUs called by this CSU

GpioToggleBits

TbaseInit

WdogKickWatchDog

XramInit

DtoaHwInit

AppInit

IadcInit

A825Init

A825Init2

A825CommInit

A825CommInit2

A429Init

OsSemPend  
SoundInit

ETIinit

LOGICInit

SoundInit

#### 10.19.1.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to InitTask.

##### 10.19.1.7.1 daugwyinit-InitTask-LLR-001

Requirement ID: H398-LLD-GWY-FNC-1760

The function shall

a) Set p\_data to p\_data added with M\_ZERO.

b) Call WdogInit to initialize the Independent Watchdog.

c) Call TbaseInit to initialize time base of the system.

##### 10.19.1.7.2 daugwyinit-InitTask-LLR-002

Requirement ID: H398-LLD-GWY-FNC-1761

The function shall refresh the Watchdog at regular intervals after each initialization and in idle state by calling WdogKickWatchDog.

##### 10.19.1.7.3 daugwyinit-InitTask-LLR-003

Requirement ID: H398-LLD-GWY-FNC-1762

The function shall initialize the RS232 ports by calling Rs232Init. And refresh the Watchdog at regular intervals after each initialization and in idle state by calling WdogKickWatchDog.

##### 10.19.1.7.4 daugwyinit-InitTask-LLR-004

Requirement ID: H398-LLD-GWY-FNC-1763

The function shall configure the FSMC and enables the bank 1 of FSMC by calling XramInit. And refresh the Watchdog at regular intervals after each initialization and in idle state by calling WdogKickWatchDog.

##### 10.19.1.7.5 daugwyinit-InitTask-LLR-005

Requirement ID: H398-LLD-GWY-FNC-1764

The function shall initialize the Internal DAC by calling DtoaHwInit. And refresh the Watchdog at regular intervals after each initialization and in idle state by calling WdogKickWatchDog.

##### 10.19.1.7.6 daugwyinit-InitTask-LLR-006

Requirement ID: H398-LLD-GWY-FNC-7471

The function shall initialize the Sound related routines by calling SoundInit. And refresh the Watchdog at regular intervals after each initialization and in idle state by calling WdogKickWatchDog.

##### 10.19.1.7.7 daugwyinit-InitTask-LLR-007

Requirement ID: H398-LLD-GWY-FNC-1766

The function shall initialize the Application Task by calling AppInit. And refresh the Watchdog at regular intervals after each initialization and in idle state by calling WdogKickWatchDog

##### 10.19.1.7.8 daugwyinit-InitTask-LLR-008

Requirement ID: H398-LLD-GWY-FNC-1767

The function shall initialize the Internal ADC by calling IadcInit. And refresh the Watchdog at regular intervals after each initialization and in idle state by calling WdogKickWatchDog

##### 10.19.1.7.9 daugwyinit-InitTask-LLR-009

Requirement ID: H398-LLD-GWY-FNC-1768

The function shall initialize the Arinc 825 Bus by calling A825Init. And refresh the Watchdog at regular intervals after each initialization and in idle state by calling WdogKickWatchDog

##### 10.19.1.7.10 daugwyinit-InitTask-LLR-010

Requirement ID: H398-LLD-GWY-FNC-1769

The function shall initialize the Arinc 825 2 Bus by calling A825Init2. And refresh the Watchdog at regular intervals after each initialization and in idle state by calling WdogKickWatchDog

##### 10.19.1.7.11 daugwyinit-InitTask-LLR-011

Requirement ID: H398-LLD-GWY-FNC-1770

The function shall initialize the Arinc 825 communication Bus by calling A825CommInit. And refresh the Watchdog at regular intervals after each initialization and in idle state by calling WdogKickWatchDog

##### 10.19.1.7.12 daugwyinit-InitTask-LLR-012

Requirement ID: H398-LLD-GWY-FNC-1771

The function shall initialize the Arinc 825 communication 2 Bus by calling A825CommInit2. And refresh the Watchdog at regular intervals after each initialization and in idle state by calling WdogKickWatchDog

##### 10.19.1.7.13 daugwyinit-InitTask-LLR-013

Requirement ID: H398-LLD-GWY-FNC-1772

The function shall initialize the Arinc 429 Bus by calling A429Init and refresh the Watchdog at regular intervals after each initialization and in idle state by calling WdogKickWatchDog when func\_state of a429chan with index M\_ZERO of Appdata\_appdata is enabled.

##### 10.19.1.7.14 daugwyinit-InitTask-LLR-015

Requirement ID: H398-LLD-GWY-FNC-1774

The function shall initialize the semaphore into timebase by calling ETIinit. And refresh the Watchdog at regular intervals after each initialization and in idle state by calling WdogKickWatchDog

##### 10.19.1.7.15 daugwyinit-InitTask-LLR-016

Requirement ID: H398-LLD-GWY-FNC-1775

The function shall initialize the task signaling parameters, stack for run time checks and creates the ARINC 429 task by calling LogicInit and refresh the Watchdog at regular intervals after each initialization and in idle state by calling WdogKickWatchDog.

##### 10.19.1.7.16 daugwyinit-InitTask-LLR-017

Requirement ID: H398-LLD-GWY-FNC-1776

The function shall do the following

a) Call ContinuousBitTaskInit to initialize the Cbit task.

b)Refresh the Watchdog at regular intervals after each initialization and in idle state by calling WdogKickWatchDog

##### 10.19.1.7.17 daugwyinit-InitTask-LLR-018

Requirement ID: H398-LLD-GWY-FNC-1777

The function shall loop infinitely and perform the following operations within loop

a) Pend on Semaphore by calling OsSemPend with the following parameters

- Allocated Semaphore for the task sem init task

- pend timeout value as M\_ZERO

- Address of local error code.

b) Call WdogKickWatchDog to reload watchdog counter.

c) Call GpioToggleBits with reference to GPIO port B (M\_GPIOB) and GPIO pin 2 (M\_GPIO\_PIN\_2) (M\_GPIOB\_LED\_HB) as parameters to toggle the heartbeat LED.

### 10.19.2 InitInit

Low Level Design Details about CSU InitInit will follow in the sub sections.

#### 10.19.2.1 Brief Description

The InitInit function installs Semaphore into timebase and creates init OS task.

#### 10.19.2.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.19.2.3 List of global variables accessed and modified

Accessed : Init\_task\_stk

Modified : Init\_task\_stk

#### 10.19.2.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.19.2.5 Return Value

None

#### 10.19.2.6 Other CSUs called by this CSU

OsSemCreate

TbaseTaskSignaling

OsTaskCreate

RterrError

RterrForever

InitTask

#### 10.19.2.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to InitInit.

##### 10.19.2.7.1 daugwyinit-InitInit-LLR-001

Requirement ID: H398-LLD-GWY-FNC-1786

The function shall install task signaling parameters for the init task by calling 'TbaseTaskSignaling' with parameters

a) M\_INIT\_TASK\_TICKS as Task Ticks and

b) return value of function OsSemCreate with parameter M\_ZERO.

##### 10.19.2.7.2 daugwyinit-InitInit-LLR-002

Requirement ID: H398-LLD-GWY-FNC-1787

The function shall loop from stack counter is M\_ZERO to M\_INIT\_TASK\_STK\_SIZE minus 1 and initialize Init\_task\_stk of index stack counter with M\_CBIT\_TASK\_STK\_VAL.

##### 10.19.2.7.3 daugwyinit-InitInit-LLR-003

Requirement ID: H398-LLD-GWY-FNC-1788

The function shall create the Init Task by calling 'OsTaskCreate' with parameters (pointer to function InitTask,M\_HW\_NULL,base address of array Init\_task\_stk with index M\_INIT\_TASK\_STK\_SIZE,M\_INIT\_TASK\_PRIO) and call the function RterrError with parameter (BIT\_ERR, M\_THREE,RterrForever)  when the function OsTaskCreate returns other than M\_OS\_NO\_ERR, otherwise do nothing.

.

## 10.20 daugwyintr

The daugwyintr CSC initializes the vector table and allows installing and uninstalling interrupts into the table.

### 10.20.1 IntrInit

Low Level Design Details about CSU IntrInit will follow in the sub sections.

#### 10.20.1.1 Brief Description

The IntrInit function copies the ISR functions from ROM to RAM.

#### 10.20.1.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.20.1.3 List of global variables accessed and modified

Accessed : None

Modified : Lu32\_stack

#### 10.20.1.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.20.1.5 Return Value

None

#### 10.20.1.6 Other CSUs called by this CSU

SpuriousInterrupt

Lfntable\_ram

#### 10.20.1.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to IntrInit

##### 10.20.1.7.1 daugwyintr-IntrInit-LLR-001

Requirement ID: H398-LLD-GWY-FNC-1798

The function shall loop from M\_ZERO until M\_MAIN\_STK\_SIZE minus 1 and set Lu32\_stack of index loop counter with M\_CBIT\_TASK\_STK\_VAL.

##### 10.20.1.7.2 daugwyintr-IntrInit-LLR-002

Requirement ID: H398-LLD-GWY-FNC-1799

The function shall loop from INTR\_RESET until INTR\_MAX minus 1 to copy interrupt function from ROM to RAM as follows:

-set Interrupt Vector Table RAM with Interrupt Vector Table ROM for all the indices when Interrupt Vector Table ROM is not M\_ZERO (for all index).

- set Interrupt Vector Table RAM with SpuriousInterrupt for all the indices when Interrupt Vector Table ROM is M\_ZERO (for all index)

### 10.20.2 IntrInstall

Low Level Design Details about CSU IntrInstall will follow in the sub sections.

#### 10.20.2.1 Brief Description

The IntrInstall function loads the ISR into the vector table for the received interrupt index.

#### 10.20.2.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.20.2.3 List of global variables accessed and modified

Accessed : U32\_critical\_sr

Modified : U32\_critical\_sr

#### 10.20.2.4 Parameter list (Input/Output)

Inputs : T\_UINT8 u8\_index - interrupt number index in the vector table

T\_INTR\_FN intr\_function - ISR fucntion

Outputs : None

#### 10.20.2.5 Return Value

None

#### 10.20.2.6 Other CSUs called by this CSU

SaveStatusReg

RestoreStatusReg

#### 10.20.2.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to IntrInstall

##### 10.20.2.7.1 daugwyintr-IntrInstall-LLR-001

Requirement ID: H398-LLD-GWY-FNC-1808

The function shall call SaveStatusReg(M\_OS\_ENTER\_CRITICAL) and set return value of SaveStatusReg to U32\_critical\_sr in order to disable interrupts before loading the ISR into the Interrupt Vector

##### 10.20.2.7.2 daugwyintr-IntrInstall-LLR-002

Requirement ID: H398-LLD-GWY-FNC-1809

The function shall set Interrupt Vector Table RAM with intr\_function when u8\_index is less than INTR\_MAX, otherwise do nothing.

##### 10.20.2.7.3 daugwyintr-IntrInstall-LLR-003

Requirement ID: H398-LLD-GWY-FNC-1810

The function shall call RestoreStatusReg (M\_OS\_EXIT\_CRITICAL) with parameter U32\_critical\_sr to restore the priority mask register after the loading is complete.

### 10.20.3 IsrReset

Low Level Design Details about CSU IsrReset will follow in the sub sections.

#### 10.20.3.1 Brief Description

The IsrReset function is the reset routine on power-up.

#### 10.20.3.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.20.3.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.20.3.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.20.3.5 Return Value

None

#### 10.20.3.6 Other CSUs called by this CSU

Crt0TransferData

#### 10.20.3.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to IsrReset

##### 10.20.3.7.1 daugwyintr-IsrReset-LLR-001

Requirement ID: H398-LLD-GWY-FNC-1819

The function shall call Crt0TransferData to run the reset routine.

### 10.20.4 NonMaskable

Low Level Design Details about CSU NonMaskable will follow in the sub sections.

#### 10.20.4.1 Brief Description

The NonMaskable function is an interrupt error handler for non-Maskable errors.

#### 10.20.4.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.20.4.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.20.4.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.20.4.5 Return Value

None

#### 10.20.4.6 Other CSUs called by this CSU

RterrError

RterrForever

#### 10.20.4.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to NonMaskable

##### 10.20.4.7.1 daugwyintr-NonMaskable-LLR-001

Requirement ID: H398-LLD-GWY-FNC-1828

The function shall call RterrError with parameters (INTR\_ERR, M\_ONE and pointer to function RterrForever) to flash code accordingly for nonMaskable errors.

### 10.20.5 HardFault

Low Level Design Details about CSU HardFault will follow in the sub sections.

#### 10.20.5.1 Brief Description

The HardFault function is the Hard Fault Handler wrapper assembly. It extracts the location of stack frame and passes it to handler in C as a pointer.

(NOTE: HardFault is caused by Bus Fault, Memory Management Fault or Usage Fault if their handler cannot be executed.)

#### 10.20.5.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.20.5.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.20.5.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.20.5.5 Return Value

None

#### 10.20.5.6 Other CSUs called by this CSU

RterrError

RterrForever

#### 10.20.5.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to HardFault

##### 10.20.5.7.1 daugwyintr-HardFault-LLR-001

Requirement ID: H398-LLD-GWY-FNC-1837

The function shall call the function RterrError with parameters (INTR\_ERR, M\_ZERO, RterrForever) to handle the error.

### 10.20.6 MemManage

Low Level Design Details about CSU MemManage will follow in the sub sections.

#### 10.20.6.1 Brief Description

The MemManage function is an interrupt error handler for memory management errors.

#### 10.20.6.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.20.6.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.20.6.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.20.6.5 Return Value

None

#### 10.20.6.6 Other CSUs called by this CSU

RterrError

RterrForever

#### 10.20.6.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to MemManage

##### 10.20.6.7.1 daugwyintr-MemManage-LLR-001

Requirement ID: H398-LLD-GWY-FNC-1846

The function shall call RterrError with parameters (INTR\_ERR,M\_TWO and pointer to function RterrForever) to flash code accordingly for memory management errors.

### 10.20.7 BusFault

Low Level Design Details about CSU BusFault will follow in the sub sections.

#### 10.20.7.1 Brief Description

The BusFault function is an interrupt error handler for bus fault errors.

#### 10.20.7.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.20.7.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.20.7.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.20.7.5 Return Value

None

#### 10.20.7.6 Other CSUs called by this CSU

RterrError

RterrForever

#### 10.20.7.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to BusFault

##### 10.20.7.7.1 daugwyintr-BusFault-LLR-001

Requirement ID: H398-LLD-GWY-FNC-1855

The function shall call RterrError with parameters (INTR\_ERR, M\_THREE and pointer to function RterrForever) to flash code accordingly for bus fault errors.

### 10.20.8 UsageFault

Low Level Design Details about CSU UsageFault will follow in the sub sections.

#### 10.20.8.1 Brief Description

The UsageFault function is an interrupt error handler for usage fault errors.

#### 10.20.8.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.20.8.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.20.8.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.20.8.5 Return Value

None

#### 10.20.8.6 Other CSUs called by this CSU

RterrError

RterrForever

#### 10.20.8.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to UsageFault

##### 10.20.8.7.1 daugwyintr-UsageFault-LLR-001

Requirement ID: H398-LLD-GWY-FNC-1864

The function shall call RterrError with parameters (INTR\_ERR, M\_FOUR and pointer to function RterrForever) to flash code accordingly for usage fault errors.

### 10.20.9 SpuriousInterrupt

Low Level Design Details about CSU SpuriousInterrupt will follow in the sub sections.

#### 10.20.9.1 Brief Description

The SpuriousInterrupt function is an interrupt error handler for spurious interrupts.

#### 10.20.9.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.20.9.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.20.9.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.20.9.5 Return Value

None

#### 10.20.9.6 Other CSUs called by this CSU

RterrError

RterrForever

#### 10.20.9.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to SpuriousInterrupt

##### 10.20.9.7.1 daugwyintr-SpuriousInterrupt-LLR-001

Requirement ID: H398-LLD-GWY-FNC-1873

The function shall call RterrError with parameters (INTR\_ERR, M\_FIVE and pointer to function RterrForever) to flash code accordingly for spurious interrupts.

## 10.21 daugwylogic

The daugwylogic CSC implements main logic functions.

### 10.21.1 CheckMinMax

Low Level Design Details about CSU CheckMinMax will follow in the sub sections.

#### 10.21.1.1 Brief Description

The CheckMinMax checks the minimum and maximum range .

#### 10.21.1.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.21.1.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.21.1.4 Parameter list (Input/Output)

Inputs : T\_FLOAT32 param - input parameter

T\_FLOAT32 min - minimum range

T\_FLOAT32 max - maximum range

Outputs : None

#### 10.21.1.5 Return Value

T\_BOOL - returns TRUE or FALSE

#### 10.21.1.6 Other CSUs called by this CSU

None

#### 10.21.1.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to CheckMinMax

##### 10.21.1.7.1 daugwylogic-CheckMinMax-LLR-001

Requirement ID: H398-LLD-GWY-FNC-1883

The function shall do the following:

1. Return FALSE when param is less than min OR param is greater than max
2. Return TRUE

### 10.21.2 HwCalibrationOffset

Low Level Design Details about CSU CalOffset will follow in the sub sections.

#### 10.21.2.1 Brief Description

The HwCalibrationOffset determines the offset value and returns it based on the channel value.

#### 10.21.2.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.21.2.3 List of global variables accessed and modified

Accessed : Nvm\_cmu\_tx

Modified : none

#### 10.21.2.4 Parameter list (Input/Output)

Inputs : T\_SINT32 value\_off - Offset value

T\_UINT16 channel\_off - Channel

T\_FLOAT32 res - Result

Outputs : T\_SINT32 value\_off - Offset value

#### 10.21.2.5 Return Value

T\_SINT32- Returns calibration offset value

#### 10.21.2.6 Other CSUs called by this CSU

None

#### 10.21.2.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to HwCalibrationOffset

##### 10.21.2.7.1 daugwylogic-HwCalibrationOffset-LLR-001

Requirement ID: H398-LLD-GWY-FNC-1892

The function shall set value\_off with the result of addition of value\_off and (res multiplied with offset with index CMU\_COFI\_EOP\_1 of hardware of NVM\_calibration of Nvm\_cmu\_tx) when channel\_off is EOP1

##### 10.21.2.7.2 daugwylogic-HwCalibrationOffset-LLR-002

Requirement ID: H398-LLD-GWY-FNC-1893

The function shall set value\_off with the result of addition of value\_off and (res multiplied with offset with index CMU\_COFI\_EOP\_2 of hardware of NVM\_calibration of Nvm\_cmu\_tx) when channel\_off is EOP2.

##### 10.21.2.7.3 daugwylogic-HwCalibrationOffset-LLR-003

Requirement ID: H398-LLD-GWY-FNC-1894

The function shall set value\_off with the result of addition of value\_off and (res multiplied with offset with index CMU\_COFI\_EOT\_1 of hardware of NVM\_calibration of Nvm\_cmu\_tx) when channel\_off is EOT1 .

##### 10.21.2.7.4 daugwylogic-HwCalibrationOffset-LLR-004

Requirement ID: H398-LLD-GWY-FNC-1895

The function shall set value\_off with the result of addition of value\_off and (res multiplied with offset with index CMU\_COFI\_EOT\_2 of hardware of NVM\_calibration of Nvm\_cmu\_tx) when channel\_off is EOT2 .

##### 10.21.2.7.5 daugwylogic-HwCalibrationOffset-LLR-005

Requirement ID: H398-LLD-GWY-FNC-1896

The function shall set value\_off with the result of addition of value\_off and (res multiplied with offset with index CMU\_COFI\_FP\_1 of hardware of NVM\_calibration of Nvm\_cmu\_tx) when channel\_off is FP1\_SIG.

##### 10.21.2.7.6 daugwylogic-HwCalibrationOffset-LLR-006

Requirement ID: H398-LLD-GWY-FNC-1897

The function shall set value\_off with the result of addition of value\_off and (res multiplied with offset with index CMU\_COFI\_FP\_2 of hardware of NVM\_calibration of Nvm\_cmu\_tx) when channel\_off is FP2\_SIG.

##### 10.21.2.7.7 daugwylogic-HwCalibrationOffset-LLR-007

Requirement ID: H398-LLD-GWY-FNC-1898

The function shall set value\_off with the result of addition of value\_off and (res multiplied with offset with index CMU\_COFI\_HP\_1 of hardware of NVM\_calibration of Nvm\_cmu\_tx) when channel\_off is LHP\_SIG .

##### 10.21.2.7.8 daugwylogic-HwCalibrationOffset-LLR-008

Requirement ID: H398-LLD-GWY-FNC-1899

The function shall set value\_off with the result of addition of value\_off and (res multiplied with offset with index CMU\_COFI\_HP\_2 of hardware of NVM\_calibration of Nvm\_cmu\_tx) when channel\_off is RHP\_SIG.

##### 10.21.2.7.9 daugwylogic-HwCalibrationOffset-LLR-009

Requirement ID: H398-LLD-GWY-FNC-1900

The function shall set value\_off with the result of addition of value\_off and (res multiplied with offset with index CMU\_COFI\_MGBT of hardware of NVM\_calibration of Nvm\_cmu\_tx) when channel\_off is MGBT.

##### 10.21.2.7.10 daugwylogic-HwCalibrationOffset-LLR-010

Requirement ID: H398-LLD-GWY-FNC-1901

The function shall set value\_off with the result of addition of value\_off and (res multiplied with offset with index CMU\_COFI\_MGBOP of hardware of NVM\_calibration of Nvm\_cmu\_tx) when channel\_off is MGBOP\_SIG.

##### 10.21.2.7.11 daugwylogic-HwCalibrationOffset-LLR-015

Requirement ID: H398-LLD-GWY-FNC-5814

The function shall do nothing when channel\_off is other than EOP1, EOP2, EOT1, EOT2, FP1\_SIG, FP2\_SIG, LHP\_SIG, RHP\_SIG, MGBT, MGBOP\_SIG.

##### 10.21.2.7.12 daugwylogic-HwCalibrationOffset-LLR-016

Requirement ID: H398-LLD-GWY-FNC-5815

The function shall return value\_off value.

### 10.21.3 EngData

Low Level Design Details about CSU EngData will follow in the sub sections.

#### 10.21.3.1 Brief Description

The EngData processes the engine data received from arinc825, perform the range check and set the channel status .

#### 10.21.3.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.21.3.3 List of global variables accessed and modified

Accessed : Cmdownload\_lp\_cmd\_data, Xram\_data\_buffer, Dau\_eng\_status

Modified : Dau\_eng\_data, Dau\_eng\_status

#### 10.21.3.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.21.3.5 Return Value

None

#### 10.21.3.6 Other CSUs called by this CSU

AppGetAnalog

CheckMinMax

HwCalibrationOffset

#### 10.21.3.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to EngData

##### 10.21.3.7.1 daugwylogic-EngData-LLR-001

Requirement ID: H398-LLD-GWY-FNC-1910

The function shall loop from index is M\_ZERO to numofengines of Cmdownload\_lp\_cmd\_data minus 1, loop from loop counter is M\_ZERO to number\_params of engine of Cmdownload\_lp\_cmd\_data minus 1 (increment loop counter by M\_ONE) and do the following

1. Set temporary parameter data with reference of paramData (with index is loop counter) of engine of Cmdownload\_lp\_cmd\_data

##### 10.21.3.7.2 daugwylogic-EngData-LLR-002

Requirement ID: H398-LLD-GWY-FNC-1911

The function shall loop from index is M\_ZERO to numofengines of Cmdownload\_lp\_cmd\_data minus 1, loop from loop counter is M\_ZERO to number\_params of engine of Cmdownload\_lp\_cmd\_data minus 1 (increment loop counter by M\_ONE) and do the following when analStat of App\_A825 of Xram\_data\_buffer with index (locnum of temporary parameter data) is OK

1. Call AppGetAnalog with parameters (locnum of temporary parameter data, table\_num of paramData of engine of Cmdownload\_lp\_cmd\_data)
2. Store the return value of APPgetAnalog with parameters locnum of temporary parameter data and table\_num of temporary parameter data to temporary engine data

##### 10.21.3.7.3 daugwylogic-EngData-LLR-003

Requirement ID: H398-LLD-GWY-FNC-1912

The function shall loop from index is M\_ZERO to numofengines of Cmdownload\_lp\_cmd\_data minus 1, loop from loop counter is M\_ZERO to number\_params of engine of Cmdownload\_lp\_cmd\_data minus 1 (increment loop counter by M\_ONE) and do nothing when

a) analStat of App\_A825 of Xram\_data\_buffer with index (locnum of temporary parameter data) is OK

b) (M\_ZERO is equal to resolution of temporary parameter data by calling the macro M\_FP\_COMPARE)

##### 10.21.3.7.4 daugwylogic-EngData-LLR-004

Requirement ID: H398-LLD-GWY-FNC-1913

The function shall loop from index is M\_ZERO to numofengines of Cmdownload\_lp\_cmd\_data minus 1, loop from loop counter is M\_ZERO to number\_params of engine of Cmdownload\_lp\_cmd\_data minus 1 (increment loop counter by M\_ONE) and do the following when analStat of App\_A825 of Xram\_data\_buffer with index (locnum of temporary parameter data) is OK and resolution of temporary parameter data is not equal to M\_ZERO

1. Call HwCalibrationOffset with parameters (temporary engine data, locnum of temporary parameter data, resolution of temporary parameter data)
2. Store the return value of HwCalibrationOffset in temporary engine data

##### 10.21.3.7.5 daugwylogic-EngData-LLR-005

Requirement ID: H398-LLD-GWY-FNC-1914

The function shall loop from index is M\_ZERO to numofengines of Cmdownload\_lp\_cmd\_data minus 1, loop from loop counter is M\_ZERO to number\_params of engine of Cmdownload\_lp\_cmd\_data minus 1 (increment loop counter by M\_ONE) and do the following when analStat of App\_A825 of Xram\_data\_buffer with index (locnum of temporary parameter data) is OK and resolution of temporary parameter data is not equal to M\_ZERO

1. Call CheckMinMax with parameters (division of temporary engine data by resolution of temporary parameter data, startRangeEng of temporary parameter data, endRangeEng of temporary parameter data)
2. Set Dau\_eng\_data of index (locnum of temporary parameter data) with M\_ZERO and Dau\_eng\_status of index (locnum of temporary parameter data) with CHAN\_ERR when the function CheckMinMax returns M\_ZERO

##### 10.21.3.7.6 daugwylogic-EngData-LLR-006

Requirement ID: H398-LLD-GWY-FNC-1915

The function shall loop from index is M\_ZERO to numofengines of Cmdownload\_lp\_cmd\_data minus 1, loop from loop counter is M\_ZERO to number\_params of engine of Cmdownload\_lp\_cmd\_data minus 1 (increment loop counter by M\_ONE) and Set Dau\_eng\_data of index (locnum of temporary parameter data) to M\_ZERO when

1. analStat of App\_A825 of Xram\_data\_buffer with index (locnum of temporary parameter data) is OK
2. resolution of temporary parameter data is not equal to M\_ZERO
3. The function CheckMinMax returns other than M\_ZERO
4. Dau\_eng\_status with index (locnum of temporary parameter data) is equal to CHAN\_ERR

##### 10.21.3.7.7 daugwylogic-EngData-LLR-007

Requirement ID: H398-LLD-GWY-FNC-1916

The function shall loop from index is M\_ZERO to numofengines of Cmdownload\_lp\_cmd\_data minus 1,loop from loop counter is M\_ZERO to number\_params of engine of Cmdownload\_lp\_cmd\_data minus 1 (increment loop counter by M\_ONE) and Set Dau\_eng\_data of index (locnum of temporary parameter data) to M\_ZERO when

1. analStat of App\_A825 of Xram\_data\_buffer with index (locnum of temporary parameter data) is OK
2. resolution of temporary parameter data is not equal to M\_ZERO
3. The function CheckMinMax returns other than M\_ZERO
4. Dau\_eng\_status with index (locnum of temporary parameter data) is other than CHAN\_ERR

##### 10.21.3.7.8 daugwylogic-EngData-LLR-008

Requirement ID: H398-LLD-GWY-FNC-1917

The function shall loop from index is M\_ZERO to numofengines of Cmdownload\_lp\_cmd\_data minus 1, loop from loop counter is M\_ZERO to number\_params of engine of Cmdownload\_lp\_cmd\_data minus 1 (increment loop counter by M\_ONE) and Set Dau\_eng\_data of index (locnum of temporary parameter data) with division of temporary engine data by resolution of temporary parameter data and set Dau\_eng\_status of index (locnum of temporary parameter data) with OK when

1. analStat of App\_A825 of Xram\_data\_buffer with index (locnum of temporary parameter data) is OK
2. resolution of temporary parameter data is not equal to M\_ZERO
3. The function CheckMinMax returns other than M\_ZERO
4. Dau\_eng\_status with index (locnum of temporary parameter data) is other than CHAN\_ERR and MISSING

##### 10.21.3.7.9 daugwylogic-EngData-LLR-009

Requirement ID: H398-LLD-GWY-FNC-1918

The function shall loop from index is M\_ZERO to numofengines of Cmdownload\_lp\_cmd\_data minus 1, loop from loop counter is M\_ZERO to number\_params of engine of Cmdownload\_lp\_cmd\_data minus 1 (increment loop counter by M\_ONE) and reset temporary engine data to M\_ZERO when analStat of App\_A825 of Xram\_data\_buffer with index (locnum of temporary parameter data) is OK and resolution of temporary parameter data is not equal to M\_ZERO

##### 10.21.3.7.10 daugwylogic-EngData-LLR-010

Requirement ID: H398-LLD-GWY-FNC-1919

The function shall loop from index is M\_ZERO to numofengines of Cmdownload\_lp\_cmd\_data minus 1, loop from loop counter is M\_ZERO to number\_params of engine of Cmdownload\_lp\_cmd\_data(increment loop counter by M\_ONE) and do the following when analStat of App\_A825 of Xram\_data\_buffer with index (locnum of temporary parameter data) is not equal to OK

1. Set Dau\_eng\_data of index (locnum of temporary parameter data) with M\_ZERO
2. Set Dau\_eng\_status of index (locnum of temporary parameter data) with analStat of App\_A825 of Xram\_data\_buffer of index (locnum of temporary parameter data)

### 10.21.4 AircraftData

Low Level Design Details about CSU AircraftData will follow in the sub sections.

#### 10.21.4.1 Brief Description

The AircraftData processes the aircraft data received from arinc825, perform the range check and set the channel status .

#### 10.21.4.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.21.4.3 List of global variables accessed and modified

Accessed : Cmdownload\_lp\_cmd\_data, Xram\_data\_buffer, Dau\_eng\_status

Modified : Dau\_eng\_data, Dau\_eng\_status

#### 10.21.4.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.21.4.5 Return Value

None

#### 10.21.4.6 Other CSUs called by this CSU

AppGetAnalog

HwCalibrationOffset

CheckMinMax

#### 10.21.4.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to AircraftData

##### 10.21.4.7.1 daugwylogic-AircraftData-LLR-001

Requirement ID: H398-LLD-GWY-FNC-1929

The function shall loop from M\_ZERO to numofair of aircraft of Cmdownload\_lp\_cmd\_data minus 1 and set temporary parameter data with paramData (with index as loop counter) of aircraft of Cmdownload\_lp\_cmd\_data

##### 10.21.4.7.2 daugwylogic-AircraftData-LLR-002

Requirement ID: H398-LLD-GWY-FNC-1930

The function shall loop from index is M\_ZERO to numofair of aircraft of Cmdownload\_lp\_cmd\_data minus 1 and do the following when analStat of App\_A825 of Xram\_data\_buffer with index (locnum of temporary parameter data) is OK

1. Call AppGetAnalog with parameters (locnum of temporary parameter data, table\_num of temporary parameter data)
2. Store the return value of APPgetAnalog in temporary aircraft data

##### 10.21.4.7.3 daugwylogic-AircraftData-LLR-003

Requirement ID: H398-LLD-GWY-FNC-1931

The function shall loop from index is M\_ZERO to numofair of aircraft of Cmdownload\_lp\_cmd\_data minus 1 and do nothing when

1. analStat of App\_A825 of Xram\_data\_buffer with index (locnum of temporary parameter data) is OK
2. (M\_ZERO is equal to resolution of temporary parameter data by calling the macro M\_FP\_COMPARE)

##### 10.21.4.7.4 daugwylogic-AircraftData-LLR-004

Requirement ID: H398-LLD-GWY-FNC-1932

The function shall loop from index is M\_ZERO to numofair of aircraft of Cmdownload\_lp\_cmd\_data minus 1 and do the following when analStat of App\_A825 of Xram\_data\_buffer with index (locnum of temporary parameter data) is OK and resolution of temporary parameter data is not equal to M\_ZERO

1. Call HwCalibrationOffset with parameters (temporary aircraft data, locnum of temporary parameter data, resolution of temporary parameter data)
2. Store the return value of HwCalibrationOffset in temporary aircraft data

##### 10.21.4.7.5 daugwylogic-AircraftData-LLR-005

Requirement ID: H398-LLD-GWY-FNC-1933

The function shall loop from index is M\_ZERO to numofair of aircraft of Cmdownload\_lp\_cmd\_data minus 1 and do the following when analStat of App\_A825 of Xram\_data\_buffer with index (locnum of temporary parameter data) is OK and resolution of temporary parameter data is not equal to M\_ZERO

1. Call CheckMinMax with parameters (division of temporary aircraft data by resolution of temporary parameter data, startRangeEng of temporary parameter data, endRangeEng of temporary parameter data)
2. Set Dau\_eng\_data of index (locnum of temporary parameter data) with M\_ZERO and Dau\_eng\_status of index (locnum of temporary parameter data) with CHAN\_ERR when the function CheckMinMax returns M\_ZERO

##### 10.21.4.7.6 daugwylogic-AircraftData-LLR-006

Requirement ID: H398-LLD-GWY-FNC-1934

The function shall loop from index is M\_ZERO to numofair of aircraft of Cmdownload\_lp\_cmd\_data minus 1 and Set Dau\_eng\_data of index (locnum of temporary parameter data) with M\_ZERO when

1. analStat of App\_A825 of Xram\_data\_buffer with index (locnum of temporary parameter data) is OK
2. resolution of temporary parameter data is not equal to M\_ZERO
3. The function CheckMinMax returns other than M\_ZERO
4. Dau\_eng\_status with index (locnum of temporary parameter data) is equal to CHAN\_ERR

##### 10.21.4.7.7 daugwylogic-AircraftData-LLR-007

Requirement ID: H398-LLD-GWY-FNC-1935

The function shall loop from index is M\_ZERO to numofair of aircraft of Cmdownload\_lp\_cmd\_data minus 1 and Set Dau\_eng\_data of index (locnum of temporary parameter data) with M\_ZERO when

1. analStat of App\_A825 of Xram\_data\_buffer with index (locnum of temporary parameter data) is OK
2. resolution of temporary parameter data is not equal to M\_ZERO
3. The function CheckMinMax returns other than M\_ZERO
4. Dau\_eng\_status with index (locnum of temporary parameter data) is equal to MISSING

##### 10.21.4.7.8 daugwylogic-AircraftData-LLR-008

Requirement ID: H398-LLD-GWY-FNC-1936

The function shall loop from index is M\_ZERO to numofair of aircraft of Cmdownload\_lp\_cmd\_data minus 1 and Set Dau\_eng\_data of index (locnum of temporary parameter data) with division of temporary aircraft data by resolution of temporary parameter data and set Dau\_eng\_status of index (locnum of temporary parameter data) with OK when

1. analStat of App\_A825 of Xram\_data\_buffer with index (locnum of temporary parameter data) is OK
2. resolution of temporary parameter data is not equal to M\_ZERO
3. The function CheckMinMax returns other than M\_ZERO
4. Dau\_eng\_status with index (locnum of temporary parameter data) is other than CHAN\_ERR and MISSING

##### 10.21.4.7.9 daugwylogic-AircraftData-LLR-009

Requirement ID: H398-LLD-GWY-FNC-1937

The function shall loop from index is M\_ZERO to numofair of aircraft of Cmdownload\_lp\_cmd\_data minus 1 and set temporary aircraft data to M\_ZERO when analStat of App\_A825 of Xram\_data\_buffer with index (locnum of temporary parameter data) is OK and resolution of temporary parameter data is not equal to M\_ZERO

##### 10.21.4.7.10 daugwylogic-AircraftData-LLR-010

Requirement ID: H398-LLD-GWY-FNC-1938

The function shall loop from index is M\_ZERO to numofair of aircraft of Cmdownload\_lp\_cmd\_data minus 1 and do the following when analStat of App\_A825 of Xram\_data\_buffer with index (locnum of temporary parameter data) is not equal to OK

1. Set Dau\_eng\_data of index (locnum of temporary parameter data) with M\_ZERO
2. Set Dau\_eng\_status of index (locnum of temporary parameter data) with analStat of App\_A825 of Xram\_data\_buffer of index (locnum of temporary parameter data)

### 10.21.5 LogicCustApp

Low Level Design Details about CSU LOGICcustApp will follow in the sub sections.

#### 10.21.5.1 Brief Description

The LogicCustApp sends the latest discerete output to discerete module .

#### 10.21.5.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.21.5.3 List of global variables accessed and modified

Accessed : Discrete\_outs

Modified : None

#### 10.21.5.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.21.5.5 Return Value

None

#### 10.21.5.6 Other CSUs called by this CSU

DoAppUpdate

#### 10.21.5.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to LOGICcustApp

##### 10.21.5.7.1 daugwylogic-LogicCustApp-LLR-001

Requirement ID: H398-LLD-GWY-FNC-1947

The function shall perform the following when loop from index is M\_ZERO to DO\_MAX minus 1 and do the following when Previous discrete outputs is not equal to Discrete\_outs for all the indices

1. Set send do variable to TRUE
2. Set Previous discrete outputs with Discrete\_outs for all the indices
3. Stop the execution from the current condition

##### 10.21.5.7.2 daugwylogic-LogicCustApp-LLR-002

Requirement ID: H398-LLD-GWY-FNC-1948

The function shall call DoAppUpdate with parameters (NSC\_SET\_DOUT1, Discrete\_outs) when send do variable is not equal to FALSE.

##### 10.21.5.7.3 daugwylogic-LogicCustApp-LLR-003

Requirement ID: H398-LLD-GWY-FNC-1949

The function shall return from the function.

### 10.21.6 MessageAvailable

Low Level Design Details about CSU MessageAvailable will follow in the sub sections.

#### 10.21.6.1 Brief Description

The MessageAvailable Checks if UAS message is available, unpacks and checks CRC.

(Implements DLE expansion-Recipient processing algorithm)

#### 10.21.6.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.21.6.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.21.6.4 Parameter list (Input/Output)

Inputs : T\_QUEUE \*queue -pointer to queue

Outputs : T\_QUEUE \*queue -pointer to queue

#### 10.21.6.5 Return Value

T\_BOOL -Returns state of processing of incoming UAS Messages

returns TRUE if processing is finished

returns FALSE if processing is not finished and queue is pointing to NULL

#### 10.21.6.6 Other CSUs called by this CSU

IsEmpty

DeQueue

UpdCrc32

InitQueue

#### 10.21.6.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to MessageAvailable

##### 10.21.6.7.1 daugwylogic-MessageAvailable-LLR-001

Requirement ID: H398-LLD-GWY-FNC-1958

The function shall set receive queue with queue.

##### 10.21.6.7.2 daugwylogic-MessageAvailable-LLR-002

Requirement ID: H398-LLD-GWY-FNC-1959

The function shall return FALSE when receive queue is M\_NULL, otherwise do nothing.

##### 10.21.6.7.3 daugwylogic-MessageAvailable-LLR-003

Requirement ID: H398-LLD-GWY-FNC-1960

The function shall loop until current state is not HRX\_FINISHED AND NOT of return value of the function IsEmpty with parameter (receive queue) is TRUE and do the following when current state is not HRX\_FINISHED AND NOT of return value of the function IsEmpty with parameter (receive queue) is TRUE

1. call DeQueue with parameter receive queue and set the return value to receive byte

##### 10.21.6.7.4 daugwylogic-MessageAvailable-LLR-004

Requirement ID: H398-LLD-GWY-FNC-1961

The function shall loop until current state is not HRX\_FINISHED AND NOT of return value of the function IsEmpty with parameter (receive queue) is TRUE and set current state to HRX\_DLE\_1ST

When

1. current state is not HRX\_FINISHED AND NOT of return value of the function IsEmpty with parameter (receive queue) is TRUE
2. current state is HRX\_WAIT
3. receive byte is M\_DLE

##### 10.21.6.7.5 daugwylogic-MessageAvailable-LLR-005

Requirement ID: H398-LLD-GWY-FNC-1962

The function shall loop until current state is not HRX\_FINISHED AND NOT of return value of the function IsEmpty with parameter (receive queue) is TRUE and set current state to HRX\_DLE\_2ND

When

1. current state is not HRX\_FINISHED AND NOT of return value of the function IsEmpty with parameter (receive queue) is TRUE
2. current state is HRX\_GET\_DATA
3. receive byte is M\_DLE

##### 10.21.6.7.6 daugwylogic-MessageAvailable-LLR-006

Requirement ID: H398-LLD-GWY-FNC-1963

The function shall loop until current state is not HRX\_FINISHED AND NOT of return value of the function IsEmpty with parameter (receive queue) is TRUE and do the following:

1. Set size of buffer with sizeof the buffer of receive queue
2. Set buffer of receive queue (index is receive buffer count) with receive byte and increment receive buffer count by one when size of buffer is greater than receive buffer count
3. Set current state with HRX\_WAIT when size of buffer is not greater than receive buffer count

When

1. current state is not HRX\_FINISHED AND NOT of return value of the function IsEmpty with parameter (receive queue) is TRUE
2. current state is HRX\_GET\_DATA
3. receive byte is not M\_DLE

##### 10.21.6.7.7 daugwylogic-MessageAvailable-LLR-007

Requirement ID: H398-LLD-GWY-FNC-1964

The function shall loop until current state is not HRX\_FINISHED AND NOT of return value of the function IsEmpty with parameter (receive queue) is TRUE and do the following when current state is not HRX\_FINISHED AND NOT of return value of the function IsEmpty with parameter (receive queue) is TRUE and current state is HRX\_DLE\_1ST:

1. Set receive buffer count with M\_ZERO and set current state with HRX\_GET\_DATA when receive byte is equal to M\_SOH
2. Set current state with HRX\_WAIT when receive byte is not equal to M\_SOH

##### 10.21.6.7.8 daugwylogic-MessageAvailable-LLR-008

Requirement ID: H398-LLD-GWY-FNC-1965

The function shall loop until current state is not HRX\_FINISHED AND NOT of return value of the function IsEmpty with parameter (receive queue) is TRUE and do the following:

1. Set receive buffer count with M\_ZERO
2. Set current state with HRX\_GET\_DATA

When

1. current state is not HRX\_FINISHED AND NOT of return value of the function IsEmpty with parameter (receive queue) is TRUE
2. current state is HRX\_DLE\_2ND
3. receive byte is M\_SOH

##### 10.21.6.7.9 daugwylogic-MessageAvailable-LLR-009

Requirement ID: H398-LLD-GWY-FNC-1966

The function shall loop until current state is not HRX\_FINISHED AND NOT of return value of the function IsEmpty with parameter (receive queue) is TRUE and do the following:

1. Set crc value with M\_CRC\_VALUE
2. Set receive buffer count with subtraction of (receive buffer count with M\_FOUR)
3. Set count value with receive buffer count
4. Set byte pointer with buffer of receive queue

When

1. current state is not HRX\_FINISHED AND NOT of return value of the function IsEmpty with parameter (receive queue) is TRUE
2. current state is HRX\_DLE\_2ND
3. receive byte is M\_EOT
4. receive buffer count is greater than or equal to M\_SIX

##### 10.21.6.7.10 daugwylogic-MessageAvailable-LLR-010

Requirement ID: H398-LLD-GWY-FNC-1967

The function shall loop until current state is not HRX\_FINISHED AND NOT of return value of the function IsEmpty with parameter (receive queue) is TRUE, loop until count value is not zero and do the following:

1. call UpdCrc32 with parameters (post increment of byte pointer value, crc value)
2. Set return value of the function UpdCrc32 to crc value
3. Decrement the count value by one

When

1. current state is not HRX\_FINISHED AND NOT of return value of the function IsEmpty with parameter (receive queue) is TRUE
2. current state is HRX\_DLE\_2ND
3. receive byte is M\_EOT
4. receive buffer count is greater than or equal to M\_SIX
5. count value is greater than M\_ZERO

##### 10.21.6.7.11 daugwylogic-MessageAvailable-LLR-011

Requirement ID: H398-LLD-GWY-FNC-1968

The function shall loop until current state is not HRX\_FINISHED AND NOT of return value of the function IsEmpty with parameter (receive queue) is TRUE and do nothing when

1. current state is not HRX\_FINISHED AND NOT of return value of the function IsEmpty with parameter (receive queue) is TRUE
2. current state is HRX\_DLE\_2ND
3. receive byte is M\_EOT
4. receive buffer count is greater than or equal to M\_SIX
5. count value is not greater than M\_ZERO

##### 10.21.6.7.12 daugwylogic-MessageAvailable-LLR-012

Requirement ID: H398-LLD-GWY-FNC-1969

The function shall loop until current state is not HRX\_FINISHED AND NOT of return value of the function IsEmpty with parameter (receive queue) is TRUE and do the following

1. Set crc msg with post increment of byte pointer value
2. Set crc msg with (Bitwise OR of crc msg with (bit shift of (post increment of byte pointer value) to left by M\_SHIFT\_BY\_8))
3. Set crc msg with (Bitwise OR of crc msg with (bit shift of (post increment of byte pointer value) to left by M\_SHIFT\_BY\_16))
4. Set crc msg with (Bitwise OR of crc msg with (bit shift of (post increment of byte pointer value) to left by M\_SHIFT\_BY\_24))

when

1. current state is not HRX\_FINISHED AND NOT of return value of the function IsEmpty with parameter (receive queue) is TRUE
2. current state is HRX\_DLE\_2ND
3. receive byte is M\_EOT
4. receive buffer count is greater than or equal to M\_SIX

##### 10.21.6.7.13 daugwylogic-MessageAvailable-LLR-013

Requirement ID: H398-LLD-GWY-FNC-1970

The function shall loop until current state is not HRX\_FINISHED AND NOT of return value of the function IsEmpty with parameter (receive queue) is TRUE and do the following

1. Set current state with HRX\_FINISHED when crc msg is equal to crc value
2. Set current state with HRX\_WAIT when crc msg is not equal to crc value

when

1. current state is not HRX\_FINISHED AND NOT of return value of the function IsEmpty with parameter (receive queue) is TRUE
2. current state is HRX\_DLE\_2ND
3. receive byte is M\_EOT
4. receive buffer count is greater than or equal to M\_SIX

##### 10.21.6.7.14 daugwylogic-MessageAvailable-LLR-014

Requirement ID: H398-LLD-GWY-FNC-1971

The function shall loop until current state is not HRX\_FINISHED AND NOT of return value of the function IsEmpty with parameter (receive queue) is TRUE and do nothing when

1. current state is not HRX\_FINISHED AND NOT of return value of the function IsEmpty with parameter (receive queue) is TRUE
2. current state is HRX\_DLE\_2ND
3. receive byte is M\_EOT
4. receive buffer count is less than M\_SIX

##### 10.21.6.7.15 daugwylogic-MessageAvailable-LLR-015

Requirement ID: H398-LLD-GWY-FNC-1972

The function shall loop until current state is not HRX\_FINISHED AND NOT of return value of the function IsEmpty with parameter (receive queue) is TRUE and set size of buffer with sizeof the buffer of recieve queue when

1. current state is not HRX\_FINISHED AND NOT of return value of the function IsEmpty with parameter (receive queue) is TRUE
2. current state is HRX\_DLE\_2ND
3. receive byte is M\_DLE\_PLC\_HLDR

##### 10.21.6.7.16 daugwylogic-MessageAvailable-LLR-016

Requirement ID: H398-LLD-GWY-FNC-1973

The function shall loop until current state is not HRX\_FINISHED AND NOT of return value of the function IsEmpty with parameter (receive queue) is TRUE and do the following

1. Set buffer of recieve queue of index (receive buffer count) with M\_DLE
2. Increment receive buffer count by one
3. Set current state with HRX\_GET\_DATA

when

1. current state is not HRX\_FINISHED AND NOT of return value of the function IsEmpty with parameter (receive queue) is TRUE
2. current state is HRX\_DLE\_2ND
3. receive byte is M\_DLE\_PLC\_HLDR
4. size of buffer is greater than receive buffer count

##### 10.21.6.7.17 daugwylogic-MessageAvailable-LLR-017

Requirement ID: H398-LLD-GWY-FNC-1974

The function shall loop until current state is not HRX\_FINISHED AND NOT of return value of the function IsEmpty with parameter (receive queue) is TRUE and do the following

1. Set current state with HRX\_WAIT

when

1. current state is not HRX\_FINISHED AND NOT of return value of the function IsEmpty with parameter (receive queue) is TRUE
2. current state is HRX\_DLE\_2ND
3. receive byte is M\_DLE\_PLC\_HLDR
4. size of buffer is not greater than receive buffer count

##### 10.21.6.7.18 daugwylogic-MessageAvailable-LLR-018

Requirement ID: H398-LLD-GWY-FNC-1975

The function shall loop until current state is not HRX\_FINISHED AND NOT of return value of the function IsEmpty with parameter (receive queue) is TRUE and set current state with HRX\_GET\_DATA

when

1. current state is not HRX\_FINISHED AND NOT of return value of the function IsEmpty with parameter (receive queue) is TRUE
2. current state is HRX\_DLE\_2ND
3. receive byte is other than M\_SOH, M\_EOT and M\_DLE\_PLC\_HLDR

##### 10.21.6.7.19 daugwylogic-MessageAvailable-LLR-019

Requirement ID: H398-LLD-GWY-FNC-1976

The function shall loop until current state is not HRX\_FINISHED AND NOT of return value of the function IsEmpty with parameter (receive queue) is TRUE and set current state with HRX\_WAIT

when

1. current state is not HRX\_FINISHED AND NOT of return value of the function IsEmpty with parameter (receive queue) is TRUE
2. current state is other than HRX\_WAIT, HRX\_GET\_DATA, HRX\_DLE\_1ST and HRX\_DLE\_2ND

##### 10.21.6.7.20 daugwylogic-MessageAvailable-LLR-020

Requirement ID: H398-LLD-GWY-FNC-1977

The function shall do nothing when (current state is not HRX\_FINISHED AND NOT of return value of the function IsEmpty with parameter (receive queue)) is FALSE

##### 10.21.6.7.21 daugwylogic-MessageAvailable-LLR-021

Requirement ID: H398-LLD-GWY-FNC-1978

The function shall call InitQueue with parameter recieve queue

##### 10.21.6.7.22 daugwylogic-MessageAvailable-LLR-022

Requirement ID: H398-LLD-GWY-FNC-1979

The function shall return TRUE when current state is HRX\_FINISHED, otherwise return FALSE

### 10.21.8 TransmitMessage

Low Level Design Details about CSU TransmitMessage will follow in the sub sections.

#### 10.21.8.1 Brief Description

The TransmitMessage builds, adds DLE expansion, and CRC to prepare message for Transmission.

#### 10.21.8.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.21.8.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.21.8.4 Parameter list (Input/Output)

Inputs : T\_UINT8 \*message -data to be sent

T\_UINT16 size -data message size

T\_UINT8 comm -communication type

Outputs : T\_UINT8 \*message -data to be sent

T\_UINT16 size -data message size

#### 10.21.8.5 Return Value

None

#### 10.21.8.6 Other CSUs called by this CSU

InitQueue

EnQueue

DeQueue

ResetCOM

CommPutChars

UpdCrc32

#### 10.21.8.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TransmitMessage

##### 10.21.8.7.1 daugwylogic-TransmitMessage-LLR-001

Requirement ID: H398-LLD-GWY-FNC-2002

The function shall call InitQueue with parameter (reference of transmit queue).

##### 10.21.8.7.2 daugwylogic-TransmitMessage-LLR-002

Requirement ID: H398-LLD-GWY-FNC-2003

The function shall call EnQueue with parameters (reference of transmit queue and M\_DLE).

##### 10.21.8.7.3 daugwylogic-TransmitMessage-LLR-003

Requirement ID: H398-LLD-GWY-FNC-2004

The function shall call EnQueue with parameters (reference of transmit queue and M\_SOH).

##### 10.21.8.7.4 daugwylogic-TransmitMessage-LLR-004

Requirement ID: H398-LLD-GWY-FNC-2005

The function shall set crc value with M\_CRC\_VALUE.

##### 10.21.8.7.5 daugwylogic-TransmitMessage-LLR-005

Requirement ID: H398-LLD-GWY-FNC-2006

The function shall loop until size is greater than M\_ZERO and do the following when size is greater than M\_ZERO

1. Call UpdCrc32 with parameters (dereference of message, crc value).

##### 10.21.8.7.6 daugwylogic-TransmitMessage-LLR-006

Requirement ID: H398-LLD-GWY-FNC-2007

The function shall loop until size is greater than M\_ZERO and do the following when size is greater than M\_ZERO and message value is M\_DLE

1. Call EnQueue with parameters (reference of transmit queue and M\_DLE)
2. Call EnQueue with parameters (reference of transmit queue and M\_DLE\_PLC\_HLDR)

##### 10.21.8.7.7 daugwylogic-TransmitMessage-LLR-007

Requirement ID: H398-LLD-GWY-FNC-2008

The function shall loop until size is greater than M\_ZERO and do the following when size is greater than M\_ZERO and message value is not M\_DLE

1. Call EnQueue with parameters (reference of transmit queue and dereference of message)

##### 10.21.8.7.8 daugwylogic-TransmitMessage-LLR-008

Requirement ID: H398-LLD-GWY-FNC-2009

The function shall loop until size is greater than M\_ZERO and do the following when size is greater than M\_ZERO

1. Increment the message by one
2. Decrement the size by one

##### 10.21.8.7.9 daugwylogic-TransmitMessage-LLR-009

Requirement ID: H398-LLD-GWY-FNC-2010

The function shall do nothing when size is not greater than M\_ZERO

##### 10.21.8.7.10 daugwylogic-TransmitMessage-LLR-010

Requirement ID: H398-LLD-GWY-FNC-2011

The function shall calculate size as sizeof the crc value

##### 10.21.8.7.11 daugwylogic-TransmitMessage-LLR-011

Requirement ID: H398-LLD-GWY-FNC-2012

The function shall loop until size is greater than M\_ZERO and do the following when size is greater than M\_ZERO

1. Set transmit byte with crc value

##### 10.21.8.7.12 daugwylogic-TransmitMessage-LLR-012

Requirement ID: H398-LLD-GWY-FNC-2013

The function shall loop until size is greater than M\_ZERO and do the following when size is greater than M\_ZERO and transmit byte is equal to M\_DLE

1. Call EnQueue with parameters reference of transmit queue and M\_DLE
2. Call EnQueue with parameters reference of transmit queue and M\_DLE\_PLC\_HLDR

##### 10.21.8.7.13 daugwylogic-TransmitMessage-LLR-013

Requirement ID: H398-LLD-GWY-FNC-2014

The function shall loop until size is greater than M\_ZERO and do the following when size is greater than M\_ZERO and transmit byte is not equal to M\_DLE

1. Call EnQueue with parameters reference of transmit queue and transmit byte

##### 10.21.8.7.14 daugwylogic-TransmitMessage-LLR-014

Requirement ID: H398-LLD-GWY-FNC-2015

The function shall loop until size is greater than M\_ZERO and do the following when size is greater than M\_ZERO

1. Update crc value as bit shift of crc value to right by M\_SHIFT\_BY\_8
2. Decrement the size by one

##### 10.21.8.7.15 daugwylogic-TransmitMessage-LLR-015

Requirement ID: H398-LLD-GWY-FNC-2016

The function shall do nothing when size is not greater than M\_ZERO

##### 10.21.8.7.16 daugwylogic-TransmitMessage-LLR-016

Requirement ID: H398-LLD-GWY-FNC-2017

The function shall call EnQueue with parameters reference of transmit queue and M\_DLE

##### 10.21.8.7.17 daugwylogic-TransmitMessage-LLR-017

Requirement ID: H398-LLD-GWY-FNC-2018

The function shall call EnQueue with parameters reference of transmit queue and M\_EOT

##### 10.21.8.7.18 daugwylogic-TransmitMessage-LLR-018

Requirement ID: H398-LLD-GWY-FNC-2020

The function shall loop until items of transmit queue with index front of transmit queue is equal M\_ZERO and do the following:

1. Set front of transmit queue with (remainder of (addition of front of transmit queue and M\_ONE) by M\_MAX\_SIZE)

##### 10.21.8.7.19 daugwylogic-TransmitMessage-LLR-019

Requirement ID: H398-LLD-GWY-FNC-2021

The function shall loop until count of transmit queue is greater than M\_ZERO and do the following:

1. Call function DeQueue with parameter reference of transmit queue and set the return value to temporary variable when communication timeout is greater than or equal to M\_ZERO.
2. Call function ResetCOM with parameter comm when communication timeout is less than M\_ZERO.
3. Call function CommPutChars with parameters (temporary variable, M\_ONE, M\_TEN, comm) and set the return value to communication timeout

##### 10.21.8.7.20 daugwylogic-TransmitMessage-LLR-020

Requirement ID: H398-LLD-GWY-FNC-2022

The function shall call InitQueue with parameter reference of transmit queue

### 10.21.13 CrossChannelCmpDis

Low Level Design Details about CSU CrossChannelCmpDis will follow in the sub sections.

#### 10.21.13.1 Brief Description

This function compares offside discrete data with onside discrete and sets the data miscomparison status.

#### 10.21.13.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.21.13.3 List of global variables accessed and modified

Accessed : Xram\_data\_buffer

Modified : Mis\_cmp\_fail

#### 10.21.13.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.21.13.5 Return Value

None

#### 10.21.13.6 Other CSUs called by this CSU

TransmitMessage

#### 10.21.13.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to CrossChannelCmpDis

##### 10.21.13.7.1 daugwylogic-CrossChannelCmpDis-LLR-001

Requirement ID: H398-LLD-GWY-FNC-2091

The function shall reset index x.

##### 10.21.13.7.2 daugwylogic-CrossChannelCmpDis-LLR-002

Requirement ID: H398-LLD-GWY-FNC-2092

The function shall loop from loop index is M\_ZERO to M\_SIXTEEN minus 1 and do the following:

1. Set Cross channel discrete transmission of index (addition of M\_TWO and loop index) with M\_ZERO

##### 10.21.13.7.3 daugwylogic-CrossChannelCmpDis-LLR-003

Requirement ID: H398-LLD-GWY-FNC-2093

The function shall loop from loop index is M\_ZERO to M\_SIXTEEN minus 1, loop from counter is M\_ZERO to M\_EIGHT minus 1 and do the following:

1. Set Cross channel discrete transmission of index (addition of M\_TWO and loop index) with Cross channel discrete transmission of index (addition of two and loop index) Bitwise OR with (bit shift of disci(with index x as index) of App\_A825 of Xram\_data\_buffer to left by counter)
2. Increment the index x by one

##### 10.21.13.7.4 daugwylogic-CrossChannelCmpDis-LLR-004

Requirement ID: H398-LLD-GWY-FNC-2094

The function shall call TransmitMessage with parameters (Cross channel discrete transmission, addition of M\_CROSS\_CHANNEL\_DIS and M\_TWO, COM4)

##### 10.21.13.7.5 daugwylogic-CrossChannelCmpDis-LLR-005

Requirement ID: H398-LLD-GWY-FNC-2095

The function shall loop from loop index y is M\_TWO to addition of (M\_CROSS\_CHANNEL\_DIS and M\_TWO) minus 1 and do the following when Cross channel discrete transmission with index (index y) is not equal to Cross channel discrete reception with index (index y)

1. Set discrete mismatch compare to TRUE
2. Stop the execution from the current condition

##### 10.21.13.7.6 daugwylogic-CrossChannelCmpDis-LLR-006

Requirement ID: H398-LLD-GWY-FNC-2096

The function shall loop from loop index y is M\_TWO to addition of (M\_CROSS\_CHANNEL\_DIS and M\_TWO) minus 1 and do the following when Cross channel discrete transmission with index (index y) is equal to Cross channel discrete reception with index (index y)

1. Set discrete mismatch compare to FALSE

##### 10.21.13.7.7 daugwylogic-CrossChannelCmpDis-LLR-007

Requirement ID: H398-LLD-GWY-FNC-2097

The function shall pre increment the discrete mismatch compare counter by one when

1. discrete mismatch compare is not M\_ZERO
2. discrete mismatch compare counter is less than (division of M\_LOGIC\_RATE by M\_TEN)

##### 10.21.13.7.8 daugwylogic-CrossChannelCmpDis-LLR-008

Requirement ID: H398-LLD-GWY-FNC-2098

The function shall set discrete\_input\_miscompare of Mis\_cmp\_fail to TRUE when

1. discrete mismatch compare is not M\_ZERO
2. discrete mismatch compare counter is not less than (division of M\_LOGIC\_RATE by M\_TEN)

##### 10.21.13.7.9 daugwylogic-CrossChannelCmpDis-LLR-009

Requirement ID: H398-LLD-GWY-FNC-2099

The function shall reset discrete mismatch compare counter clear when

1. discrete mismatch compare is not M\_ZERO

##### 10.21.13.7.10 daugwylogic-CrossChannelCmpDis-LLR-010

Requirement ID: H398-LLD-GWY-FNC-2100

The function shall pre increment the discrete mismatch compare counter clear by one when

1. discrete mismatch compare is M\_ZERO
2. discrete mismatch compare counter clear is less than (division of M\_LOGIC\_RATE by M\_TEN)

##### 10.21.13.7.11 daugwylogic-CrossChannelCmpDis-LLR-011

Requirement ID: H398-LLD-GWY-FNC-2101

The function shall set discrete\_input\_miscompare of Mis\_cmp\_fail to FALSE when

1. discrete mismatch compare is M\_ZERO
2. discrete mismatch compare counter clear is not less than (division of M\_LOGIC\_RATE by M\_TEN)

##### 10.21.13.7.12 daugwylogic-CrossChannelCmpDis-LLR-012

Requirement ID: H398-LLD-GWY-FNC-2102

The function shall reset discrete mismatch compare counter when

1. discrete mismatch compare is M\_ZERO

### 10.21.16 CrossChannelComparison

Low Level Design Details about CSU CrossChannelComparison will follow in the sub sections.

#### 10.21.16.1 Brief Description

The CrossChannelComparison compares cross channel of discrete, pressure gauge and Analog input fault.

#### 10.21.16.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.21.16.3 List of global variables accessed and modified

Accessed : Xram\_data\_buffer,Crossexcitationtx, Nvm\_cmu\_tx

Modified : None

#### 10.21.16.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.21.16.5 Return Value

None

#### 10.21.16.6 Other CSUs called by this CSU

HwCopy

TransmitMessage

CrossChannelCmpDis

CrossChannelCmpPG

CrossChannelCmpMT

CrossChannelCmpRT

CrossChannelCmpAF

CrossChannelCmpFuel

#### 10.21.16.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to CrossChannelComparison

##### 10.21.16.7.1 daugwylogic-CrossChannelComparison-LLR-001

Requirement ID: H398-LLD-GWY-FNC-2111

The function shall pre-increment the Ten times counter.

##### 10.21.16.7.2 daugwylogic-CrossChannelComparison-LLR-002

Requirement ID: H398-LLD-GWY-FNC-2112

The function shall do the following when Ten times counter value is M\_ONE:

1. Call HwCopy with parameters (reference of Cross channel data with index M\_TWO,reference of tx\_cross\_channel\_rs232 of Xram\_data\_buffer, sizeof(T\_CROSS\_CHANNEL\_RS232)) to copy cross channel tranmsit data to local array.
2. Set Cross channel data with index 68 to Eleven.
3. Set Cross channel data with index 69 to Fourteen.
4. Call TransmitMessage with parameters (Cross channel data, addition of sizeof(T\_CROSS\_CHANNEL\_RS232) and M\_TWO, COM4) to transmit cross channel data.

##### 10.21.16.7.3 daugwylogic-CrossChannelComparison-LLR-003

Requirement ID: H398-LLD-GWY-FNC-2113

The function shall call CrossChannelCmpDis for discrete cross channel comparison when Ten times counter value is M\_TWO

##### 10.21.16.7.7 daugwylogic-CrossChannelComparison-LLR-009

Requirement ID: H398-LLD-GWY-FNC-5843

The function shall call CrossChannelAirbusAIF for Analog Input Fault Monitor when Ten times counter value is M\_SIX

##### 10.21.16.7.12 daugwylogic-CrossChannelComparison-LLR-014

Requirement ID: H398-LLD-GWY-FNC-5991

The function shall do the following if Ten times counter value is M\_EIGHT:

1) set cross excitation transmitter with index eight to Force channel change flag bitwise OR with bf Forced Channel Change of flags of NVM\_aircraft of Nvm\_cmu\_tx.

2) call TransmitMessage() with parameters Crossexcitationtx, M\_CROSS\_EXCITATION, COM4 for Transmit Excitation between channels.

##### 10.21.16.7.13 daugwylogic-CrossChannelComparison-LLR-006

Requirement ID: H398-LLD-GWY-FNC-2116

The function shall do nothing when Ten times counter value is other than M\_ONE,M\_TWO,M\_THREE, M\_FOUR, M\_FIVE, M\_SIX, M\_SEVEN and M\_EIGHT.

##### 10.21.16.7.14 daugwylogic-CrossChannelComparison-LLR-007

Requirement ID: H398-LLD-GWY-FNC-2117

The function shall reset M\_ZERO times counter when Ten times counter is greater than or equal to (division of M\_LOGIC\_RATE by M\_TEN)

##### 10.21.16.7.15 daugwylogic-CrossChannelComparison-LLR-016

Requirement ID: H398-LLD-GWY-FNC-7413

The function shall do the following

* Call the function CrossChannelEssential.
* Call the function CrossChannelCritical.

##### 10.21.16.7.16 daugwylogic-CrossChannelComparison-LLR-017

Requirement ID: H398-LLD-GWY-FNC-7414

The function shall do the following:

* Increment the critical timer.
* Call the function TransmitMessage with parameters (Crosschannel current transmitter, XCHANNEL\_MAX\_2 and COM4) and

Set critical timer to M\_ZERO When critical timer is greater than or equal to M\_100\_MILLISECOND.

##### 10.21.16.7.17 daugwylogic-CrossChannelComparison-LLR-018

Requirement ID: H398-LLD-GWY-FNC-7415

The function shall do the following by incrementing the essential timer and when essential valid is FALSE.

* Set critical timer to M\_ZERO
* Call the function TransmitMessage with parameters (Crosschanneless transmitter, XCHANNEL\_MAX and COM4)
* Set essential valid to TRUE

When essential timer is greater than or equal to M\_150\_MILLISECOND.

##### 10.21.16.7.18 daugwylogic-CrossChannelComparison-LLR-019

Requirement ID: H398-LLD-GWY-FNC-7416

The function shall do the following .

* Set essential timer to M\_ZERO
* Call the function TransmitMessage with parameters (Crosschanneless transmitter, XCHANNEL\_MAX and COM4)

When essential timer is greater than or equal to M\_400\_MILLISECOND.

### 10.21.17 LogicRS232

Low Level Design Details about CSU LogicRS232 will follow in the sub sections.

#### 10.21.17.1 Brief Description

The LogicRS232 copies first two bytes from receive buffer to command for identifier.

#### 10.21.17.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.21.17.3 List of global variables accessed and modified

Accessed : Xram\_data\_buffer,Crossexcitationrx

Modified : Rs232\_transmit\_data,Rs232\_present

#### 10.21.17.4 Parameter list (Input/Output)

Inputs : T\_UINT8 comm - RS232 COM ports

Outputs : None

#### 10.21.17.5 Return Value

None

#### 10.21.17.6 Other CSUs called by this CSU

HwCopy

HwMemset

InitQueue

IsFull

#### 10.21.17.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to LogicRS232

##### 10.21.17.7.1 daugwylogic-LogicRS232-LLR-001

Requirement ID: H398-LLD-GWY-FNC-2126

The function shall call HwCopy with parameters (reference of command variable, buffer of receive com port with index comm,M\_TWO)

##### 10.21.17.7.2 daugwylogic-LogicRS232-LLR-002

Requirement ID: H398-LLD-GWY-FNC-2127

The function shall do the following when command variable is M\_RS232\_IDENTIFIER\_D5:

1. Set Rs232\_transmit\_data to TRUE
2. Reset Rs232 transmit data counter
3. Call HwCopy with parameters (reference of rx\_cross\_channel\_rs232 of Xram\_data\_buffer, reference of buffer (with index M\_TWO) of receive com port with index comm, sizeof(T\_CROSS\_CHANNEL\_RS232)) when buffer (with index 68) of receive com port with index comm equal to Eleven AND buffer (with index 69) of receive com port with index comm equal to Fourteen AND App\_emode equal to APP\_NORMAL.

##### 10.21.17.7.3 daugwylogic-LogicRS232-LLR-009

Requirement ID: H398-LLD-GWY-FNC-5848

The function shall do the following when command variable is M\_RS232\_IDENTIFIER\_A2:

a) Set Rs232\_transmit\_data to TRUE

b) Set Rs232\_present to TRUE

c) Reset Rs232 transmit data counter

d) Call HwCopy with parameters (reference of Cross channel application mode received with index M\_TWO, reference of buffer (with index M\_TWO) of receive com port with index comm, M\_CROSS\_EXCITATION minus 2), when (buffer of receive com port (with index comm) with index M\_CROSS\_EXCITATION minus 2 is equal to thirteen AND buffer of receive com port (with index comm) with index M\_CROSS\_EXCITATION minus 1 is equal to twelve) is True.

##### 10.21.17.7.5 daugwylogic-LogicRS232-LLR-003

Requirement ID: H398-LLD-GWY-FNC-2128

The function shall do the following when command variable is M\_RS232\_IDENTIFIER\_B1:

1. Call HwCopy with parameters (reference of Cross channel discrete receive with index M\_TWO, reference of buffer (with index M\_TWO of) receive com port with index comm, M\_SIXTEEN)

##### 10.21.17.7.6 daugwylogic-LogicRS232-LLR-004

Requirement ID: H398-LLD-GWY-FNC-2129

The function shall do the following when command variable is M\_RS232\_IDENTIFIER\_B2:

Call HwCopy with parameters (reference of Cross channeless receive with index M\_TWO, reference of buffer

(with index M\_TWO) of receive com port with index comm, M\_FORTY)

##### 10.21.17.7.7 daugwylogic-LogicRS232-LLR-005

Requirement ID: H398-LLD-GWY-FNC-2130

The function shall do the following when command variable is M\_RS232\_IDENTIFIER\_B3:

1. Call HwCopy with parameters (reference of Cross channel current receive with index M\_TWO, reference of buffer (with index M\_TWO) of receive com port with index comm, M\_THIRTY\_SIX)

##### 10.21.17.7.9 daugwylogic-LogicRS232-LLR-012

Requirement ID: H398-LLD-GWY-FNC-5990

The function shall do the following when command variable is M\_RS232\_IDENTIFIER\_B5:

Call HwCopy with parameters (reference of Cross channel Analog interrupt Fault data receive with index M\_TWO, reference of buffer

(with index M\_TWO) of receive com port with index comm, M\_TEN).

##### 10.21.17.7.10 daugwylogic-LogicRS232-LLR-006

Requirement ID: H398-LLD-GWY-FNC-2131

The function shall do nothing when command variable is other than M\_RS232\_IDENTIFIER\_A1, M\_RS232\_IDENTIFIER\_A2, M\_RS232\_IDENTIFIER\_A3, M\_RS232\_IDENTIFIER\_B1, M\_RS232\_IDENTIFIER\_B2, M\_RS232\_IDENTIFIER\_B3, M\_RS232\_IDENTIFIER\_B4 and M\_RS232\_IDENTIFIER\_B5

##### 10.21.17.7.11 daugwylogic-LogicRS232-LLR-007

Requirement ID: H398-LLD-GWY-FNC-2132

The function shall call HwMemset with parameters (buffer of receive com port with index comm, M\_ZERO, sizeof (buffer of receive com port with index comm)).

##### 10.21.17.7.12 daugwylogic-LogicRS232-LLR-008

Requirement ID: H398-LLD-GWY-FNC-2133

The function shall do the following:

1. Call IsFull with parameter reference of receive com port with index comm
2. Call InitQueue with parameter reference of receive com port with index comm when IsFull returns other than M\_ZERO

### 10.21.19 LogicTask

Low Level Design Details about CSU LogicTask will follow in the sub sections.

#### 10.21.19.1 Brief Description

The LogicTask receives ARINC words and performs CBIT.

#### 10.21.19.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.21.19.3 List of global variables accessed and modified

Accessed : Analog\_dat\_received, Discrete\_dat\_received, Nvm\_cmu\_tx,

App\_emode,App\_powerup,App\_strap,Data\_received

Modified : App\_strap,Data\_received

#### 10.21.19.4 Parameter list (Input/Output)

Inputs : None

Outputs : void \*pdata - output data

#### 10.21.19.5 Return Value

None

#### 10.21.19.6 Other CSUs called by this CSU

OsSemPend

WdogKickWatchDog

LogicCustApp

EngData

AircraftData

ProcessAllLogics

CrossChannelComparison

ShadinFuelFlow

StrappingConfigurationLogic

#### 10.21.19.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to LogicTask

##### 10.21.19.7.1 daugwylogic-LogicTask-LLR-001

Requirement ID: H398-LLD-GWY-FNC-2142

The function shall set pdata to M\_HW\_NULL.

##### 10.21.19.7.2 daugwylogic-LogicTask-LLR-002

Requirement ID: H398-LLD-GWY-FNC-2143

The function shall loop forever and call OsSemPend with parameters (Semaphore logic task, M\_ZERO, reference of error) to wait forever for the semaphore.

##### 10.21.19.7.3 daugwylogic-LogicTask-LLR-003

Requirement ID: H398-LLD-GWY-FNC-2144

The function shall loop forever and call WdogKickWatchDog to reset watchdog.

##### 10.21.19.7.4 daugwylogic-LogicTask-LLR-004

Requirement ID: H398-LLD-GWY-FNC-2145

The function shall loop forever and set Data\_received to TRUE

when Analog\_dat\_received AND Discrete\_dat\_received returns TRUE.

##### 10.21.19.7.5 daugwylogic-LogicTask-LLR-005

Requirement ID: H398-LLD-GWY-FNC-2146

The function shall loop forever and do the following if Analog\_dat\_received AND Discrete\_dat\_received returns FALSE :

1. Increment data receive counter when Acquire\_cmd\_sent\_flag is equal to TRUE and data receive counter is less   
    than multiplication of (M\_THREE,M\_TIMER\_TEMP) returns TRUE.
2. Call RterrError function with parameters A825\_INTER\_PBIT\_ERR,M\_ONE and RterrForever when Acquire\_cmd\_sent\_flag is equal to TRUE and data receive counter is less   
    than multiplication of (M\_THREE,M\_TIMER\_TEMP) returns FALSE and Analog\_dat\_received is equal to TRUE.
3. Call RterrError function with parameters A825\_INTER\_PBIT\_ERR,M\_TWO and RterrForever  
    when Discrete\_dat\_received is equal to TRUE and data receive counter is less than multiplication of (M\_THREE,M\_TIMER\_TEMP) returns FALSE otherwise call RterrError function with parameters A825\_INTER\_PBIT\_ERR, M\_THREE and RterrForever.
4. Set Data\_received to FALSE when Acquire\_cmd\_sent\_flag is equal to TRUE

##### 10.21.19.7.6 daugwylogic-LogicTask-LLR-006

Requirement ID: H398-LLD-GWY-FNC-2147

The function shall loop forever and do the following When rev\_no(with index M\_ZERO) of Cmdownload\_lp\_cmd\_data is ‘H’. and ((App\_powerup AND (App\_emode is equal to APP\_NORMAL) AND Data\_received) OR (NOT of app init AND (App\_emode is equal to APP\_NORMAL) AND Data\_received)) returns TRUE

1. set app init to TRUE when App\_powerup is equal to TRUE
2. Call LogicCustApp for customizing application logic, if App\_powerup is equal to TRUE.
3. Call EngData to set engine data, if App\_powerup is equal to TRUE.
4. Call AircraftData to set aircraft data, if App\_powerup is equal to TRUE.
5. Call DiscreteFilter to check the discrete signals change of state, if App\_powerup is equal to TRUE.
6. Call ProcessAllLogics to process all the logics, if App\_powerup is equal to TRUE.
7. Call CrossChannelComparison for Cross Channel Comparrison, if App\_powerup is equal to TRUE.
8. Set App\_strap to TRUE, if App\_powerup is equal to TRUE.

##### 10.21.19.7.8 daugwylogic-LogicTask-LLR-008

Requirement ID: H398-LLD-GWY-FNC-6191

The function shall loop forever and do the following when rev\_no(with index M\_ZERO) of Cmdownload\_lp\_cmd\_data is ‘H’ and ((App\_powerup AND (App\_emode is equal to APP\_NORMAL) AND Data\_received) OR (NOT of app init AND (App\_emode is equal to APP\_NORMAL) AND Data\_received)) returns TRUE and App\_powerup is not equal to TRUE :

1) Call GroundState.

### 10.21.20 LogicTaskcom4

Low Level Design Details about CSU LogicTaskcom4 will follow in the sub sections.

#### 10.21.20.1 Brief Description

This function initializes logic task and process RS232 recieve data using COM4

#### 10.21.20.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.21.20.3 List of global variables accessed and modified

Accessed : Crossexcitationrx,Crossexcitationtx,Logic\_recdlu,Xram\_data\_buffer

Modified : Rs232\_transmit\_data,Xram\_data\_buffer

#### 10.21.20.4 Parameter list (Input/Output)

Inputs : None

Outputs : void \*pdata - output data

#### 10.21.20.5 Return Value

None

#### 10.21.20.6 Other CSUs called by this CSU

InitQueue

OsSemPend

WdogKickWatchDog

CommGetChars

EnQueue

MessageAvailable

LogicRS232

IsFull

HwMemset

#### 10.21.20.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to LogicTaskcom4

##### 10.21.20.7.1 daugwylogic-LogicTaskcom4-LLR-001

Requirement ID: H398-LLD-GWY-FNC-2157

The function shall set pdata to M\_HW\_NULL.

##### 10.21.20.7.2 daugwylogic-LogicTaskcom4-LLR-002

Requirement ID: H398-LLD-GWY-FNC-2158

The function shall call InitQueue with parameter (reference of receive com port with index COM4).

##### 10.21.20.7.3 daugwylogic-LogicTaskcom4-LLR-003

Requirement ID: H398-LLD-GWY-FNC-2159

The function shall loop forever and call OsSemPend with parameters (Semaphore logic com4 task, M\_ZERO, reference of error) to wait forever for the semaphore.

##### 10.21.20.7.4 daugwylogic-LogicTaskcom4-LLR-004

Requirement ID: H398-LLD-GWY-FNC-2160

The function shall loop forever and call WdogKickWatchDog to reset watchdog.

##### 10.21.20.7.5 daugwylogic-LogicTaskcom4-LLR-005

Requirement ID: H398-LLD-GWY-FNC-2161

The function shall loop forever and increment the Rs232 transmit data counter by one.

##### 10.21.20.7.6 daugwylogic-LogicTaskcom4-LLR-006

Requirement ID: H398-LLD-GWY-FNC-2162

The function shall loop forever and do the following when Rs232 transmit data counter is greater than (product of M\_THREE and M\_DATA\_TRANSFER\_RATE):

a) Set Rs232\_transmit\_data to FALSE

b) Reset Rs232\_transmit\_data\_counter

c) Call HwMemset with parameters (reference of rx\_cross\_channel\_rs232 of Xram\_data\_buffer, M\_ZERO, sizeof T\_CROSS\_CHANNEL\_RS232).

d) Set headsetStat of rx\_cross\_channel\_rs232 of Xram\_data\_buffer to MISSING.

e) Call HwMemset with parameters (reference of Cross\_channel\_disin with index M\_TWO, M\_ZERO, M\_SIXTEEN).

e) Call HwMemset with parameters (reference of Cross\_channel\_ess\_rx with index M\_TWO, M\_ZERO, M\_THIRTY\_TWO).

f) Call HwMemset with parameters (reference of Cross\_channel\_crt\_rx with index M\_TWO, M\_ZERO, M\_EIGHT).

g) Call HwMemset with parameters (reference of Cross\_channel\_aif\_rx with index M\_TWO, M\_HEX\_THIRTY\_THREE, M\_TEN).

h) Call HwMemset with parameters (reference of Cross\_excitation\_rx with index M\_TWO, M\_ZERO, M\_CROSS\_EXCITATION minus M\_TWO).

i) Call HwMemset with parameters (reference of Cross\_excitation\_tx with index M\_FOUR, M\_ZERO, M\_FOUR).

##### 10.21.20.7.7 daugwylogic-LogicTaskcom4-LLR-007

Requirement ID: H398-LLD-GWY-FNC-2163

The function shall loop forever, loop when return value of function CommGetChars with parameters (reference of receive byte, M\_ONE, M\_ZERO, COM4) is greater than or equal to M\_ZERO and do the following :

1. Call EnQueue with parameters (reference of receive com port with index COM4, receive byte)

##### 10.21.20.7.8 daugwylogic-LogicTaskcom4-LLR-008

Requirement ID: H398-LLD-GWY-FNC-2164

The function shall loop forever, loop when return value of function CommGetChars with parameters (reference of receive byte, M\_ONE, M\_ZERO, COM4) is greater than or equal to M\_ZERO and do the following when receive byte is M\_HEX\_FOUR AND receive byte2 is M\_HEX\_TEN:

1. Call MessageAvailable with parameter (reference of receive com port with index COM4)
2. Call LogicRS232 with parameter COM4

##### 10.21.20.7.9 daugwylogic-LogicTaskcom4-LLR-009

Requirement ID: H398-LLD-GWY-FNC-2165

The function shall loop forever, loop when return value of function CommGetChars with parameters (reference of receive byte, M\_ONE, M\_ZERO, COM4) is greater than or equal to M\_ZERO and do the following when (receive byte is M\_HEX\_FOUR AND receive byte2 is M\_HEX\_SIXTEEN) is FALSE and the function call IsFull with parameter (reference of receive com port with index COM4) returns other than M\_ZERO:

1. Call InitQueue with parameter (reference of receive com port with index COM4)

##### 10.21.20.7.10 daugwylogic-LogicTaskcom4-LLR-010

Requirement ID: H398-LLD-GWY-FNC-2166

The function shall loop forever, loop when return value of function CommGetChars with parameters (reference of receive byte, M\_ONE, M\_ZERO, COM4) is greater than or equal to M\_ZERO and do the following:

1. Set receive byte2 with receive byte

### 10.21.21 LogicInit

Low Level Design Details about CSU LOGICInit will follow in the sub sections.

#### 10.21.21.1 Brief Description

The LogicInit initializes the task signaling parameters and stack for run time checks and creates the ARINC 429 task

#### 10.21.21.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.21.21.3 List of global variables accessed and modified

Accessed : Logic\_task\_stk, Logic\_com4\_task\_stk

Modified : Logic\_task\_stk, Logic\_com4\_task\_stk

#### 10.21.21.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.21.21.5 Return Value

None

#### 10.21.21.6 Other CSUs called by this CSU

TbaseTaskSignaling

OsSemCreate

OsTaskCreate

LogicTask

RterrError

RterrForever

LogicTaskcom4

#### 10.21.21.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to LOGICInit

##### 10.21.21.7.1 daugwylogic-LogicInit-LLR-001

Requirement ID: H398-LLD-GWY-FNC-2177

The function shall do the following:

1. call OsSemCreate with parameter M\_ZERO and store the return value in Semaphore logic task
2. install task signaling parameters by calling TbaseTaskSignaling with parameters (M\_LOGIC\_TASK\_TICKS, Semaphore logic task)

##### 10.21.21.7.2 daugwylogic-LogicInit-LLR-002

Requirement ID: H398-LLD-GWY-FNC-2178

The function shall loop from stack counter is M\_ZERO to M\_LOGIC\_TASK\_STK\_SIZE minus 1 and initialize the stack for run time checks by setting Logic\_task\_stk of index (stack counter) with M\_CBIT\_TASK\_STK\_VAL

##### 10.21.21.7.3 daugwylogic-LogicInit-LLR-003

Requirement ID: H398-LLD-GWY-FNC-2179

The function shall do the following:

1. Call OsTaskCreate with parameters (function pointer to LogicTask, M\_HW\_NULL, reference of Logic\_task\_stk with size M\_LOGIC\_TASK\_STK\_SIZE, M\_LOGIC\_TASK\_PRIO)
2. Call RterrError with parameters (BIT\_ERR, M\_THREE, RterrForever) when OsTaskCreate returns other than M\_OS\_NO\_ERR
3. Do nothing when OsTaskCreate returns M\_OS\_NO\_ERR

##### 10.21.21.7.4 daugwylogic-LogicInit-LLR-004

Requirement ID: H398-LLD-GWY-FNC-2180

The function shall do the following:

1. call OsSemCreate with parameter M\_ZERO and store the return value in Semaphore logic com4 task
2. install task signaling parameters by calling TbaseTaskSignaling with parameters (M\_COM4\_TASK\_TICKS, Semaphore logic com4 task)

##### 10.21.21.7.5 daugwylogic-LogicInit-LLR-005

Requirement ID: H398-LLD-GWY-FNC-2181

The function shall loop from stack counter is M\_ZERO to M\_COM4\_TASK\_STK\_SIZE minus 1 and initialize the stack for run time checks by setting Logic\_com4\_task\_stk of index (stack counter) with M\_CBIT\_TASK\_STK\_VAL

##### 10.21.21.7.6 daugwylogic-LogicInit-LLR-006

Requirement ID: H398-LLD-GWY-FNC-2182

The function shall do the following:

1. Call OsTaskCreate with parameters (function pointer to LogicTaskcom4, M\_HW\_NULL, reference of Logic\_com4\_task\_stk with size M\_COM4\_TASK\_STK\_SIZE, M\_COM4\_TASK\_PRIO)
2. Call RterrError with parameters (BIT\_ERR, M\_THREE, RterrForever) when OsTaskCreate returns other than M\_OS\_NO\_ERR
3. Do nothing when OsTaskCreate returns M\_OS\_NO\_ERR

### 10.21.22 CrossChannelEssential

Low Level Design Details about CSU CrossChannelEssential will follow in the sub sections.

#### 10.21.22.1 Brief Description

This function compares offside data with onside data and sets the data miscomparison status.

#### 10.21.22.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.21.22.3 List of global variables accessed and modified

Accessed : Dau\_eng\_data

Rs232\_transmit\_data

Modified : Mis\_cmp\_fail

#### 10.21.22.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.21.22.5 Return Value

None

#### 10.21.22.6 Other CSUs called by this CSU

HwCopy

#### 10.21.22.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to CrossChannelEssential

##### 10.21.22.7.1 daugwylogic-CrossChannelEssential-LLR-001

Requirement ID: H398-LLD-GWY-FNC-7313

The function shall perform the following:

1. Set Threshold with index EOP\_1 to M\_EOP\_THRESHOLD
2. Set Threshold with index EOP\_2 to M\_EOP\_THRESHOLD
3. Set Threshold with index EOT\_1 to M\_EOT\_THRESHOLD
4. Set Threshold with index EOT\_2 to M\_EOT\_THRESHOLD
5. Set Threshold with index FP\_1 to M\_FP\_THRESHOLD
6. Set Threshold with index FP\_2 to M\_FP\_THRESHOLD
7. Set Threshold with index HP\_1 to M\_HP\_THRESHOLD
8. Set Threshold with index HP\_2 to M\_HP\_THRESHOLD
9. Set Threshold with index MGBOP\_T to M\_MGBOP\_THRESHOLD
10. Set Threshold with index MGBT\_T to M\_MGBT\_THRESHOLD

##### 10.21.22.7.2 daugwylogic-CrossChannelEssential-LLR-002

Requirement ID: H398-LLD-GWY-FNC-7314

The function shall perform the following:

1. Call the function HwCopy with parameters (address of Crosschanneless transmitter with index XCHANNEL\_EOP\_1, address of Dau\_eng\_data with index EOP1 and M\_FOUR).
2. Call the function HwCopy with parameters (address of Crosschanneless transmitter with index XCHANNEL\_EOP\_2, address of Dau\_eng\_data with index EOP2 and M\_FOUR)
3. Call the function HwCopy with parameters (address of Crosschanneless transmitter with index XCHANNEL\_EOT\_1, address of Dau\_eng\_data with index EOT1 and M\_FOUR)
4. Call the function HwCopy with parameters (address of Crosschanneless transmitter with index XCHANNEL\_EOT\_2, address of Dau\_eng\_data with index EOT2 and M\_FOUR)
5. Call the function HwCopy with parameters (address of Crosschanneless transmitter with index XCHANNEL\_FP\_1, address of Dau\_eng\_data with index FP1\_SIG and M\_FOUR)
6. Call the function HwCopy with parameters (address of Crosschanneless transmitter with index XCHANNEL\_FP\_2, address of Dau\_eng\_data with index FP2\_SIG and M\_FOUR)
7. Call the function HwCopy with parameters (address of Crosschanneless transmitter with index XCHANNEL\_HP\_1, address of Dau\_eng\_data with index LHP\_SIG and M\_FOUR)
8. Call the function HwCopy with parameters (address of Crosschanneless transmitter with index XCHANNEL\_HP\_2, address of Dau\_eng\_data with index RHP\_SIG and M\_FOUR)
9. Call the function HwCopy with parameters (address of Crosschanneless transmitter with index XCHANNEL\_MGBOP, address of Dau\_eng\_data with index MGBOP\_SIG and M\_FOUR)
10. Call the function HwCopy with parameters (address of Crosschanneless transmitter with index XCHANNEL\_MGBT, address of Dau\_eng\_data with index MGBT and M\_FOUR)

##### 10.21.22.7.3 daugwylogic-CrossChannelEssential-LLR-003

Requirement ID: H398-LLD-GWY-FNC-7315

The function shall loop XCHANNEL\_EOP\_1 to XCHANNEL\_MAX minus one by setting countz to countz plus four and perform the following:

1. Call the function HwCopy with parameters (address of temporary transmitter, address of Crosschanneless transmitter with index countz and M\_FOUR).
2. Call the function HwCopy with parameters (address of temporary transmitter, address of Crosschanneless recevier with index countz and M\_FOUR).

##### 10.21.22.7.4 daugwylogic-CrossChannelEssential-LLR-004

Requirement ID: H398-LLD-GWY-FNC-7316

The function shall loop XCHANNEL\_EOP\_1 to XCHANNEL\_MAX minus one by setting count to count plus four and perform the following:

1. Set miscompare of index count to TRUE when (NOT (M\_FP\_COMPARE with parameters(temporary transmitter and temporary receiver))) AND (((difference of temporary transmitter and temporary receiver) greater than Threshold level with index[(ratio of (difference of count and M\_TWO) toM\_FOUR])

OR ((difference of temporary receiver and temporary transmitter) is greater than Threshold level with index[(ratio of (difference of count and M\_TWO) toM\_FOUR]),otherwise Set miscompare of index count to FALSE.

2. Increment count.

##### 10.21.22.7.5 daugwylogic-CrossChannelEssential-LLR-005

Requirement ID: H398-LLD-GWY-FNC-7317

The function shall loop M\_ZERO to M\_TEN minus one by incrementing count and perform the following:

1. Pre-Increment miscompare counter with index count when miscompare counter with index count is less than the product of (M\_LOGIC\_RATE and M\_TWO),otherwise set analog\_input\_miscompare with index count of Mis\_cmp\_fail to TRUE.
2. Set miscompare counter clear of index count to M\_ZERO.

When M\_ZERO is not equal to miscompare with index count.

##### 10.21.22.7.6 daugwylogic-CrossChannelEssential-LLR-006

Requirement ID: H398-LLD-GWY-FNC-7318

The function shall loop M\_ZERO to M\_TEN minus one by incrementing count and perform the following:

1. Pre-Increment miscompare counter clear with index count when miscompare counter clear with index count is less than M\_LOGIC\_RATE,otherwise set analog\_input\_miscompare with index count of Mis\_cmp\_fail toFALSE.
2. Set miscompare counter clear of index count to M\_ZERO.

When M\_ZERO is equal to miscompare with index count.

##### 10.21.22.7.7 daugwylogic-CrossChannelEssential-LLR-007

Requirement ID: H398-LLD-GWY-FNC-7319

The function shall perform the following when Rs232\_transmit\_data is FALSE:

* Set analog\_input\_miscompare with index count of Mis\_cmp\_fail to FALSE when the function is looping from M\_ZERO to M\_TEN minus one.

##### 10.21.22.7.8 daugwylogic-CrossChannelEssential-LLR-008

Requirement ID: H398-LLD-GWY-FNC-7320

The function shall loop M\_ZERO to M\_TEN minus one by incrementing count and perform the following:

* Set Essential Miscompare Fail to TRUE when analog\_input\_miscompare with index count of Mis\_cmp\_fail to TRUE and breaks the condition

otherwise set Essential Miscompare Fail to FALSE.

### 10.21.23 CrossChannelCritical

Low Level Design Details about CSU CrossChannelCritical will follow in the sub sections.

#### 10.21.23.1 Brief Description

This function compares offside data with onside data and sets the data miscomparison status.

#### 10.21.23.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.21.23.3 List of global variables accessed and modified

Accessed                :      Dau\_eng\_data

                                       Rs232\_transmit\_data

Essential\_Miscompare\_Fail

CriticalMiscompareFail

Deltang\_sigvalue

Modified                  :      Mis\_cmp\_fail

               AnalogMiscompareFail

CriticalMiscompareFail

#### 10.21.23.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.21.23.5 Return Value

None

#### 10.21.23.6 Other CSUs called by this CSU

HwCopy

#### 10.21.23.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to CrossChannelCritical

##### 10.21.23.7.1 daugwylogic-CrossChannelCritical-LLR-001

Requirement ID: H398-LLD-GWY-FNC-7425

The function shall perform the following:

1. Set Threshold with index TOTAL\_TRQ to M\_TRQ\_THRESHOLD
2. Set Threshold with index TRQ\_1 to M\_TRQ\_THRESHOLD
3. Set Threshold with index TRQ\_2 to M\_TRQ\_THRESHOLD
4. Set Threshold with index T4\_1 to M\_T4\_THRESHOLD
5. Set Threshold with index T4\_2 to M\_T4\_THRESHOLD
6. Set Threshold with index NG\_1 to M\_NG\_THRESHOLD
7. Set Threshold with index NG\_2 to M\_NG\_THRESHOLD
8. Set Threshold with index DELTA\_NG\_1 to M\_DELTANG\_THRESHOLD
9. Set Threshold with index DELTA\_NG\_2 to M\_DELTANG\_THRESHOLD

##### 10.21.23.7.2 daugwylogic-CrossChannelCritical-LLR-002

Requirement ID: H398-LLD-GWY-FNC-7426

The function shall perform the following:

1. Call HwCopy with parameters (reference of Cross channel current transmitter with index XCHANNEL\_TOTAL\_TRQ,reference of computedTorque12, M\_FOUR).
2. Call HwCopy with parameters (reference of Cross channel current transmitter with index XCHANNEL\_TRQ\_1,reference of Dau\_eng\_data with index TRQ1, M\_FOUR).
3. Call HwCopy with parameters (reference of Cross channel current transmitter with index XCHANNEL\_TRQ\_2,reference of Dau\_eng\_data with index TRQ2, M\_FOUR).
4. Call HwCopy with parameters (reference of Cross channel current transmitter with index XCHANNEL\_T4\_1,reference of Dau\_eng\_data with index TOT1, M\_FOUR).
5. Call HwCopy with parameters (reference of Cross channel current transmitter with index XCHANNEL\_T4\_2,reference of Dau\_eng\_data with index TOT2, M\_FOUR).
6. Call HwCopy with parameters (reference of Cross channel current transmitter with index XCHANNEL\_NG\_1,reference of Dau\_eng\_data with index TACH\_NG1, M\_FOUR).
7. Call HwCopy with parameters (reference of Cross channel current transmitter with index XCHANNEL\_NG\_2,reference of Dau\_eng\_data with index TACH\_NG2, M\_FOUR).
8. Call HwCopy with parameters (reference of Cross channel current transmitter with index XCHANNEL\_DELTA\_NG\_1,reference of Deltang\_sigvalue with index ZERO, M\_FOUR).
9. Call HwCopy with parameters (reference of Cross channel current transmitter with index XCHANNEL\_DELTA\_NG\_2,reference of Deltang\_sigvalue with index ONE, M\_FOUR).

##### 10.21.23.7.3 daugwylogic-CrossChannelCritical-LLR-003

Requirement ID: H398-LLD-GWY-FNC-7427

The function shall loop XCHANNEL\_TOTAL\_TRQ to XCHANNEL\_MAX\_2 minus one by setting countz to countz plus four and perform the following:

1. Call the function HwCopy with parameters (address of temporary transmitter, address of Crosschannel current transmitter with index countz and M\_FOUR).
2. Call the function HwCopy with parameters (address of temporary transmitter, address of Crosschannel current recevier with index countz and M\_FOUR).

##### 10.21.23.7.4 daugwylogic-CrossChannelCritical-LLR-004

Requirement ID: H398-LLD-GWY-FNC-7428

The function shall loop XCHANNEL\_TOTAL\_TRQ to XCHANNEL\_MAX\_2 minus one by setting countz to countz plus four and perform the following:

1. Set miscompare of index count to TRUE when (NOT (M\_FP\_COMPARE with parameters(temporary transmitter and temporary receiver))) AND (((difference of temporary transmitter and temporary receiver) greater than Threshold level with index[(ratio of (difference of count and M\_TWO) toM\_FOUR]) OR ((difference of temporary transmitter and temporary receiver) is greater than Threshold level with index[(ratio of (difference of count and M\_TWO) toM\_FOUR]),otherwise Set miscompare of index count to FALSE.
2. Increment count.

##### 10.21.23.7.5 daugwylogic-CrossChannelCritical-LLR-005

Requirement ID: H398-LLD-GWY-FNC-7429

The function shall loop M\_ZERO to M\_NINE minus one by incrementing count and perform the following:

1. Pre-Increment miscompare counter with index count when miscompare counter with index count is less than the ratio of (M\_LOGIC\_RATE and M\_TWO),

otherwise set analog\_input\_miscompare with index (count plus M\_TEN) of Mis\_cmp\_fail

to TRUE.

2. Set miscompare counter clear of index count to M\_ZERO.

When M\_ZERO is not equal to miscompare with index count.

##### 10.21.23.7.6 daugwylogic-CrossChannelCritical-LLR-006

Requirement ID: H398-LLD-GWY-FNC-7430

The function shall loop M\_ZERO to M\_TEN minus one by incrementing count and perform the following:

1. Pre-Increment miscompare counter with index count when miscompare counter with index count is less than M\_LOGIC\_RATE,otherwise set analog\_input\_miscompare with index (count plus M\_TEN )of Mis\_cmp\_fail to FALSE.
2. Set miscompare counter clear of index count to M\_ZERO.

When M\_ZERO is equal to miscompare with index count.

##### 10.21.23.7.7 daugwylogic-CrossChannelCritical-LLR-007

Requirement ID: H398-LLD-GWY-FNC-7431

The function shall perform the following when Rs232\_transmit\_data is FALSE:

* Set analog\_input\_miscompare with index count of Mis\_cmp\_fail to FALSE when the function is looping from M\_TEN to M\_NINETEEN minus one.

##### 10.21.23.7.8 daugwylogic-CrossChannelCritical-LLR-008

Requirement ID: H398-LLD-GWY-FNC-7432

The function shall loop M\_TEN to M\_NINETEEN minus one by incrementing count and perform the following:

1. Set critical Miscompare Fail to TRUE when analog\_input\_miscompare with index count of Mis\_cmp\_fail to TRUE and breaks the statement,otherwise set critical Miscompare Fail to FALSE.

##### 10.21.23.7.9 daugwylogic-CrossChannelCritical-LLR-009

Requirement ID: H398-LLD-GWY-FNC-7433

The function shall perform the following:

1. Set AnalogMiscompareFail to TRUE

when (EssentialMiscompareFail is TRUE )OR (CriticalMiscompareFail is TRUE),

otherwise Set AnalogMiscompareFail to FALSE.

### 10.21.24 CrossChannelAirbusAIF

Low Level Design Details about CSU CrossChannelAirbusAIF will follow in the sub sections.

#### 10.21.24.1 Brief Description

This function compares Analog Input Fault Monitor data and sets the data miscomparison status.

#### 10.21.24.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.21.24.3 List of global variables accessed and modified

Accessed : Dau\_eng\_status

Modified : Mis\_cmp\_fail

#### 10.21.24.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.21.24.5 Return Value

None

#### 10.21.24.6 Other CSUs called by this CSU

TransmitMessage

#### 10.21.24.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to CrossChannelAirbusAIF

##### 10.21.24.7.1 daugwylogic-CrossChannelAirbusAIF-LLR-001

Requirement ID: H398-LLD-GWY-FNC-7442

The function shall perform the following:

1. Set Crosschannel analog input fault transmitter with index M\_TWO to BITWISE OR ((BITWISE AND of (Dau\_eng\_status of index EOP2 LEFT SHIFT by M\_FOUR) and M\_BYTE\_MSB\_MASK)

BITWISE AND of (Dau\_eng\_status of index EOP1 and M\_BYTE\_LSB\_MASK))

2. Set Crosschannel analog input fault transmitter with index M\_THREE to BITWISE OR ((BITWISE AND of (Dau\_eng\_status of index EOT2 LEFT SHIFT by M\_FOUR) and M\_BYTE\_MSB\_MASK) BITWISE AND of (Dau\_eng\_status of index EOT1 and M\_BYTE\_LSB\_MASK)).

3. Set Crosschannel analog input fault transmitter with index M\_FOUR to

BITWISE AND of (Dau\_eng\_status of index MGBT and M\_BYTE\_LSB\_MASK)).

4. Set Crosschannel analog input fault transmitter with index M\_FIVE to BITWISE OR ((BITWISE AND of (Dau\_eng\_status of index FP2\_SIG LEFT SHIFT by M\_FOUR) and M\_BYTE\_MSB\_MASK) BITWISE AND of (Dau\_eng\_status of index FP1\_SIG and M\_BYTE\_LSB\_MASK)).

5. Set Crosschannel analog input fault transmitter with index M\_SIX to BITWISE OR ((BITWISE AND of (Dau\_eng\_status of index RHP\_SIG LEFT SHIFT by M\_FOUR) and M\_BYTE\_MSB\_MASK) BITWISE AND of (Dau\_eng\_status of index LHP\_SIG and M\_BYTE\_LSB\_MASK)).

6. Set Crosschannel analog input fault transmitter with index M\_SEVEN to BITWISE AND of (Dau\_eng\_status of index MGBOP\_SIG and M\_BYTE\_LSB\_MASK)).

7. Set Crosschannel analog input fault transmitter with index M\_EIGHT to BITWISE OR ((BITWISE AND of (Dau\_eng\_status of index TRQ2 LEFT SHIFT by M\_FOUR) and M\_BYTE\_MSB\_MASK) BITWISE AND of (Dau\_eng\_status of index TRQ1 and M\_BYTE\_LSB\_MASK)).

8. Set Crosschannel analog input fault transmitter with index M\_NINE to BITWISE OR ((BITWISE AND of (Dau\_eng\_status of index TOT2 LEFT SHIFT by M\_FOUR) and M\_BYTE\_MSB\_MASK)

BITWISE AND of (Dau\_eng\_status of index TOT1 and M\_BYTE\_LSB\_MASK)).

9. Set Crosschannel analog input fault transmitter with index M\_TEN to BITWISE OR ((BITWISE AND of (Dau\_eng\_status of index TACH\_NG2 LEFT SHIFT by M\_FOUR) and M\_BYTE\_MSB\_MASK) BITWISE AND of (Dau\_eng\_status of index TACH\_NG1 and M\_BYTE\_LSB\_MASK)).

10. Set Crosschannel analog input fault transmitter with index M\_ELEVEN to BITWISE OR ((BITWISE AND of (Dau\_eng\_status of index DELTA\_NG2 LEFT SHIFT by M\_FOUR) and M\_BYTE\_MSB\_MASK) BITWISE AND of (Dau\_eng\_status of index DELTA\_NG1 and M\_BYTE\_LSB\_MASK)).

11. Call the function TransmitMessage with parameters (Crosschannel analog interrupt fault transmitter, M\_TEN plus M\_TWO and COM4)

##### 10.21.24.7.2 daugwylogic-CrossChannelAirbusAIF-LLR-002

Requirement ID: H398-LLD-GWY-FNC-7443

The function shall loop M\_TWO to M\_TEN plus M\_TWO by incrementing the count each time and perform the following:

* Set analog fault miscompare to TRUE when Crosschannel analog interrupt fault transmitter of index count is not equal to Crosschannel analog interrupt fault receiver of index count and breaks the statement,otherwise Set analog fault miscompare to FALSE.

##### 10.21.24.7.3 daugwylogic-CrossChannelAirbusAIF-LLR-003

Requirement ID: H398-LLD-GWY-FNC-7444

The function perform the following when analog fault miscompare is TRUE:

* Preincrement the analog fault miscompare counter when analog fault miscompare counter is less than (division of M\_LOGIC\_RATE by M\_TEN),otherwise set airbus\_analog\_fault of Mis\_cmp\_fail to TRUE.
* Set analog fault miscompare counter clear to M\_ZERO

##### 10.21.24.7.4 daugwylogic-CrossChannelAirbusAIF-LLR-004

Requirement ID: H398-LLD-GWY-FNC-7445

The function perform the following when analog fault miscompare is not equal to TRUE:

* Preincrement the analog fault miscompare counter clear when analog fault miscompare counter clear is less than (division of M\_LOGIC\_RATE by M\_TEN),otherwise set airbus\_analog\_fault of Mis\_cmp\_fail to FALSE.
* Set analog fault miscompare counter to M\_ZERO.

### 10.21.25 DiscreteFilter

Low Level Design Details about CSU DiscreteFilter will follow in the sub sections.

#### 10.21.25.1 Brief Description

This function checks the discrete signals change of state holds for 100ms and more.

#### 10.21.25.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.21.25.3 List of global variables accessed and modified

Accessed : Xram\_data\_buffer

Modified : Xram\_data\_buffer

#### 10.21.25.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.21.25.5 Return Value

None

#### 10.21.25.6 Other CSUs called by this CSU

None

#### 10.21.25.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to DiscreteFilter

##### 10.21.25.7.1 daugwylogic-DiscreteFilter-LLR-001

Requirement ID: H398-LLD-GWY-FNC-7484

The function shall loops through M\_ZERO to M\_ONE\_TWENTY\_EIGHT and performs the following:

* Set App old value with index loop counter to App new value with index loop counter.
* Set App new value with index loop counter to temp disci with index loop counter of App\_A825 of Xram\_data\_buffer

##### 10.21.25.7.2 daugwylogic-DiscreteFilter-LLR-002

Requirement ID: H398-LLD-GWY-FNC-7485

The function shall loops through M\_ZERO to M\_ONE\_TWENTY\_EIGHT and performs the following:

* Set disci with loop counter of App A825 of Xram data buffer is equal to app new value with loop counter when loop counter is equal to DI\_2\_OEI30\_ENG1 OR loop counter is equal to DI\_3\_OEI30\_ENG2 OR loop counter is equal to DI\_4\_OEI2\_ENG1 OR loop counter is equal to DI\_5\_OEI2\_ENG2 OR loop counter is equal to DI\_11\_ENG\_GOV1 OR loop counter is equal to DI\_12\_ENG\_GOV2 ,otherwise
* Set App filter timer with index loop counter to M\_ZERO when app old value with index loop counter is not equal to app new value with index loop counter.

##### 10.21.25.7.3 daugwylogic-DiscreteFilter-LLR-003

Requirement ID: H398-LLD-GWY-FNC-7486

The function shall loops through M\_ZERO to M\_ONE\_TWENTY\_EIGHT and performs the following:

* Set disci with index loop counter of App\_A825 of Xram\_data\_buffer to App new value with index loop counter when App filter timer with index loop counter is equal to M\_100\_MILLISECOND.

Otherwise,

Set App filter timer with index loop counter to App filter timer with index loop counter added with M\_ONE

## 10.22 daugwylogic2

The daugwylogic2 file defines the implementation of A429 label reception and transmission.

### 10.22.2 ProcessAllLogics

Low Level Design Details about CSU ProcessAllLogics will follow in the sub sections.

#### 10.22.2.1 Brief Description

This function processes all the logic functions.

#### 10.22.2.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.22.2.3 List of global variables accessed and modified

Accessed : Init\_param  
 Two\_sequence\_done

Modified : Vnc\_sdi

Init\_param  
 Two\_sequence\_done

#### 10.22.2.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.22.2.5 Return Value

None

#### 10.22.2.6 Other CSUs called by this CSU

TxProcessFlightDirectory

ProcessLabel0306

ProcessLabel0307

FuelQuantitySelect

FuelPollingProcess

FuelSantiyCheckLogic

DiscreteOutputCtrlProcessing

AudioOutputProcessing

ProgrammableCASMessage

TxBNRLabelsToCMU

TxDDWLabelsToCMU

DuPreceedanceLogic

TxProcessLabel

AircraftOperatingModes

DiscreteDataWords

SpecialExceedanceCAS

GpioReadInputDataBit

#### 10.22.2.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to ProcessAllLogics

##### 10.22.2.7.1 daugwylogic2-ProcessAllLogics-LLR-001

Requirement ID: H398-LLD-GWY-FNC-2214

The function shall do the following:

1. Set Vnc\_sdi to ((return value of function ‘GpioReadInputDataBit’(M\_HW\_GET\_RCI) with parameters M\_GPIOC and M\_GPIOC\_SYS\_SEL) added to M\_ONE).

2. Call the function DiscreteSignals.

3. Call the function EngineStateLogic.

4. Call the function TxProcessAirbusLabel.

5.Increment Two\_counter\_sequence by one when Two\_counter\_sequence is less than M\_TWO otherwise perform as follows:

* Set Two\_sequence\_done to TRUE
* Call TxBNRLabelsToCMU with parameter Tx\_label when App\_switching is not equal to TRUE and modulus of Transmit temparory flag with M\_TWO is equal to M\_ZERO.
* Call 'TxDDWLabelsToCMU' with parameter Transmit label when App\_switching is not equal to TRUE and modulus of Transmit temparory flag with M\_TWO is not equal to M\_ZERO.
* Increment Transmit label by one when modulus of Transmit temparory flag with M\_TWO is not equal to M\_ZERO.
* Increment Transmit temparory flag by one when Transmit temparory flag is less than or equal to M\_TWO, otherwise set Transmit label and Transmit temparory flag to M\_ZERO.

### 10.22.5 DUDiscreteInputsStatus

Low Level Design Details about CSU DUDiscreteInputsStatus will follow in the sub sections.

#### 10.22.5.1 Brief Description

The DUDiscreteInputsStatus function processes the Discrete input status of DU.

#### 10.22.5.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.22.5.3 List of global variables accessed and modified

Accessed : Rx\_a429

Dupreceedance

Vnc\_sdi

Modified : None

#### 10.22.5.4 Parameter list (Input/Output)

Inputs : T\_A429\_INPUT\_LABELS\_ACA label- A429 label

T\_UINT8 bit- bit range

T\_DU\_PROIRITY priorityOrder- priority order

T\_UINT8 duNumber- display unit number

Outputs : None

#### 10.22.5.5 Return Value

T\_BOOL- returns DU output

#### 10.22.5.6 Other CSUs called by this CSU

None

#### 10.22.5.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to DUDiscreteInputsStatus.

##### 10.22.5.7.1 daugwylogic2-DUDiscreteInputsStatus-LLR-001

Requirement ID: H398-LLD-GWY-FNC-2251

The function shall perform the following,

When priorityOrder is equal to ALL\_DU and loop from display index equal to M\_ZERO till M\_MAX\_DISPLAY minus 1:

set output variable to output variable bitwise OR with (data of Rx\_a429 with indexes ((du channel with index display index) and (label)) right shifted by bit bitwise AND with M\_BIT\_ONE\_MASK) when status of Rx\_a429 with indices ((du channel with index display index) and (label)) is equal to OK.

##### 10.22.5.7.7 daugwylogic2-DUDiscreteInputsStatus-LLR-006

Requirement ID: H398-LLD-GWY-FNC-2256

The function shall do nothing when priorityOrder is other than ALL\_DU, DU1\_DU2, DU3\_DU4, DU1\_DU4, DU3\_SINGLE and DU\_HIGH\_PRIORITY.

##### 10.22.5.7.8 daugwylogic2-DUDiscreteInputsStatus-LLR-007

Requirement ID: H398-LLD-GWY-FNC-2257

The function shall return output variable.

### 10.22.11 CalcSysChecksum

Low Level Design Details about CSU CalcSysChecksum will follow in the sub sections.

#### 10.22.11.1 Brief Description

The CalcSysChecksum function computes system checksum.

#### 10.22.11.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.22.11.3 List of global variables accessed and modified

Accessed : App\_analogappcrc

App\_discreteappcrc

App\_cmuappcrc

App\_analogconfigcrc

App\_discreteconfigcrc

App\_analogbootcrc

App\_discretebootcrc

App\_cmubootcrc

App\_analogbootconfigcrc

App\_discretebootconfigcrc

App\_cmubootconfigcrc

Modified : None

#### 10.22.11.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.22.11.5 Return Value

T\_UINT32 - return computed CRC value

#### 10.22.11.6 Other CSUs called by this CSU

None

#### 10.22.11.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to CalcSysChecksum.

##### 10.22.11.7.1 daugwylogic2-CalcSysChecksum-LLR-001

Requirement ID: H398-LLD-GWY-FNC-2679

The function shall

1. Set CRC for Gateway Flight Application with actual CRC ‘M\_MEMMAP\_FLIGHT\_CRC\_ADDR’.
2. Set CRC for Gateway acd with actual CRC ‘M\_MEMMAP\_DATA\_1\_CRC\_ADDR’.
3. Set CRC for Gateway boot with actual CRC ‘M\_MEMMAP\_BOOT\_CRC\_ADDR’.
4. Set CRC for Gateway boot config with actual CRC ‘M\_MEMMAP\_BOOT\_CONFIG\_CRC\_ADDR’.
5. Set checksum variable to CRC for Gateway Flight Application added to App\_analogappcrc added to App\_discreteappcrc added to App\_cmuappcrc.
6. Set checksum variable to checksum variable added to CRC for Gateway acd added to App\_analogconfigcrc added to App\_discreteconfigcrc.
7. Set checksum variable to checksum variable added to CRC for Gateway boot added to App\_analogbootcrc added to App\_discretebootcrc added to App\_cmubootcrc.
8. Set checksum variable to checksum variable added to CRC for Gateway boot config added to App\_analogbootconfigcrc added to App\_discretebootconfigcrc added to App\_cmubootconfigcrc.
9. Set checksum variable to (Negation of checksum variable) added to M\_ONE.
10. Return checksum variable.

## 10.25 daugwylookup

The daugwylookup CSC contains implementation of lookup table routines.

### 10.25.1 LookUpTableLookUp

Low Level Design Details about CSU LookUpTableLookUp will follow in the sub sections.

#### 10.25.1.1 Brief Description

The LookUpTableLookUp function calculates sensor unit value for the corresponding ADC reading.

#### 10.25.1.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.25.1.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.25.1.4 Parameter list (Input/Output)

Inputs: T\_LOOKUP\_TABLE \*table – pointer to lookup table.

T\_SINT32 ind\_var -Independent Variable to find Dependent value.

T\_LOOKUP\_STAT \*stat - status of the operation.

Outputs: T\_LOOKUP\_STAT \*stat - status of the operation.

#### 10.25.1.5 Return Value

T\_SINT32 - Calculated Sensor unit value for the corresponding ADC reading.

#### 10.25.1.6 Other CSUs called by this CSU

None

#### 10.25.1.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to LookupTableLookup

##### 10.25.1.7.1 daugwylookup-LookupTableLookup-LLR-001

Requirement ID: H398-LLD-GWY-FNC-3667

The function shall return M\_ZERO when pi32\_x of table is M\_HW\_NULL OR pi32\_y of table is M\_HW\_NULL OR stat is M\_HW\_NULL

##### 10.25.1.7.2 daugwylookup-LookupTableLookup-LLR-002

Requirement ID: H398-LLD-GWY-FNC-3668

The function shall do the following when pi32\_x of table, pi32\_y of table and stat are not M\_HW\_NULL

1. Set number of points to post increment of array x value which is associated with pi32\_x of table
2. Increment array y
3. Set last x to array x with index (subtraction of number of points by M\_ONE)
4. Store array x value to temporary register

##### 10.25.1.7.3 daugwylookup-LookupTableLookup-LLR-003

Requirement ID: H398-LLD-GWY-FNC-3669

The function shall do the following

1. Set error status with LS\_OUTSIDE
2. Set stat with error status
3. Return array y (i.e lowest Sensor unit value from the sensor table)

when

- pi32\_x of table, pi32\_y of table and stat are not M\_HW\_NULL

- temporary register value is less than or equal to last x (First value in the ADC table less than or equal to last value in the ADC table)

- Actual ADC reading for the sensor is less than temporary register

##### 10.25.1.7.4 daugwylookup-LookupTableLookup-LLR-004

Requirement ID: H398-LLD-GWY-FNC-3670

The function shall do the following

1. Set array y with addition of array y and (subtraction of number of points by M\_ONE)
2. Set error status with LS\_OUTSIDE when Actual ADC reading for the sensor is greater than last x
3. Do nothing when Actual ADC reading for the sensor is not greater than last x
4. Set stat with error status
5. Return array y

when

- pi32\_x of table, pi32\_y of table and stat are not M\_HW\_NULL

- temporary register value is less than or equal to last x (First value in the ADC table less than or equal to last value in the ADC table)

- Actual ADC reading for the sensor is greater than or equal to temporary register

- Actual ADC reading for the sensor is greater than or equal to last x

##### 10.25.1.7.5 daugwylookup-LookupTableLookup-LLR-005

Requirement ID: H398-LLD-GWY-FNC-3671

The function shall loop till the condition (NOT((Actual ADC reading for the sensor is less than temporary register) AND (Actual ADC reading for the sensor is greater than or equal to dereference of (subtraction of array x by one)))) is TRUE and do the following

1. Right shift number of points by one when number of points is greater than M\_ONE
2. Do nothing when number of points is not greater than M\_ONE

when

- pi32\_x of table, pi32\_y of table and stat are not M\_HW\_NULL

- temporary register value is less than or equal to last x (First value in the ADC table less than or equal to last value in the ADC table)

- Actual ADC reading for the sensor is greater than or equal to temporary register

- Actual ADC reading for the sensor is less than last x

##### 10.25.1.7.6 daugwylookup-LookupTableLookup-LLR-006

Requirement ID: H398-LLD-GWY-FNC-3672

The function shall loop till the condition (NOT((Actual ADC reading for the sensor is less than temporary register) AND (Actual ADC reading for the sensor is greater than or equal to dereference of (subtraction of array x by one)))) is TRUE and do the following

1. Increment array x by number of points and increment array y by number of points when temporary register is less than or equal to Actual ADC reading for the sensor
2. Decrement array x by number of points and decrement array y by number of points when temporary register is greater than Actual ADC reading for the sensor

when

- pi32\_x of table, pi32\_y of table and stat are not M\_HW\_NULL

- temporary register value is less than or equal to last x (First value in the ADC table less than or equal to last value in the ADC table)

- Actual ADC reading for the sensor is greater than or equal to temporary register

- Actual ADC reading for the sensor is less than last x

##### 10.25.1.7.7 daugwylookup-LookupTableLookup-LLR-007

Requirement ID: H398-LLD-GWY-FNC-3673

The function shall loop till the condition (NOT((Actual ADC reading for the sensor is less than temporary register) AND (Actual ADC reading for the sensor is greater than or equal to dereference of (subtraction of array x by M\_ONE)))) is TRUE and set temporary register with array x

when

- pi32\_x of table, pi32\_y of table and stat are not M\_HW\_NULL

- temporary register value is less than or equal to last x (First value in the ADC table less than or equal to last value in the ADC table)

- Actual ADC reading for the sensor is greater than or equal to temporary register

- Actual ADC reading for the sensor is less than last x

##### 10.25.1.7.8 daugwylookup-LookupTableLookup-LLR-008

Requirement ID: H398-LLD-GWY-FNC-3674

The function shall do the following

1. Set error status with LS\_OUTSIDE
2. Set stat value with error status
3. Return array y value

when

- pi32\_x of table, pi32\_y of table and stat are not M\_HW\_NULL

- temporary register value is greater than last x (First value in the ADC table greater than last value in the ADC table)

- Actual ADC reading for the sensor is greater than temporary register

##### 10.25.1.7.9 daugwylookup-LookupTableLookup-LLR-009

Requirement ID: H398-LLD-GWY-FNC-3675

The function shall set array y with (array y added to number of points subtracted by M\_ONE)

when

- pi32\_x of table, pi32\_y of table and stat are not M\_HW\_NULL

- temporary register value is greater than last x (First value in the ADC table is greater than last value in the ADC table)

- Actual ADC reading for the sensor is not greater than temporary register

- Actual ADC reading for the sensor is less than or equal to last x

##### 10.25.1.7.10 daugwylookup-LookupTableLookup-LLR-010

Requirement ID: H398-LLD-GWY-FNC-3676

The function shall do the following:

1. Error status with LS\_OUTSIDE when Actual ADC reading for the sensor is less than last x
2. Do nothing when Actual ADC reading for the sensor is equal to last x

when

- pi32\_x of table, pi32\_y of table and stat are not M\_HW\_NULL

- temporary register value is greater than last x (First value in the ADC table is greater than last value in the ADC table)

- Actual ADC reading for the sensor is not greater than temporary register

- Actual ADC reading for the sensor is less than or equal to last x

##### 10.25.1.7.11 daugwylookup-LookupTableLookup-LLR-011

Requirement ID: H398-LLD-GWY-FNC-3677

The function shall do the following:

1. Set stat value with error status
2. Return array y value

when

- pi32\_x of table, pi32\_y of table and stat are not M\_HW\_NULL

- temporary register value is greater than last x (First value in the ADC table is greater than last value in the ADC table)

- Actual ADC reading for the sensor is not greater than temporary register

- Actual ADC reading for the sensor is less than or equal to last x

##### 10.25.1.7.12 daugwylookup-LookupTableLookup-LLR-012

Requirement ID: H398-LLD-GWY-FNC-3678

The function shall loop until (NOT of((Actual ADC reading for the sensor is greater than temporary register) AND (Actual ADC reading for the sensor less than or equal to dereference of (subtraction of array x by M\_ONE)))) returns TRUE and do the following:

1. Do the right shifting of number of points by one when number of points is greater than M\_ONE
2. Do nothing when number of points is not greater than M\_ONE

when

- pi32\_x of table, pi32\_y of table and stat are not M\_HW\_NULL

- temporary register value is greater than last x (First value in the ADC table is greater than last value in the ADC table)

- Actual ADC reading for the sensor is not greater than temporary register

- Actual ADC reading for the sensor is greater than last x

##### 10.25.1.7.13 daugwylookup-LookupTableLookup-LLR-013

Requirement ID: H398-LLD-GWY-FNC-3679

The function shall loop until (NOT of((Actual ADC reading for the sensor is greater than temporary register) AND (Actual ADC reading for the sensor less than or equal to dereference of (subtraction of array x by M\_ONE)))) returns TRUE and do the following:

1. Increment the array x by number of points and increment array y by number of points when temporary register is greater than or equal to Actual ADC reading for the sensor
2. Decrement the array x by number of points and decrement array y by number of points when temporary register is less than Actual ADC reading for the sensor

when

- pi32\_x of table, pi32\_y of table and stat are not M\_HW\_NULL

- temporary register value is greater than last x (First value in the ADC table is greater than last value in the ADC table)

- Actual ADC reading for the sensor is not greater than temporary register

- Actual ADC reading for the sensor is greater than last x

##### 10.25.1.7.14 daugwylookup-LookupTableLookup-LLR-014

Requirement ID: H398-LLD-GWY-FNC-3680

The function shall loop until (NOT of((Actual ADC reading for the sensor is greater than temporary register) AND (Actual ADC reading for the sensor less than or equal to dereference of (subtraction of array x by M\_ONE)))) returns TRUE and set temporary register with value of array x

when

- pi32\_x of table, pi32\_y of table and stat are not M\_HW\_NULL

- temporary register value is greater than last x (First value in the ADC table is greater than last value in the ADC table)

- Actual ADC reading for the sensor is not greater than temporary register

- Actual ADC reading for the sensor is greater than last x

##### 10.25.1.7.15 daugwylookup-LookupTableLookup-LLR-015

Requirement ID: H398-LLD-GWY-FNC-3681

The function shall set stat value with error status when pi32\_x of table, pi32\_y of table and stat are not M\_HW\_NULL

##### 10.25.1.7.16 daugwylookup-LookupTableLookup-LLR-016

Requirement ID: H398-LLD-GWY-FNC-3682

The function shall do the following when pi32\_x of table, pi32\_y of table and stat are not M\_HW\_NULL

1. Set segment delta x with array x value
2. Decrement segment delta x by pre decrement of array x value
3. Set segment delta y with array y value
4. Decrement segment delta y by pre decrement of array y value

##### 10.25.1.7.17 daugwylookup-LookupTableLookup-LLR-017

Requirement ID: H398-LLD-GWY-FNC-3683

The function shall do the following when pi32\_x of table, pi32\_y of table and stat are not M\_HW\_NULL

1. Set delta x with subtraction of Actual ADC reading for the sensor by array x value
2. Set results variable (product of (product of delta x and segment delta y) and M\_TWO)

##### 10.25.1.7.18 daugwylookup-LookupTableLookup-LLR-018

Requirement ID: H398-LLD-GWY-FNC-3684

The function shall return M\_DEC\_ZERO when

1. pi32\_x of table, pi32\_y of table and stat are not M\_HW\_NULL
2. Segment delta x is equal to M\_DEC\_ZERO, otherwise do nothing

##### 10.25.1.7.19 daugwylookup-LookupTableLookup-LLR-019

Requirement ID: H398-LLD-GWY-FNC-3685

The function shall set results variable with division of results variable by segment delta x when

1. pi32\_x of table, pi32\_y of table and stat are not M\_HW\_NULL

##### 10.25.1.7.20 daugwylookup-LookupTableLookup-LLR-020

Requirement ID: H398-LLD-GWY-FNC-3686

The function shall do the following when pi32\_x of table, pi32\_y of table and stat are not M\_HW\_NULL

1. Set negation to TRUE and set results variable with minus results variable to make the result positive when results variable is less than M\_DEC\_ZERO
2. Do nothing when results variable is not less than M\_DEC\_ZERO

##### 10.25.1.7.21 daugwylookup-LookupTableLookup-LLR-021

Requirement ID: H398-LLD-GWY-FNC-3687

The function shall do the following when pi32\_x of table, pi32\_y of table and stat are not M\_HW\_NULL

1. Set results variable with addition of results variable and M\_ONE.
2. Set results variable with results variable bit shifted to right by M\_ONE.

##### 10.25.1.7.22 daugwylookup-LookupTableLookup-LLR-022

Requirement ID: H398-LLD-GWY-FNC-3688

The function shall do the following when pi32\_x of table, pi32\_y of table and stat are not M\_HW\_NULL

1. Set results variable with minus results variable when negation is TRUE
2. Do nothing when negation is not TRUE

##### 10.25.1.7.23 daugwylookup-LookupTableLookup-LLR-023

Requirement ID: H398-LLD-GWY-FNC-3689

The function shall set results variable with addition of results and array y value when pi32\_x of table, pi32\_y of table and stat are not M\_HW\_NULL

##### 10.25.1.7.24 daugwylookup-LookupTableLookup-LLR-024

Requirement ID: H398-LLD-GWY-FNC-3690

The function shall return results variable

## 10.26 daugwymain

The daugwymain CSC checks different crc sections and initializes the drivers and application.

### 10.26.1 MainFunc

Low Level Design Details about CSU MainFunc will follow in the sub sections.

#### 10.26.1.1 Brief Description

The MainFunc function checks different crc sections and initializes the drivers and applications.

#### 10.26.1.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.26.1.3 List of global variables accessed and modified

Accessed : Cm\_data

Modified : A429\_mode

#### 10.26.1.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.26.1.5 Return Value

T\_UINT16 - returns zero

#### 10.26.1.6 Other CSUs called by this CSU

HwInit

OsInit

InitInit

OsStart

PbitCheck

CrcCheck

#### 10.26.1.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to MainFunc

##### 10.26.1.7.1 daugwymain-MainFunc-LLR-001

Requirement ID: H398-LLD-GWY-FNC-3700

The function shall call HwInit to Initialize System and Peripheral Clocks, hardware pins and interrupt controllers.

##### 10.26.1.7.2 daugwymain-MainFunc-LLR-002

Requirement ID: H398-LLD-GWY-FNC-3701

The function shall do the following when b\_out of Cm\_data is FALSE, otherwise set A429\_mode with A429\_NORMAL

-Perform CRC check for application configuration data and application data by calling function CrcCheck.

-Perform processor test by calling function PbitCheck.

##### 10.26.1.7.3 daugwymain-MainFunc-LLR-003

Requirement ID: H398-LLD-GWY-FNC-3702

The function shall initialize the Kernel by calling function OsInit.

##### 10.26.1.7.4 daugwymain-MainFunc-LLR-004

Requirement ID: H398-LLD-GWY-FNC-3703

The function shall initialize the Init Task by calling function InitInit.

##### 10.26.1.7.5 daugwymain-MainFunc-LLR-005

Requirement ID: H398-LLD-GWY-FNC-3704

The function shall start the Kernel by calling function OsStart.

##### 10.26.1.7.6 daugwymain-MainFunc-LLR-006

Requirement ID: H398-LLD-GWY-FNC-3705

The function shall return M\_ZERO.

## 10.27 daugwymontab

The daugwymontab CSC defines the Monitor Configuration data.

It does not contain any functions.

### 10.27.1 Brief Description

This file defines the Monitor Configuration data.

### 10.27.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

### 10.27.3 List of global variables accessed and modified

Accessed : None

Modified : None

### 10.27.4 Parameter list (Input/Output)

None

### 10.27.5 Return Value

None

### 10.27.6 Other CSUs called by this CSU

None

### 10.27.7 Description of list of LLRs allocated

This Module does not provide any public operation as its functionality is to just define the Monitor Configuration data. Hence, the LLR name does not contain a CSU name.

#### 10.27.7.1 daugwymontab-LLR-001

Requirement ID: H398-LLD-GWY-FNC-3714

The CSC daugwymontab shall define the Monitor Configuration data (Limits value).

## 10.28 daugwyoscpu

The daugwyoscpu CSC defines the OS Task creation function.

### 10.28.1 OsTaskCreate

Low Level Design Details about CSU OsTaskCreate will follow in the sub sections.

#### 10.28.1.1 Brief Description

The OsTaskCreate function is the OS Task creation function.

#### 10.28.1.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.28.1.3 List of global variables accessed and modified

Accessed : Os\_tcb\_prio\_tbl

Os\_running

U32\_critical\_sr

Modified : U32\_critical\_sr

#### 10.28.1.4 Parameter list (Input/Output)

Inputs : void (\*ptask) (void \*dptr) function pointer to the task

void \*pdata pointer to task parameters

T\_UINT8 u8\_prio task priority

void \*pt\_os pointer to top of stack

Outputs : None

#### 10.28.1.5 Return Value

T\_UINT8 - Returns task creation status

M\_OS\_PRIO\_EXIST : Task already exists

M\_OS\_NO\_ERR : Task created successfully

M\_OS\_PRIO\_INVALID : Invalid priority

#### 10.28.1.6 Other CSUs called by this CSU

SaveStatusReg

RestoreStatusReg

OsTcbInit

OsSched

#### 10.28.1.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to OsTaskCreate.

##### 10.28.1.7.1 daugwyoscpu-OsTaskCreate-LLR-001

Requirement ID: H398-LLD-GWY-FNC-3724

The function shall return with M\_OS\_PRIO\_INVALID when u8\_prio is greater than M\_OS\_LOWEST\_PRIO otherwise do nothing.

##### 10.28.1.7.2 daugwyoscpu-OsTaskCreate-LLR-002

Requirement ID: H398-LLD-GWY-FNC-3725

The function shall call SaveStatusReg(M\_OS\_ENTER\_CRITICAL) and set return value of SaveStatusReg to U32\_critical\_sr to disable interrupts.

##### 10.28.1.7.3 daugwyoscpu-OsTaskCreate-LLR-003

Requirement ID: H398-LLD-GWY-FNC-3726

The function shall do the following to avoid creating task if already exist, when Os\_tcb\_prio\_tbl with index u8\_prio is equal to M\_ZERO:

1. Call RestoreStatusReg(M\_OS\_EXIT\_CRITICAL) with parameter U32\_critical\_sr

##### 10.28.1.7.4 daugwyoscpu-OsTaskCreate-LLR-004

Requirement ID: H398-LLD-GWY-FNC-3727

The function shall do the following to avoid creating task if already exist, when Os\_tcb\_prio\_tbl with index u8\_prio is equal to M\_ZERO:

1. Set Stack top with pt\_os
2. Set dereference of (pre-decrement of Stack top) with M\_FPCCR.
3. Set dereference of (pre-decrement of Stack top) with M\_FPSCR
4. Set Stack top with (subtraction of Stack top with M\_STK\_16)
5. Set dereference of (pre-decrement of Stack top) with M\_XPSR
6. Set dereference of (pre-decrement of Stack top) with ptask
7. Set dereference of (pre-decrement of Stack top) with M\_R14\_LR
8. Set Stack top with (subtraction of Stack top with M\_STK\_4)
9. Set dereference of (pre-decrement of Stack top) with pdata
10. Set Stack top with (subtraction of Stack top with M\_STK\_24)

##### 10.28.1.7.5 daugwyoscpu-OsTaskCreate-LLR-005

Requirement ID: H398-LLD-GWY-FNC-3728

The function shall do the following to avoid creating task if already exist, when Os\_tcb\_prio\_tbl with index u8\_prio is equal to M\_ZERO:

1. Call OsTcbInit with parameters (u8\_prio and Stack top) and store the return value in error status.

##### 10.28.1.7.6 daugwyoscpu-OsTaskCreate-LLR-006

Requirement ID: H398-LLD-GWY-FNC-3729

The function shall do the following to avoid creating task if already exist, when Os\_tcb\_prio\_tbl with index u8\_prio is equal to M\_ZERO:

1. Call OsSched when error status is equal to M\_OS\_NO\_ERR and Os\_running is TRUE
2. Do nothing when any of the below condition is satisfied:

* error status is not equal to M\_OS\_NO\_ERR
* Os\_running is not TRUE

##### 10.28.1.7.7 daugwyoscpu-OsTaskCreate-LLR-007

Requirement ID: H398-LLD-GWY-FNC-3730

The function shall return the error status when Os\_tcb\_prio\_tbl with index u8\_prio is equal to M\_ZERO.

##### 10.28.1.7.8 daugwyoscpu-OsTaskCreate-LLR-008

Requirement ID: H398-LLD-GWY-FNC-3731

The function shall do the following when Os\_tcb\_prio\_tbl with index u8\_prio is not equal to M\_ZERO:

1. Call RestoreStatusReg(M\_OS\_EXIT\_CRITICAL) with parameter U32\_critical\_sr

##### 10.28.1.7.9 daugwyoscpu-OsTaskCreate-LLR-009

Requirement ID: H398-LLD-GWY-FNC-3732

The function shall return M\_OS\_PRIO\_EXIST when Os\_tcb\_prio\_tbl with index u8\_prio is not equal to M\_ZERO

## 10.29 daugwyoscpua

The daugwyoscpua CSC defines asm routines for uCOS.

### 10.29.1 OsCtxSw

Low Level Design Details about CSU OsCtxSw will follow in the sub sections.

#### 10.29.1.1 Brief Description

The OsCtxSw function performs a task level context switch.

#### 10.29.1.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD(H398-003-012-GWY).

#### 10.29.1.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.29.1.4 Parameter list (Input/Output)

Inputs : None

Outputs :None

#### 10.29.1.5 Return Value

None

#### 10.29.1.6 Other CSUs called by this CSU

None

#### 10.29.1.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to OsCtxSw.

##### 10.29.1.7.1 daugwyoscpua-OsCtxSw-LLR-001

Requirement ID: H398-LLD-GWY-FNC-3742

The function shall do the following to trigger the PendSV exception

- Load the Interrupt control state register(M\_NVIC\_INT\_CTRL) to R0

- Load the Value to trigger PendSV isr (M\_NVIC\_PENDSVSET) to R1

- Write R1 content to address of R0.

- Return from the routine

### 10.29.2 OsIntCtxSw

Low Level Design Details about CSU OSIntCtxSw will follow in the sub sections.

#### 10.29.2.1 Brief Description

The OSIntCtxSw function performs a task level context switch same as OsCtxSw.

#### 10.29.2.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD(H398-003-012-GWY).

#### 10.29.2.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.29.2.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.29.2.5 Return Value

None

#### 10.29.2.6 Other CSUs called by this CSU

None

#### 10.29.2.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to OsIntCtxSw

##### 10.29.2.7.1 daugwyoscpua-OsIntCtxSw-LLR-001

Requirement ID: H398-LLD-GWY-FNC-3751

The function shall do the following to trigger the PendSV exception

- Load the Interrupt control state register(M\_NVIC\_INT\_CTRL) to R0

- Load the Value to trigger PendSV isr(M\_NVIC\_PENDSVSET) to R1

- write R1 content to address of R0.

- Return from the routine

### 10.29.3 SaveStatusReg

Low Level Design Details about CSU SaveStatusReg will follow in the sub sections.

#### 10.29.3.1 Brief Description

The SaveStatusReg function save the interrupt mask register and then disables interrupts to enter into critical section.

#### 10.29.3.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD(H398-003-012-GWY).

#### 10.29.3.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.29.3.4 Parameter list (Input/Output)

Inputs : None

Outputs :None

#### 10.29.3.5 Return Value

T\_UINT32 - returns Critical status register value

#### 10.29.3.6 Other CSUs called by this CSU

None

#### 10.29.3.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to SaveStatusReg.

##### 10.29.3.7.1 daugwyoscpua-SaveStatusReg-LLR-001

Requirement ID: H398-LLD-GWY-FNC-3760

The function shall do the following to save the interrupt mask register and then disables interrupts

- Read PRIMASK register value and write it to R0

- Disable IRq (interrupts) by setting the PRIMASK register value.

- Return from the routine

### 10.29.4 RestoreStatusReg

Low Level Design Details about CSU RestoreStatusReg will follow in the sub sections.

#### 10.29.4.1 Brief Description

The RestoreStatusReg function restores the interrupt disable mask to its original value.

#### 10.29.4.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD(H398-003-012-GWY).

#### 10.29.4.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.29.4.4 Parameter list (Input/Output)

Inputs : T\_UINT32 U32\_critical\_sr - Critical status register value

Outputs : None

#### 10.29.4.5 Return Value

None

#### 10.29.4.6 Other CSUs called by this CSU

None

#### 10.29.4.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to RestoreStatusReg.

##### 10.29.4.7.1 daugwyoscpua-RestoreStatusReg-LLR-001

Requirement ID: H398-LLD-GWY-FNC-3769

The function shall write the contents of R0 into PRIMASK register to enable interrupts (Restore saved interrupts).

### 10.29.5 OsStartHighRdy

Low Level Design Details about CSU OsStartHighRdy will follow in the sub sections.

#### 10.29.5.1 Brief Description

The OsStartHighRdy function starts running the highest priority task.

#### 10.29.5.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD(H398-003-012-GWY).

#### 10.29.5.3 List of global variables accessed and modified

Accessed : Os\_running

Modified : None

#### 10.29.5.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.29.5.5 Return Value

None

#### 10.29.5.6 Other CSUs called by this CSU

None

#### 10.29.5.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to OsStartHighRdy.

##### 10.29.5.7.1 daugwyoscpua-OsStartHighRdy-LLR-001

Requirement ID: H398-LLD-GWY-FNC-3778

The function shall do the following to set the PendSV exception priority

- Load the System priority register(M\_NVIC\_SYSPRI14) to R0

- Load the PendSV priority value(M\_NVIC\_PENDSV\_PRI) to R1

- Write R1 content as unsigned byte to address of R0.

##### 10.29.5.7.2 daugwyoscpua-OsStartHighRdy-LLR-002

Requirement ID: H398-LLD-GWY-FNC-3779

The function shall do the following to Set the PSP to M\_DEFAULT\_VAL for initial context switch call

- Write value of M\_DEFAULT\_VAL to R0

- Writes the contents of R0 into PSP register.

##### 10.29.5.7.3 daugwyoscpua-OsStartHighRdy-LLR-003

Requirement ID: H398-LLD-GWY-FNC-3780

The function shall do the following to Set the Os\_running flag to TRUE (indicate that multitasking will start)

- Load the Os\_running flag address to R0

- Write value of M\_DEFAULT\_TRUE\_VAL to R1

- Write R1 content as unsigned byte to address of R0.

##### 10.29.5.7.4 daugwyoscpua-OsStartHighRdy-LLR-004

Requirement ID: H398-LLD-GWY-FNC-3781

The function shall do the following to trigger the PendSV exception

- Load the Interrupt control state register(M\_NVIC\_INT\_CTRL) to R0

- Load the Value to trigger PendSV isr(M\_NVIC\_PENDSVSET) to R1

- Write R1 content as unsigned byte to address of R0.

##### 10.29.5.7.5 daugwyoscpua-OsStartHighRdy-LLR-005

Requirement ID: H398-LLD-GWY-FNC-3782

The function shall do the following to enable interrupts

- Enable Interrupt request by clearing the PRIMASK register value.

##### 10.29.5.7.6 daugwyoscpua-OsStartHighRdy-LLR-006

Requirement ID: H398-LLD-GWY-FNC-3783

The function shall enter infinite wait state when highest priority task could not be scheduled.

### 10.29.6 PendSvHandler

Low Level Design Details about CSU PendSvHandler will follow in the sub sections.

#### 10.29.6.1 Brief Description

The PendSvHandler function handles all context switching for uCOS.

#### 10.29.6.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD(H398-003-012-GWY).

#### 10.29.6.3 List of global variables accessed and modified

Accessed : Os\_tcb\_cur

Os\_tcb\_high\_rdy

Modified : None

#### 10.29.6.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.29.6.5 Return Value

None

#### 10.29.6.6 Other CSUs called by this CSU

None

#### 10.29.6.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to PendSvHandler.

##### 10.29.6.7.1 daugwyoscpua-PendSvHandler-LLR-001

Requirement ID: H398-LLD-GWY-FNC-3792

The function shall do the following to Prevent interruption during context switch

- Disable Interrupt request by setting the I-bit in the CPSR.

##### 10.29.6.7.2 daugwyoscpua-PendSvHandler-LLR-002

Requirement ID: H398-LLD-GWY-FNC-3793

The function shall do the following to save the process stack pointer (PSP)

- Read PSP register value and write it to R0.

##### 10.29.6.7.3 daugwyoscpua-PendSvHandler-LLR-003

Requirement ID: H398-LLD-GWY-FNC-3794

The function shall do the following to skip register save the first time

- Forward branch to 'PendSvHandler\_NoSave' when R0 is zero.

##### 10.29.6.7.4 daugwyoscpua-PendSvHandler-LLR-004

Requirement ID: H398-LLD-GWY-FNC-3795

The function shall do the following to save remaining register R4-R11 on process stack

- Decrement R0 Before each access and load the register R4-R11 to the Decremented address.

##### 10.29.6.7.5 daugwyoscpua-PendSvHandler-LLR-005

Requirement ID: H398-LLD-GWY-FNC-3796

The function shall do the following to save remaining floating point register S16-S31

- Decrement R0 Before each access and store the register S16-S31 to the Decremented address.

##### 10.29.6.7.6 daugwyoscpua-PendSvHandler-LLR-006

Requirement ID: H398-LLD-GWY-FNC-3797

The function shall do the following to load the Stack pointer to the TCB stack

- Load the current TCB address(Os\_tcb\_cur) to R1

- Load the address content in R1 to R1

- write R0 content to address of R1

##### 10.29.6.7.7 daugwyoscpua-PendSvHandler-LLR-007

Requirement ID: H398-LLD-GWY-FNC-3798

The function shall do the following to set highest priority task ready to execute into the current TCB (i.e., Os\_tcb\_cur equal to Os\_tcb\_high\_rdy) when branch PendSvHandler\_NoSave is invoked

- Load the current TCB(Os\_tcb\_cur) address to R0

- Load the address of highest priority task ready to execute (Os\_tcb\_high\_rdy) to R1

- Load the content of address in R1 to R2

- write R2 content to address of R0

- Load the address content in R2 to R0

(i.e. - Set Stack Pointer (R0) to os\_tcb\_stkptr value of Os\_tcb\_high\_rdy)

- Increment R0 After each access and restore the register S16- S31

from the Incremented address

- Increment R0 After each access and load the register R4 - R11 from

the Incremented address

- Load PSP with the new stack pointer (R0)

- Load the link register to link register bitwise OR M\_EXCEPTION\_RETURN

- Enable Interrupt request by clearing the PRIMASK register value.

- Return from the routine

## 10.30 daugwyqueue

The daugwyqueue CSC contains implementation of queue used for DAU operations.

### 10.30.1 InitQueue

Low Level Design Details about CSU InitQueue will follow in the sub sections.

#### 10.30.1.1 Brief Description

The InitQueue function initializes Queue.

#### 10.30.1.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.30.1.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.30.1.4 Parameter list (Input/Output)

Inputs : T\_QUEUE \*queu - pointer to the queue

Outputs : T\_QUEUE \*queu - pointer to the queue

#### 10.30.1.5 Return Value

None

#### 10.30.1.6 Other CSUs called by this CSU

HwMemset

#### 10.30.1.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to InitQueue

##### 10.30.1.7.1 daugwyqueue-InitQueue-LLR-001

Requirement ID: H398-LLD-GWY-FNC-3808

The function shall do the following

1. set count of queu to M\_ZERO
2. set front of queu to M\_ZERO
3. set rear of queu to M\_MINUS\_ONE

##### 10.30.1.7.2 daugwyqueue-InitQueue-LLR-002

Requirement ID: H398-LLD-GWY-FNC-3809

The function shall initialize the queue with zero by calling HwMemset with parameters items of queu, M\_ZERO and size of (items of queu).

### 10.30.2 IsEmpty

Low Level Design Details about CSU IsEmpty will follow in the sub sections.

10.30.2.1 Brief Description

The IsEmpty function checks if Queue is empty.

#### 10.30.2.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.30.2.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.30.2.4 Parameter list (Input/Output)

Inputs : T\_QUEUE \*queu - pointer to the queue

Outputs : None

#### 10.30.2.5 Return Value

T\_BOOL Return the status of Queue,

return true if queue is empty,

return false if queue is not empty.

#### 10.30.2.6 Other CSUs called by this CSU

None

#### 10.30.2.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to IsEmpty

##### 10.30.2.7.1 daugwyqueue-IsEmpty-LLR-001

Requirement ID: H398-LLD-GWY-FNC-3818

The function shall return true when count of queu is equal to M\_ZERO, otherwise return false.

### 10.30.3 IsFull

Low Level Design Details about CSU IsFull will follow in the sub sections.

#### 10.30.3.1 Brief Description

The IsFull function checks if Queue is full.

#### 10.30.3.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.30.3.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.30.3.4 Parameter list (Input/Output)

Inputs : T\_QUEUE \*queu - pointer to the queue

Outputs : None

#### 10.30.3.5 Return Value

T\_BOOL Return the status of Queue,

return true if queue is full,

return false if queue is not full.

#### 10.30.3.6 Other CSUs called by this CSU

None

#### 10.30.3.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to IsFull

##### 10.30.3.7.1 daugwyqueue-IsFull-LLR-001

Requirement ID: H398-LLD-GWY-FNC-3827

The function shall return true when count of queu is equal to M\_MAX\_SIZE, otherwise return false.

### 10.30.4 EnQueue

Low Level Design Details about CSU EnQueue will follow in the sub sections.

#### 10.30.4.1 Brief Description

The EnQueue function places value into queue.

#### 10.30.4.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.30.4.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.30.4.4 Parameter list (Input/Output)

Inputs : T\_QUEUE \*queu - pointer to the queue,

T\_UINT8 x\_item - item to be placed into the queue

Outputs : T\_QUEUE \*queu - pointer to the queue

#### 10.30.4.5 Return Value

None

#### 10.30.4.6 Other CSUs called by this CSU

None

#### 10.30.4.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to EnQueue

##### 10.30.4.7.1 daugwyqueue-EnQueue-LLR-001

Requirement ID: H398-LLD-GWY-FNC-3836

The function shall return from the function when count of queu is equal to M\_MAX\_SIZE.

##### 10.30.4.7.2 daugwyqueue-EnQueue-LLR-002

Requirement ID: H398-LLD-GWY-FNC-3837

The function shall do the following when count of queu is not equal to M\_MAX\_SIZE:

1. increment the count of queu by M\_ONE
2. Set rear of queu with (modulus of (addition of rear of queu and M\_ONE) by M\_MAX\_SIZE)
3. Set items of queu for index rear of queu with x\_item

### 10.30.5 DeQueue

Low Level Design Details about CSU DeQueue will follow in the sub sections.

#### 10.30.5.1 Brief Description

The DeQueue function get item from queue.

#### 10.30.5.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.30.5.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.30.5.4 Parameter list (Input/Output)

Inputs : T\_QUEUE \*queu - pointer to the queue

Outputs : T\_QUEUE \*queu - pointer to the queue

#### 10.30.5.5 Return Value

T\_UINT8 Return the item from queue buffer.

#### 10.30.5.6 Other CSUs called by this CSU

None

#### 10.30.5.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to DeQueue

##### 10.30.5.7.1 daugwyqueue-DeQueue-LLR-001

Requirement ID: H398-LLD-GWY-FNC-3846

The function shall do the following :

1. Decrement the count of queu by M\_ONE
2. Set item x with items of queu for index front of queu
3. Set front of queu with (modulus of (addition of front of queu and M\_ONE) and M\_MAX\_SIZE)
4. Return the item x value

## 10.31 daugwyrevno

The daugwyrevno CSC defines the Top level software configuration number and Gateway Software Application Part Numbers.

It does not contain any functions.

### 10.31.1 Brief Description

This file defines the Gateway Module Application software part number and Top level software configuration number.

### 10.31.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

### 10.31.3 List of global variables accessed and modified

Accessed : None

Modified : None

### 10.31.4 Parameter list (Input/Output)

None

### 10.31.5 Return Value

None

### 10.31.6 Other CSUs called by this CSU

### 10.31.7 Description of list of LLRs allocated

This Module does not provide any public operation as its functionality is to just provide the revision number. Hence, the LLR name does not contain a CSU name.

#### 10.31.7.1 daugwyrevno-LLR-001

Requirement ID: H398-LLD-GWY-FNC-3855

The CSC daugwyrevno shall define the Top level software configuration number (SCN) and Gateway Application Software Part Number by setting Revno\_id of index M\_REVNO\_LEN with M\_REVNO\_REVISION

## 10.32 daugwyrs232

The daugwyrs232 CSC defines Implementation of rs232/rs422 driver, Initialization of all the COM ports and Receive data from right PFD and EICAS.

### 10.32.1 TxEnable

Low Level Design Details about CSU TxEnable will follow in the sub sections.

#### 10.32.1.1 Brief Description

The TxEnable function enable the USARTx transmit interrupt based on u8\_com\_port.

#### 10.32.1.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.32.1.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.32.1.4 Parameter list (Input/Output)

Inputs : T\_UINT8 u8\_com\_port - comport number

Outputs : None

#### 10.32.1.5 Return Value

None

#### 10.32.1.6 Other CSUs called by this CSU

UsartItConfig

#### 10.32.1.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TxEnable

##### 10.32.1.7.1 daugwyrs232-TxEnable-LLR-002

Requirement ID: H398-LLD-GWY-FNC-3866

The function shall enable the M\_USART2 transmit interrupt by calling 'UsartItConfig' with parameters M\_USART2, M\_USART\_IT\_TXE and ENABLE when u8\_com\_port is RS232\_COM2.

##### 10.32.1.7.2 daugwyrs232-TxEnable-LLR-004

Requirement ID: H398-LLD-GWY-FNC-3868

The function shall enable the M\_UART4 transmit interrupt by calling 'UsartItConfig' with parameters M\_USART4, M\_USART\_IT\_TXE and ENABLE when u8\_com\_port is RS232\_COM4.

##### 10.32.1.7.3 daugwyrs232-TxEnable-LLR-005

Requirement ID: H398-LLD-GWY-FNC-3869

The function shall do nothing when u8\_com\_port is other than RS232\_COM1, RS232\_COM2, RS232\_COM3 and RS232\_COM4..

### 10.32.2 Rs232TxBlock

Low Level Design Details about CSU Rs232TxBlock will follow in the sub sections.

#### 10.32.2.1 Brief Description

The Rs232TxBlock function transmits the data from source buffer through RS232.

#### 10.32.2.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.32.2.3 List of global variables accessed and modified

Accessed : U32\_critical\_sr

Modified : U32\_critical\_sr

#### 10.32.2.4 Parameter list (Input/Output)

Inputs : T\_UINT16 u16\_cnt - length of data to be sent,

T\_UINT8 u8\_com\_port - comport number.

Outputs : T\_UINT8 \*pu8\_block - source buffer,

T\_UINT16 u16\_cnt - length of data to be sent

#### 10.32.2.5 Return Value

T\_RS232\_STATUS Return the transmission status of RS232,

RS232\_OK

RS232\_BAD\_COMM

RS232\_WAITING

RS232\_OK

#### 10.32.2.6 Other CSUs called by this CSU

TxEnable

SaveStatusReg

RestoreStatusReg

#### 10.32.2.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to Rs232TxBlock

##### 10.32.2.7.1 daugwyrs232-Rs232TxBlock-LLR-001

Requirement ID: H398-LLD-GWY-FNC-3878

The function shall return RS232\_OK when the u16\_cnt is M\_ZERO.

##### 10.32.2.7.2 daugwyrs232-Rs232TxBlock-LLR-002

Requirement ID: H398-LLD-GWY-FNC-3879

The function shall return RS232\_BAD\_COMM when the u16\_cnt is greater than M\_RS232\_TXBUF\_SIZE.

##### 10.32.2.7.3 daugwyrs232-Rs232TxBlock-LLR-003

Requirement ID: H398-LLD-GWY-FNC-3880

The function shall return RS232\_WAITING when the u16\_cnt is greater than subtraction of transmitter counter (u16\_tx\_cntr) of RS232 data structure of corresponding u8\_com\_port from M\_RS232\_TXBUF\_SIZE.

##### 10.32.2.7.4 daugwyrs232-Rs232TxBlock-LLR-004

Requirement ID: H398-LLD-GWY-FNC-3881

The function shall do the following till u16\_cnt reaches M\_ZERO

a) disable the interrupts by macro function call M\_OS\_ENTER\_CRITICAL,

b) store the character to be sent(pu8\_block) into address at transmission tail (pu8\_tx\_tail) of RS232 data structure of corresponding u8\_com\_port,

c) increment the transmission tail (pu8\_tx\_tail) of RS232 data structure of corresponding u8\_com\_port by M\_ONE.

d) assign base address of transmission buffer (au8\_tx\_buf) of RS232 data structure of corresponding u8\_com\_port to transmission tail of RS232 data structure of corresponding u8\_com\_port when the transmission tail (pu8\_tx\_tail) of RS232 data structure of corresponding u8\_com\_port is equal to end address of transmission buffer with index M\_RS232\_TXBUF\_SIZE of RS232 data structure of corresponding u8\_com\_port,

e) increment the transmission counter (u16\_tx\_cntr) of RS232 data structure of corresponding u8\_com\_port by M\_ONE,

f) enable the interrupts by macro function call M\_OS\_EXIT\_CRITICAL,

g) increment pu8\_block by M\_ONE,

h) decrement u16\_cnt by M\_ONE.

When the u16\_cnt is other than M\_ZERO and u16\_cnt is less than or equal to to subtraction of transmitter counter (u16\_tx\_cntr) of RS232 data structure of corresponding u8\_com\_port from M\_RS232\_TXBUF\_SIZE.

##### 10.32.2.7.5 daugwyrs232-Rs232TxBlock-LLR-005

Requirement ID: H398-LLD-GWY-FNC-3882

The function shall perform the following When the u16\_cnt is other than M\_ZERO and and u16\_cnt is less than or equal to to subtraction of transmitter counter (u16\_tx\_cntr) of RS232 data structure of corresponding u8\_com\_port from M\_RS232\_TXBUF\_SIZE.

1. Return enable transmit interrupt by calling 'TxEnable' with parameter u8\_com\_port when data is there for transmission.

##### 10.32.2.7.6 daugwyrs232-Rs232TxBlock-LLR-006

Requirement ID: H398-LLD-GWY-FNC-3883

The function shall return RS232\_OK when data is transmitted successfully.

### 10.32.3 Rs232RxBlock

Low Level Design Details about CSU Rs232RxBlock will follow in the sub sections.

#### 10.32.3.1 Brief Description

The Rs232RxBlock function retrieves the data from receive buffer of RS232.

#### 10.32.3.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.32.3.3 List of global variables accessed and modified

Accessed : U32\_critical\_sr

Modified : U32\_critical\_sr

#### 10.32.3.4 Parameter list (Input/Output)

Inputs : T\_UINT8 \*pu8\_block - destination buffer,

T\_UINT16 u16\_cnt - length of data to be copied,

T\_UINT8 u8\_com\_port - comport number.

Outputs : T\_UINT8 \*pu8\_block - destination buffer,

T\_UINT16 u16\_cnt - length of data to be sent.

#### 10.32.3.5 Return Value

T\_RS232\_STATUS Return the receive status of RS232,

RS232\_OK

RS232\_BAD\_COMM

RS232\_WAITING

RS232\_OK

#### 10.32.3.6 Other CSUs called by this CSU

SaveStatusReg

RestoreStatusReg

UsartItConfig

#### 10.32.3.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to Rs232RxBlock

##### 10.32.3.7.1 daugwyrs232-Rs232RxBlock-LLR-001

Requirement ID: H398-LLD-GWY-FNC-3892

The function shall enable the M\_UART1 transmit interrupt by calling 'UsartItConfig' with parameters M\_USART1, M\_USART\_IT\_RNXE and ENABLE and return RS232\_OK when the u16\_cnt is equal to M\_ZERO.

##### 10.32.3.7.2 daugwyrs232-Rs232RxBlock-LLR-002

Requirement ID: H398-LLD-GWY-FNC-3893

The function shall return RS232\_BAD\_COMM when the u16\_cnt is greater than M\_RS232\_RXBUF\_SIZE.

##### 10.32.3.7.3 daugwyrs232-Rs232RxBlock-LLR-003

Requirement ID: H398-LLD-GWY-FNC-3894

The function shall return RS232\_WAITING when the u16\_cnt is greater than receiver counter (u16\_rx\_cntr) of RS232 data structure of corresponding u8\_com\_port.

##### 10.32.3.7.4 daugwyrs232-Rs232RxBlock-LLR-004

Requirement ID: H398-LLD-GWY-FNC-3895

The function shall do the following till u16\_cnt is other than M\_ZERO

a) disable the interrupts by macro function call M\_OS\_ENTER\_CRITICAL,

b) store the character into pu8\_block from value at address of receiver head (pu8\_rx\_head) of RS232 data structure of corresponding u8\_com\_port,

c) increment the receiver head (pu8\_rx\_head) of RS232 data structure of corresponding u8\_com\_port by M\_ONE.

When the u16\_cnt is other than M\_ZERO, less than M\_RS232\_RXBUF\_SIZE and less than receiver counter of RS232 data structure of corresponding u8\_com\_port.

##### 10.32.3.7.5 daugwyrs232-Rs232RxBlock-LLR-005

Requirement ID: H398-LLD-GWY-FNC-3896

The function shall do the following till u16\_cnt is other than M\_ZERO

a) assign base address of receiver buffer of RS232 data structure of corresponding u8\_com\_port to receiver head (pu8\_rx\_head) of RS232 data structure of corresponding u8\_com\_port when the receiver head (pu8\_rx\_head) of RS232 data structure of corresponding u8\_com\_port is equal to end address of receiver buffer with index M\_RS232\_RXBUF\_SIZE of RS232 data structure of corresponding u8\_com\_port,

b) decrement the reciever counter (u16\_rx\_cntr) of RS232 data structure of corresponding u8\_com\_port by M\_ONE,

c) enable the interrupts by macro function call M\_OS\_EXIT\_CRITICAL,

d) increment pu8\_block by M\_ONE,

e) decrement u16\_cnt by M\_ONE.

When the u16\_cnt is other than M\_ZERO, less than M\_RS232\_RXBUF\_SIZE and less than receiver counter of RS232 data structure of corresponding u8\_com\_port.

##### 10.32.3.7.6 daugwyrs232-Rs232RxBlock-LLR-006

Requirement ID: H398-LLD-GWY-FNC-3897

The function shall return RS232\_OK when data is copied from receive buffer of RS232.

### 10.32.4 InitComData

Low Level Design Details about CSU InitComData will follow in the sub sections.

#### 10.32.4.1 Brief Description

The InitComData function initializes RS232 data structure.

#### 10.32.4.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.32.4.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.32.4.4 Parameter list (Input/Output)

Inputs : T\_RS232\_DATA \*ps\_com\_data - pointer to the RS232 data structure

Outputs : T\_RS232\_DATA \*ps\_com\_data - pointer to the RS232 data structure

#### 10.32.4.5 Return Value

None

#### 10.32.4.6 Other CSUs called by this CSU

None

#### 10.32.4.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to InitComData

##### 10.32.4.7.1 daugwyrs232-InitComData-LLR-001

Requirement ID: H398-LLD-GWY-FNC-3906

The function shall initialize RS232 data structure as follows

a) set transmitter counter (u16\_tx\_cntr) of ps\_com\_data to M\_ZERO,

b) set transmitter head (pu8\_tx\_head) of ps\_com\_data to base address of transmitter buffer with index as( M\_ZERO) in ps\_com\_data,

c) set transmitter tail (pu8\_tx\_tail) of ps\_com\_data to base address of transmitter buffer with index as( M\_ZERO) in ps\_com\_data,

d) set receiver counter (u16\_rx\_cntr) of ps\_com\_data to M\_ZERO,

e) set receiver head (pu8\_rx\_head) of ps\_com\_data to base address of receiver buffer with index as( M\_ZERO) in ps\_com\_data,

f) set receiver tail (pu8\_rx\_tail) of ps\_com\_data to base address of receiver buffer with index as( M\_ZERO) in ps\_com\_data.

### 10.32.5 TransmitChar

Low Level Design Details about CSU TransmitChar will follow in the sub sections.

#### 10.32.5.1 Brief Description

The TransmitChar function transmits the data from RS232 based on u8\_com\_port.

#### 10.32.5.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.32.5.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.32.5.4 Parameter list (Input/Output)

Inputs : T\_UINT8 u8\_char - byte of data to be sent out,

T\_UINT8 u8\_com\_port - comport number.

Outputs : None

#### 10.32.5.5 Return Value

None

#### 10.32.5.6 Other CSUs called by this CSU

UsartSendData

#### 10.32.5.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TransmitChar

##### 10.32.5.7.1 daugwyrs232-TransmitChar-LLR-002

Requirement ID: H398-LLD-GWY-FNC-3916

The function shall transmit the byte of data by calling 'UsartSendData' with parameters M\_USART2 and u8\_char when u8\_com\_port is RS232\_COM2.

##### 10.32.5.7.2 daugwyrs232-TransmitChar-LLR-004

Requirement ID: H398-LLD-GWY-FNC-3918

The function shall transmit the byte of data by calling 'UsartSendData' with parameters M\_UART4 and u8\_char when u8\_com\_port is RS232\_COM4.

##### 10.32.5.7.3 daugwyrs232-TransmitChar-LLR-005

Requirement ID: H398-LLD-GWY-FNC-3919

The function shall do nothing when u8\_com\_port is other than RS232\_COM2 and RS232\_COM4.

### 10.32.6 ResetCOM

Low Level Design Details about CSU ResetCOM will follow in the sub sections.

#### 10.32.6.1 Brief Description

The ResetCOM function resets the RS232 based u8\_com\_port.

#### 10.32.6.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.32.6.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.32.6.4 Parameter list (Input/Output)

Inputs : T\_UINT8 u8\_com\_port - comport number.

Outputs : None

#### 10.32.6.5 Return Value

None

#### 10.32.6.6 Other CSUs called by this CSU

InitCOM2

InitCOM4

COM2Intr

COM4Intr

#### 10.32.6.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to ResetCOM

##### 10.32.6.7.1 daugwyrs232-ResetCOM-LLR-002

Requirement ID: H398-LLD-GWY-FNC-3929

The function shall perform the following

Copies the M\_USART2 to usart com port2 and initialize the com port2 by calling ‘InitCOM2‘ with parameter COM2Intr when u8\_com\_port is RS232\_COM2.

##### 10.32.6.7.2 daugwyrs232-ResetCOM-LLR-004

Requirement ID: H398-LLD-GWY-FNC-3931

The function shall perform the following

Copies the M\_USART4 to usart com port4 and initialize the com port4 by calling ‘InitCOM4‘ with parameter COM4Intr when u8\_com\_port is RS232\_COM4

##### 10.32.6.7.3 daugwyrs232-ResetCOM-LLR-005

Requirement ID: H398-LLD-GWY-FNC-3932

The function shall do nothing when u8\_com\_port is other than RS232\_COM2 and RS232\_COM4.

### 10.32.7 DisableTxIntr

Low Level Design Details about CSU DisableTxIntr will follow in the sub sections.

#### 10.32.7.1 Brief Description

The DisableTxIntr function disable the USARTx transmit interrupt based on u8\_com\_port.

#### 10.32.7.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.32.7.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.32.7.4 Parameter list (Input/Output)

Inputs : T\_UINT8 u8\_com\_port - comport number

Outputs : None

#### 10.32.7.5 Return Value

None

#### 10.32.7.6 Other CSUs called by this CSU

UsartItConfig

#### 10.32.7.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to DisableTxIntr

##### 10.32.7.7.1 daugwyrs232-DisableTxIntr-LLR-002

Requirement ID: H398-LLD-GWY-FNC-3942

The function shall disable the USART2 transmit interrupt by calling 'UsartItConfig' with parameters M\_USART2, M\_USART\_IT\_TXE and DISABLE when u8\_com\_port is RS232\_COM2.

##### 10.32.7.7.2 daugwyrs232-DisableTxIntr-LLR-004

Requirement ID: H398-LLD-GWY-FNC-3944

The function shall disable the USART4 transmit interrupt by calling 'UsartItConfig' with parameters M\_UART4, M\_USART\_IT\_TXE and DISABLE when u8\_com\_port is RS232\_COM4.

##### 10.32.7.7.3 daugwyrs232-DisableTxIntr-LLR-005

Requirement ID: H398-LLD-GWY-FNC-3945

The function shall do nothing when u8\_com\_port is other than RS232\_COM2 and RS232\_COM4.

### 10.32.8 ReceiveChar

Low Level Design Details about CSU ReceiveChar will follow in the sub sections.

#### 10.32.8.1 Brief Description

The ReceiveChar function receives data from RS232 based on u8\_com\_port.

#### 10.32.8.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.32.8.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.32.8.4 Parameter list (Input/Output)

Inputs : T\_UINT8 \*pu8\_dest - Pointer to the destination buffer

in which read data is stored,

T\_UINT8 u8\_com\_port - comport number.

Outputs : T\_UINT8 \*pu8\_dest - Pointer to the destination buffer

in which read data is stored

#### 10.32.8.5 Return Value

None

#### 10.32.8.6 Other CSUs called by this CSU

UsartReceiveData

#### 10.32.8.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to ReceiveChar

##### 10.32.8.7.1 daugwyrs232-ReceiveChar-LLR-002

Requirement ID: H398-LLD-GWY-FNC-3955

The function shall receive byte of data (reference of pu8\_dest) by calling 'UsartReceiveData' with parameter M\_USART2 when u8\_com\_port is RS232\_COM2.

##### 10.32.8.7.2 daugwyrs232-ReceiveChar-LLR-004

Requirement ID: H398-LLD-GWY-FNC-3957

The function shall receive byte of data (reference of pu8\_dest) by calling 'UsartReceiveData' with parameter M\_UART4 when u8\_com\_port is RS232\_COM4.

##### 10.32.8.7.3 daugwyrs232-ReceiveChar-LLR-005

Requirement ID: H398-LLD-GWY-FNC-3958

The function shall receive M\_ZERO bytes of data to reference of pu8\_dest when u8\_com\_port is other than RS232\_COM2 and RS232\_COM4.

### 10.32.9 DisableRxIntr

Low Level Design Details about CSU DisableRxIntr will follow in the sub sections.

#### 10.32.9.1 Brief Description

The DisableRxIntr function disables the USARTx receive interrupt based on u8\_com\_port.

#### 10.32.9.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.32.9.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.32.9.4 Parameter list (Input/Output)

Inputs : T\_UINT8 u8\_com\_port - comport number.

Outputs : None

#### 10.32.9.5 Return Value

None

#### 10.32.9.6 Other CSUs called by this CSU

UsartItConfig

#### 10.32.9.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to DisableRxIntr

##### 10.32.9.7.1 daugwyrs232-DisableRxIntr-LLR-002

Requirement ID: H398-LLD-GWY-FNC-3968

The function shall disable the USART2 receive interrupt by calling 'UsartItConfig' with parameters M\_USART2, M\_USART\_IT\_RXNE and DISABLE when u8\_com\_port is RS232\_COM2.

##### 10.32.9.7.2 daugwyrs232-DisableRxIntr-LLR-004

Requirement ID: H398-LLD-GWY-FNC-3970

The function shall disable the UART4 receive interrupt by calling 'UsartItConfig' with parameters M\_UART4, M\_USART\_IT\_RXNE and DISABLE when u8\_com\_port is RS232\_COM4.

##### 10.32.9.7.3 daugwyrs232-DisableRxIntr-LLR-005

Requirement ID: H398-LLD-GWY-FNC-3971

The function shall do nothing when u8\_com\_port is other than RS232\_COM2 and RS232\_COM4.

### 10.32.10 Rs232TxIsr

Low Level Design Details about CSU Rs232TxIsr will follow in the sub sections.

#### 10.32.10.1 Brief Description

The Rs232TxIsr function implements interrupt service routine for RS232 transmit.

#### 10.32.10.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.32.10.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.32.10.4 Parameter list (Input/Output)

Inputs : T\_RS232\_DATA \*ps\_com\_data - pointer to the RS232 data structure,

T\_UINT8 u8\_com\_port - comport number.

Outputs : T\_RS232\_DATA \*ps\_com\_data - pointer to the RS232 data structure.

#### 10.32.10.5 Return Value

None

#### 10.32.10.6 Other CSUs called by this CSU

TransmitChar,

DisableTxIntr

#### 10.32.10.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to Rs232TxIsr

##### 10.32.10.7.1 daugwyrs232-Rs232TxIsr-LLR-001

Requirement ID: H398-LLD-GWY-FNC-3980

The function shall do the following when transmitter counter (u16\_tx\_cntr) of ps\_com\_data is not equal to M\_NULL

a) transmit a byte of data from the transmit buffer by calling 'TransmitChar' with parameters data address pu8\_tx\_head of ps\_com\_data and u8\_com\_port,

b) decrement transmitter counter (u16\_tx\_cntr) of ps\_com\_data by M\_ONE,

c) increment transmitter head (pu8\_tx\_head) of ps\_com\_data by M\_ONE,

d) assign base address of transmitter buffer in ps\_com\_data to transmitter head (pu8\_tx\_head) of ps\_com\_data when the transmitter head (pu8\_tx\_head) of ps\_com\_data is equal to end address of transmitter buffer with index M\_RS232\_TXBUF\_SIZE in ps\_com\_data, Otherwise do nothing.

##### 10.32.10.7.2 daugwyrs232-Rs232TxIsr-LLR-002

Requirement ID: H398-LLD-GWY-FNC-3981

The function shall disable transmit interrupt by calling 'DisableTxIntr' with parameter u8\_com\_port when transmitter counter (u16\_tx\_cntr) of ps\_com\_data equal to M\_NULL.

### 10.32.11 Rs232RxIsr

Low Level Design Details about CSU Rs232RxIsr will follow in the sub sections.

#### 10.32.11.1 Brief Description

The Rs232RxIsr function implements interrupt service routine for RS232 receive.

#### 10.32.11.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.32.11.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.32.11.4 Parameter list (Input/Output)

Inputs : T\_RS232\_DATA \*ps\_com\_data - pointer to the RS232 data structure,

T\_UINT8 u8\_com\_port - comport number.

Outputs : T\_RS232\_DATA \*ps\_com\_data - pointer to the RS232 data structure.

#### 10.32.11.5 Return Value

None

#### 10.32.11.6 Other CSUs called by this CSU

ReceiveChar,

DisableRxIntr

#### 10.32.11.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to Rs232RxIsr

##### 10.32.11.7.1 daugwyrs232-Rs232RxIsr-LLR-001

Requirement ID: H398-LLD-GWY-FNC-3990

The function shall do the following when receiver counter (u16\_rx\_cntr) of ps\_com\_data not equal to M\_RS232\_RXBUF\_SIZE

a) receive data from the receive buffer by calling 'ReceiveChar' with parameters pu8\_rx\_tail of ps\_com\_data and u8\_com\_port,

b) increment receiver counter (u16\_rx\_cntr) of ps\_com\_data by M\_ONE,

c) increment receiver tail (pu8\_rx\_tail) of ps\_com\_data by M\_ONE,

d) assign base address of receiver buffer in ps\_com\_data to receiver tail (pu8\_rx\_tail) of ps\_com\_data when the receiver tail (pu8\_rx\_tail) of ps\_com\_data is equal to end address of receiver buffer with index M\_RS232\_RXBUF\_SIZE in ps\_com\_data.

##### 10.32.11.7.2 daugwyrs232-Rs232RxIsr-LLR-002

Requirement ID: H398-LLD-GWY-FNC-3991

The function shall disable receive interrupt by calling 'DisableRxIntr' with parameter (u8\_com\_port) when receiver counter (u16\_rx\_cntr) of ps\_com\_data equal to M\_RS232\_RXBUF\_SIZE

### 10.32.12 COM2Intr

Low Level Design Details about CSU COM2Intr will follow in the sub sectins

#### 10.32.12.1 Brief Description

The COM2Intr function implements Interrupt service routine for COM PORT 2

#### 10.32.12.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY)

#### 10.32.12.3 List of global variables accessed and modified

Accessed : Lpbk\_tx\_com, Cm\_data

Modified : None

#### 10.32.12.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.32.12.5 Return Value

None

#### 10.32.12.6 Other CSUs called by this CSU

UsartGetItStatus,

Rs232RxIsr,

Rs232TxIsr,

CommGetChars,

RterrError,

InitComData,

RterrForever

#### 10.32.12.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to COM2Intr

##### 10.32.12.7.1 daugwyrs232-COM2Intr-LLR-001

Requirement ID: H398-LLD-GWY-FNC-4033

The function shall do the following when data is present in the hardware receive buffer (i.e., when function call 'UsartGetItStatus' with parameter Usart\_com2 and M\_USART\_IT\_RXNE return SET) and loopback transmit flag (Lpbk\_tx\_com with index RS232\_COM2) is TRUE

a) Call InitComData with parameter (as reference to the RS232 data with index RS232\_COM2)

b) Call Rs232RxIsr with parameters (with reference to the Rs232 data with index RS232\_COM2, RS232\_COM2)

##### 10.32.12.7.2 daugwyrs232-COM2Intr-LLR-002

Requirement ID: H398-LLD-GWY-FNC-4034

The function shall call Rs232RxIsr with parameters (with reference to the RS232 data with index RS232\_COM2 as parameters, RS232\_COM2) when data is present in the hardware receive buffer (i.e., when function call 'UsartGetItStatus' with parameter Usart\_com2 and M\_USART\_IT\_RXNE) and loopback transmit flag (Lpbk\_tx\_com with index as RS232\_COM2) is FALSE.

##### 10.32.12.7.3 daugwyrs232-COM2Intr-LLR-003

Requirement ID: H398-LLD-GWY-FNC-4035

The function shall perform the following

1. Sets the timeout to call the function CommGetChars with parameters as (reference of destination address, M\_ONE, M\_TIMEOUT0, RS232\_COM2) when checkup of error of Cm\_data is TRUE AND Lpbk\_tx\_com of index RS232\_COM2 is TRUE. Otherwise do nothing
2. call RterrError with parameters as (RS232\_LOOPBACK\_ERR, M\_TWO, RterrForever) and Set RS422\_loop\_back\_fail\_flag to TRUE,when Lpbk\_tx\_com of index RS232\_COM2 is equal to TRUE and destination is not equal to ‘2’.

3 .do nothing when function call 'UsartGetItStatus' with parameter M\_USART2 and M\_USART\_IT\_RXNE return other than SET) .

##### 10.32.12.7.4 daugwyrs232-COM2Intr-LLR-004

Requirement ID: H398-LLD-GWY-FNC-4036

The function shall call Rs232TxIsr with parameters as (reference to the RS232 data with index RS232\_COM2, RS232\_COM2) when hardware transmit buffer is empty (i.e., when function call 'UsartGetItStatus' with parameter Usart com2 and M\_USART\_IT\_TXE return SET), Otherwise do nothing.

### 10.32.13 InitCOM2

Low Level Design Details about CSU InitCOM2 will follow in the sub sections

#### 10.32.13.1 Brief Description

The InitCOM2 function initializes COM PORT 2.

#### 10.32.13.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.32.13.3 List of global variables accessed and modified

Accessed : Appdata\_appdata

Modified : None

#### 10.32.13.4 Parameter list (Input/Output)

Inputs : T\_INTR\_FN isr\_func - Reference to ISR for COM 2

Outputs : None

#### 10.32.13.5 Return Value

None

#### 10.32.13.6 Other CSUs called by this CS

InitComData,

UsartDeInit,

RccApb1PeriphClockCmd,

RccAhb1PeriphClockCmd,

IntrInstall,

GpioPinAFConfig,

GpioInit,

NvicInit,

UsartInit,

UsartItConfig,

UsartCmd.

#### 10.32.13.7 Description of list of LLRs allocate

The following section will list the LLRs allocated to InitCOM2

##### 10.32.13.7.1 daugwyrs232-InitCOM2-LLR-001

Requirement ID: H398-LLD-GWY-FNC-4045

The function shall initialize com port 2 RS232 data structure by calling 'InitComData' with reference to the RS232 data structure with index RS232\_COM2 as parameter

##### 10.32.13.7.2 daugwyrs232-InitCOM2-LLR-002

Requirement ID: H398-LLD-GWY-FNC-4046

The function shall deinitializes the USART2 peripheral registers to their default reset values by calling 'UsartDeInit' with parameter M\_USART2

##### 10.32.13.7.3 daugwyrs232-InitCOM2-LLR-003

Requirement ID: H398-LLD-GWY-FNC-4047

The function shall enable the USART2 peripheral clock by calling 'RccApb1PeriphClockCmd' with parameters M\_RCC\_APB1PERIPH\_USART2, ENABLE

##### 10.32.13.7.4 daugwyrs232-InitCOM2-LLR-004

Requirement ID: H398-LLD-GWY-FNC-4048

The function shall enable GPIO port A clock by calling 'RccAhb1PeriphClockCmd' with parameters M\_RCC\_AHB1PERIPH\_GPIOA and ENABLE

##### 10.32.13.7.5 daugwyrs232-InitCOM2-LLR-005

Requirement ID: H398-LLD-GWY-FNC-4049

The function shall load the reference to the isr\_func into the vector table for USART2 interrupt by calling 'IntrInstall' with parameters INTR\_USART\_2 and reference to isr\_func

##### 10.32.13.7.6 daugwyrs232-InitCOM2-LLR-006

Requirement ID: H398-LLD-GWY-FNC-4050

The function shall change the mapping of pins M\_GPIO\_PINSOURCE2 and M\_GPIO\_PINSOURCE3 of GPIO port A to USART2 Alternate Function mapping by calling 'GpioPinAFConfig' with parameters M\_GPIOA, source pins and M\_GPIO\_AF\_USART2.

##### 10.32.13.7.7 daugwyrs232-InitCOM2-LLR-007

Requirement ID: H398-LLD-GWY-FNC-4051

The function shall do the following,

a) Set gpio\_pin to M\_GPIOA\_RS232\_TX2,

b) Set GPIO port output speed register(gpio\_speed) with GPIO\_SPEED\_50MHZ,

c) Set GPIO port mode register(gpio\_mode) to GPIO\_MODE\_AF,

d) Set GPIO port output type register(gpio\_otype) to GPIO\_OTYPE\_PP and

e) Set GPIO port pull-up/pull-down register(gpio\_pupd) to GPIO\_PUPD\_NOPULL

f) Call 'GpioInit' with parameters M\_GPIOA and gpio init structure

##### 10.32.13.7.8 daugwyrs232-InitCOM2-LLR-008

Requirement ID: H398-LLD-GWY-FNC-4052

The function shall set gpio\_pin of gpio init structure to M\_GPIOA\_RS232\_RX2 and Call the 'GpioInit' with parameters M\_GPIOA and reference to gpio init structure.

##### 10.32.13.7.9 daugwyrs232-InitCOM2-LLR-009

Requirement ID: H398-LLD-GWY-FNC-4053

The function shall do the follwoing

a) nvic\_irq\_channel is set with USART2\_IRQN,

b) nvic\_irq\_channel\_preempt\_pri is set with M\_NVIC\_IRQ\_CHANNEL\_PREEMPT\_PRI9,

c) nvic\_irq\_channel\_subpriority is set with M\_NVIC\_IRQ\_CHANNEL\_SUBPRIORITY1 and

d) nvic\_irq\_channel\_cmd is set with ENABLE.

e) Call 'NvicInit' with reference to nvic init structure

##### 10.32.13.7.10 daugwyrs232-InitCOM2-LLR-010

Requirement ID: H398-LLD-GWY-FNC-4054

The function shall initialize USART2 peripheral as follows

a) usart\_baudrate is set with baud rate of com 2 configuration of Appdata\_appdata

b) usart\_wordlength is set with word length of com port 2,

c) usart\_stopbits is set with stop bit of com port 2,

d) usart\_parity is set with parity of com port 2,

e) usart\_hardware\_flowcontrol is set with M\_USART\_HARDWAREFLOWCONTROL\_NONE,

f) usart\_mode is set with bitwise OR M\_USART\_MODE\_RX, M\_USART\_MODE\_TX

g) Call 'UsartInit' with to with parameters (M\_USART2 , reference to uart init structure )

##### 10.32.13.7.11 daugwyrs232-InitCOM2-LLR-011

Requirement ID: H398-LLD-GWY-FNC-4055

The function shall enable the USART2 receive interrupt by calling 'UsartItConfig' with parameters M\_USART2, M\_USART\_IT\_RXNE and ENABLE

##### 10.32.13.7.12 daugwyrs232-InitCOM2-LLR-012

Requirement ID: H398-LLD-GWY-FNC-4056

The function shall enable the USART2 by calling 'UsartCmd' with parameters M\_USART2 and ENABLE

### 10.32.14 COM4Intr

Low Level Design Details about CSU COM4Intr will follow in the sub sections

#### 10.32.14.1 Brief Description

The COM4Intr function implements Interrupt service routine for COM PORT 4

#### 10.32.14.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.32.14.3 List of global variables accessed and modified

Accessed : Lpbk\_tx\_com

Modified : None

#### 10.32.14.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.32.14.5 Return Value

None

#### 10.32.14.6 Other CSUs called by this CSU

UsartGetItStatus,

Rs232RxIsr,

Rs232TxIsr,

InitComData

#### 10.32.14.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to COM4Intr.

##### 10.32.14.7.1 daugwyrs232-COM4Intr-LLR-001

Requirement ID: H398-LLD-GWY-FNC-4097

The function shall check for receive buffer full by calling 'UsartGetItStatus' with parameters Usart\_com4, M\_USART\_IT\_RXNE is equal to SET. When loopback transmit flag (Lpbk\_tx\_com with index as RS232\_COM4) is TRUE do the following

a) Call InitComData with parameter as with reference to the RS232 data with index RS232\_COM4.

b) get the data from receive buffer and store into RS232 data structure by calling 'Rs232RxIsr' with parameter as with reference to the RS232 data with index RS232\_COM4, RS232\_COM4

##### 10.32.14.7.2 daugwyrs232-COM4Intr-LLR-002

Requirement ID: H398-LLD-GWY-FNC-4098

The function shall check for receive buffer full by calling 'UsartGetItStatus' with parameters Usart\_com4, M\_USART\_IT\_RXNE is equal to SET. When the Lpbk\_tx\_com with index RS232\_COM4 is FALSE do the following

a) Get the data from receive buffer and store into RS232 data structure by calling 'Rs232RxIsr' with parameter as with reference to the As\_comdata with index RS232\_COM4, RS232\_COM4.

##### 10.32.14.7.3 daugwyrs232-COM4Intr-LLR-003

Requirement ID: H398-LLD-GWY-FNC-4099

The function shall check for receive buffer full by calling 'UsartGetItStatus' with parameters Usart\_com4, M\_USART\_IT\_RXNE is equal to SET. When the Lpbk\_tx\_com with index RS232\_COM4 is TRUE do the following

b)Set Lpbk\_tx\_com with index RS232\_COM4 is equal to TRUE.

- return of CommGetChars with parameters (address of u8 dest,M\_ONE, M\_TIMEOUT0 and RS232\_COM4) is set to time\_out.

- when u8 dest is not equal to four call RterrError with parameters RS232\_LOOPBACK\_ERR,M\_FOUR and RterrForever

- Set FALSE to Lpbk\_tx\_com with index RS232\_COM4

c) do nothing when Cm\_data.error.checkup AND Lpbk\_tx\_com with index RS232\_COM4 returns FALSE.

b) do nothing when 'UsartGetItStatus' with parameters Usart\_com4, M\_USART\_IT\_RXNE is not equal to SET.

##### 10.32.14.7.4 daugwyrs232-COM4Intr-LLR-004

Requirement ID: H398-LLD-GWY-FNC-6208

The function shall do the following.

Call 'Rs232TxIsr' with parameters (address of As\_comdata with index RS232\_COM4, RS232\_COM4). when the function 'UsartGetItStatus' with parameters Usart com4, M\_USART\_IT\_TXE is equal to SET, Otherwise do nothing

### 10.32.15 InitCOM4

Low Level Design Details about CSU InitCOM4 will follow in the sub sections.

#### 10.32.15.1 Brief Description

The InitCOM4 function initializes COM PORT 4.

#### 10.32.15.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY)

#### 10.32.15.3 List of global variables accessed and modified

Accessed : Appdata\_appdata

Modified : None

#### 10.32.15.4 Parameter list (Input/Output)

Inputs : T\_INTR\_FN isr\_func - Reference to ISR for COM 4

Outputs : None

#### 10.32.15.5 Return Value

None

#### 10.32.15.6 Other CSUs called by this CSU

InitComData,

UsartDeInit,

RccApb1PeriphClockCmd,

RccAhb1PeriphClockCmd,

IntrInstall,

GpioPinAFConfig,

GpioInit,

NvicInit,

UsartInit,

UsartItConfig,

UsartCmd

#### 10.32.15.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to InitCOM4

##### 10.32.15.7.1 daugwyrs232-InitCOM4-LLR-001

Requirement ID: H398-LLD-GWY-FNC-4108

The function shall initialize com port 4 RS232 data structure by calling 'InitComData' with reference to the RS232 data structure with index RS232\_COM4 as parameter.

##### 10.32.15.7.2 daugwyrs232-InitCOM4-LLR-002

Requirement ID: H398-LLD-GWY-FNC-4109

The function shall de-initializes the USART4 peripheral registers to their default reset values by calling 'UsartDeInit' with parameter M\_UART4

##### 10.32.15.7.3 daugwyrs232-InitCOM4-LLR-003

Requirement ID: H398-LLD-GWY-FNC-4110

The function shall enable the UART4 peripheral clock by calling 'RccApb1PeriphClockCmd' with parameters M\_RCC\_APB1PERIPH\_UART4, ENABLE.

##### 10.32.15.7.4 daugwyrs232-InitCOM4-LLR-004

Requirement ID: H398-LLD-GWY-FNC-4111

The function shall enable GPIO port A clock by calling 'RccAhb1PeriphClockCmd' with parameters M\_RCC\_AHB1PERIPH\_GPIOA and ENABLE.

##### 10.32.15.7.5 daugwyrs232-InitCOM4-LLR-005

Requirement ID: H398-LLD-GWY-FNC-4112

The function shall load the reference to the isr\_func into the vector table for UART4 interrupt by calling 'IntrInstall' with parameters INTR\_UART\_4 and isr\_func.

##### 10.32.15.7.6 daugwyrs232-InitCOM4-LLR-006

Requirement ID: H398-LLD-GWY-FNC-4113

The function shall change the mapping of pins M\_GPIO\_PINSOURCE0 and M\_GPIO\_PINSOURCE1 of GPIO port A to UART4 Alternate Function mapping by calling 'GpioPinAFConfig' with parameters M\_GPIOA, source pins and M\_GPIO\_AF\_UART4.

##### 10.32.15.7.7 daugwyrs232-InitCOM4-LLR-007

Requirement ID: H398-LLD-GWY-FNC-4114

The function shall do the following,

a)Set gpio\_pin to M\_GPIOA\_RS232\_TX4,

b)Set GPIO port output speed register(gpio\_speed) with GPIO\_SPEED\_50MHZ,

c)Set GPIO port mode register(gpio\_mode) to GPIO\_MODE\_AF,

d)Set GPIO port output type register(gpio\_otype) to GPIO\_OTYPE\_PP and

e)Set GPIO port pull-up/pull-down register(gpio\_pupd) to GPIO\_PUPD\_NOPULL.

f) Call'GpioInit' with parameters M\_GPIOA and reference to gpio init structure

##### 10.32.15.7.8 daugwyrs232-InitCOM4-LLR-008

Requirement ID: H398-LLD-GWY-FNC-4115

The function shall set gpio\_pin of gpio init structure to M\_GPIOA\_RS232\_RX4 and Call 'GpioInit' with parameters M\_GPIOA and reference to gpio init structure.

##### 10.32.15.7.9 daugwyrs232-InitCOM4-LLR-009

Requirement ID: H398-LLD-GWY-FNC-4116

The function shall initialize NVIC peripheral as follows

a) nvic\_irq\_channel is set with UART4\_IRQN,

b) nvic\_irq\_channel\_preempt\_pri is set with M\_NVIC\_IRQ\_CHANNEL\_PREEMPT\_PRI9,

c) nvic\_irq\_channel\_subpriority is set with M\_NVIC\_IRQ\_CHANNEL\_SUBPRIORITY5 and

d) nvic\_irq\_channel\_cmd is set with ENABLE.

e) Call 'NvicInit' with reference to nvic init structure

##### 10.32.15.7.10 daugwyrs232-InitCOM4-LLR-010

Requirement ID: H398-LLD-GWY-FNC-4117

The function shall initialize UART4 peripheral as follows

a) usart\_baudrate is set with baud rate of com 4 configuration of Appdata\_appdata

b) usart\_wordlength is set with word length of comport 4,

c) usart\_stopbits is set with stop bit of comport 4,

d) usart\_parity is set with parity of comport 4,

e) usart\_hardware\_flowcontrol is set with M\_USART\_HARDWAREFLOWCONTROL\_NONE,

f) usart\_mode is set with bitwise OR M\_USART\_MODE\_RX, M\_USART\_MODE\_TX.

g) Call 'UsartInit' with M\_UART4, reference to usart init structure.

##### 10.32.15.7.11 daugwyrs232-InitCOM4-LLR-011

Requirement ID: H398-LLD-GWY-FNC-4118

The function shall enable the UART4 receive interrupt by calling 'UsartItConfig' with parameters M\_UART4, M\_USART\_IT\_RXNE and ENABLE.

##### 10.32.15.7.12 daugwyrs232-InitCOM4-LLR-012

Requirement ID: H398-LLD-GWY-FNC-4119

The function shall enable the UART4 by calling 'UsartCmd' with parameters M\_UART4 and ENABLE.

### 10.32.16 Rs232Init

Low Level Design Details about CSU Rs232Init will follow in the sub sections.

#### 10.32.16.1 Brief Description

The Rs232Init function initializes com port 1, 2, 3, 4, 6, Uas\_message1 for right PFD and Uas\_message2 for EICAS.

#### 10.32.16.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.32.16.3 List of global variables accessed and modified

Accessed : Appdata\_appdata

Modified : None

#### 10.32.16.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.32.16.5 Return Value

None

#### 10.32.16.6 Other CSUs called by this CSU

InitCOM2

InitCOM4

COM2Intr

COM4Intr

#### 10.32.16.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to Rs232Init

##### 10.32.16.7.1 daugwyrs232-Rs232Init-LLR-001

Requirement ID: H398-LLD-GWY-FNC-4128

The function shall loop through M\_ZERO to RS232\_MAX\_COM minus 1 increment the loopcounter by one and do the following when func\_state of comms with index as index i of Appdata\_appdata is ENABLED

- Set word length for comport initialization structure to M\_USART\_WORDLENGTH\_8B when the data\_bits of comms of Appdata\_appdata is equal to M\_WORDLENGTH\_8, otherwise set word length for comport 1 to M\_USART\_WORDLENGTH\_9B.

##### 10.32.16.7.2 daugwyrs232-Rs232Init-LLR-002

Requirement ID: H398-LLD-GWY-FNC-4129

The function shall loop through M\_ZERO to RS232\_MAX\_COM minus 1 increment the loopcounter by one and do the following when func\_state of comms with index as index i of Appdata\_appdata is ENABLED

- Set the parity for comport 1 to M\_USART\_PARITY\_NO when the parity of comms of Appdata\_appdata is M\_NO\_PARITY

- Set the parity for comport 1 to M\_USART\_PARITY\_EVEN when the parity of comms of Appdata\_appdata is M\_EVEN\_PARITY

- Set the parity for comport 1 to M\_USART\_PARITY\_ODD when the parity of comms of Appdata\_appdata is M\_ODD\_PARITY

- otherwise do nothing.

##### 10.32.16.7.3 daugwyrs232-Rs232Init-LLR-003

Requirement ID: H398-LLD-GWY-FNC-4130

The function shall loop through M\_ZERO to RS232\_MAX\_COM minus 1 increment the loopcounter by one and do the following when func\_state of comms with index as index i of Appdata\_appdata is ENABLED.

a) Set stopbits of com\_port with loop index of Comport\_init to M\_USART\_STOPBITS\_1, when ((M\_ONE\_STOP\_BIT is equal to stop of comms with loop index of Appdata\_appdata) by calling M\_FP\_COMARE)

b) Set stopbits of com\_port with loop index of Comport\_init to M\_USART\_STOPBITS\_0\_P\_5, when ((M\_POINT5\_STOP\_BIT is equal to stop of comms with loop index of Appdata\_appdata) by calling M\_FP\_COMARE)

c) Set stopbits of com\_port with loop index of Comport\_init to M\_USART\_STOPBITS\_2, when ((M\_TWO\_STOP\_BIT is equal to stop of comms with loop index of Appdata\_appdata) by calling M\_FP\_COMARE)

d) Set stopbits of com\_port with loop index of Comport\_init to M\_USART\_STOPBITS\_1\_P\_5, when none of the above conditions are satisfied.

##### 10.32.16.7.4 daugwyrs232-Rs232Init-LLR-005

Requirement ID: H398-LLD-GWY-FNC-4132

The function shall loop through M\_ZERO to RS232\_MAX\_COM minus 1 increment the loopcounter by one and do the following when type of Com port is RS232\_COM2 and func\_state of Com with index as index i of Appdata\_appdata is ENABLED.

a) store M\_USART2 to usart address for com port 2

b) call InitCOM2 with parameter COM2Intr

##### 10.32.16.7.5 daugwyrs232-Rs232Init-LLR-007

Requirement ID: H398-LLD-GWY-FNC-4134

The function shall loop through M\_ZERO to RS232\_MAX\_COM minus 1 increment the loopcounter by one and do the following when type of Com port RS232\_COM4 and func\_state of Com with index as index i of Appdata\_appdata is ENABLED.

a) store M\_USART4 to usart address for com port 4

b) call InitCOM4 with parameter COM4Intr

##### 10.32.16.7.6 daugwyrs232-Rs232Init-LLR-008

Requirement ID: H398-LLD-GWY-FNC-4135

The function shall loop through M\_ZERO to RS232\_MAX\_COM minus 1 increment the loopcounter by one and does nothing when type of Com port is other than RS232\_COM1, RS232\_COM2, RS232\_COM3 and RS232\_COM4 and func\_state of Com with index as index i of Appdata\_appdata is ENABLED.

## 10.33 daugwyrterr

The daugwyrterr CSC provides routines to display run time errors

### 10.33.1 RterrError

Low Level Design Details about CSU RterrError will follow in the sub sections.

#### 10.33.1.1 Brief Description

The RterrError function reads error type and flashes code accordingly

#### 10.33.1.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY)

#### 10.33.1.3 List of global variables accessed and modified

Accessed : None

Modified : Rterr\_new

#### 10.33.1.4 Parameter list (Input/Output)

Inputs : T\_ERROR\_TYPE ucerr error type

T\_SINT32 sub - sub field to distingush the error type

T\_BOOL (\*forever)() pointer to function RterrForever

(NOTE: The function pointer 'forever' calls the function RterrForever, which is used to implement the loop forever functionality)

Outputs : None

#### 10.33.1.5 Return Value

None

#### 10.33.1.6 Other CSUs called by this CSU

WdogKickWatchDog

UpdateNVRam

WriteErrorToFlash

#### 10.33.1.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to RterrError.

##### 10.33.1.7.1 daugwyrterr-RterrError-LLR-001

Requirement ID: H398-LLD-GWY-FNC-4145

The function shall do the following

- disable all interrupts

- call functions WriteErrorToFlash,UpdateNVRam with parameters ucerr, sub.

-enable all interrupts and return from the function when ucerr is greater than M\_TWELVE.

##### 10.33.1.7.2 daugwyrterr-RterrError-LLR-002

Requirement ID: H398-LLD-GWY-FNC-4146

The function shall disable the counter by resetting the 0th bit of the timer 7 control register 1(M\_TIM\_CR1\_CEN).

##### 10.33.1.7.3 daugwyrterr-RterrError-LLR-003

Requirement ID: H398-LLD-GWY-FNC-4147

The function shall set ahb1enr of M\_RCC by performing BITWISE OR with M\_RCC\_AHB1PERIPH\_GPIOA.

##### 10.33.1.7.4 daugwyrterr-RterrError-LLR-004

Requirement ID: H398-LLD-GWY-FNC-4148

The function shall set ahb1enr of M\_RCC by performing BITWISE OR with M\_RCC\_AHB1PERIPH\_GPIOC.

##### 10.33.1.7.5 daugwyrterr-RterrError-LLR-005

Requirement ID: H398-LLD-GWY-FNC-4149

The function shall configure the GPIO port A output pins as follows

- Mode register (moder) set to M\_GPIOA\_MODER\_SET

- Output Speed register (ospeedr) set to M\_GPIOA\_OSPEEDR\_SET

- Output type register (otyper) set to M\_GPIOA\_OTYPER\_SET

- Pull-up/pull-down register (pupdr) set to M\_GPIOA\_PUPDR\_SET.

##### 10.33.1.7.6 daugwyrterr-RterrError-LLR-006

Requirement ID: H398-LLD-GWY-FNC-4150

The function shall configure the GPIO port C output pins as follows

- Mode register (moder) set to M\_GPIOC\_MODER\_SET

- Output Speed register (ospeedr) set to M\_GPIOC\_OSPEEDR\_SET

- Output type register (otyper) set to M\_GPIOC\_OTYPER\_SET

- Pull-up/pull-down register (pupdr) set to M\_GPIOC\_PUPDR\_SET.

##### 10.33.1.7.7 daugwyrterr-RterrError-LLR-007

Requirement ID: H398-LLD-GWY-FNC-4151

The function shall set warn lines High to Disable Red LEDs by setting the corresponding bits for pin 14 (M\_GPIOC\_GREEN\_LED) and pin 15 (M\_GPIOC\_RED\_LED) in the GPIO port C bit set/reset high register (bsrrh).

##### 10.33.1.7.8 daugwyrterr-RterrError-LLR-008

Requirement ID: H398-LLD-GWY-FNC-4152

The function shall loop forever and perform the following operations when the error type is FLIGHT\_APP\_CRC\_ERR

a) call GpioSetBits with reference to GPIO port B (M\_GPIOB) and GPIO pin 2 (M\_GPIO\_PIN\_2) as parameters to switch on the heartbeat LED.

b) Loop till M\_DLY\_1550000 times to provide delay and call WdogKickWatchDog to reload watchdog counter in each loop.

c) Set the GPIO port C bit set/reset low register(bsrrl) to M\_GPIOC\_RED\_LED (pin 15).

d) Loop till M\_DLY\_1550000 times and to provide delay and call WdogKickWatchDog to reload watchdog counter in each loop.

e) Set the GPIO port C bit set/reset high register(bsrrh) to M\_GPIOC\_RED\_LED (pin 15).

f) Loop till M\_DLY\_16800000 times and call WdogKickWatchDog to reload watchdog counter in each loop.

##### 10.33.1.7.9 daugwyrterr-RterrError-LLR-009

Requirement ID: H398-LLD-GWY-FNC-4153

The function shall loop forever and perform the following operations when the error type is MCD\_ERR

a) call GpioSetBits with reference to GPIO port B (M\_GPIOB) and GPIO pin 2 (M\_GPIO\_PIN\_2) as parameters to switch on the heartbeat LED.

b) Loop M\_DLY\_3 times and do the following in each loop

i) Loop M\_DLY\_1550000 times to provide delay and call WdogKickWatchDog to reload watchdog counter in each loop.

ii) Set the GPIO port C bit set/reset low register(bsrrl) to M\_GPIOC\_RED\_LED (pin 15).

iii) Loop M\_DLY\_1550000 times to provide delay and call WdogKickWatchDog to reload watchdog counter in each loop.

iv) Set the GPIO port C bit set/reset high register(bsrrh) to M\_GPIOC\_RED\_LED (pin 15).

c) Loop M\_DLY\_16800000 times and call WdogKickWatchDog to reload watchdog counter in each loop.

##### 10.33.1.7.10 daugwyrterr-RterrError-LLR-017

Requirement ID: H398-LLD-GWY-FNC-4154

The function shall loop forever and perform the following operations when the error type is A825\_INTER\_PBIT\_ERR.

a) call GpioSetBits with reference to GPIO port B (M\_GPIOB) and GPIO pin 2 (M\_GPIO\_PIN\_2) as parameters to switch on the heartbeat LED.

b) Loop M\_DLY\_7 times and do the following in each loop

i) Loop M\_DLY\_1550000 times to provide delay and call WdogKickWatchDog to reload watchdog counter in each loop.

ii) Set the GPIO port C bit set/reset low register(bsrrl) to M\_GPIOC\_RED\_LED (pin 15).

iii) Loop M\_DLY\_1550000 times to provide delay and call WdogKickWatchDog to reload watchdog counter in each loop.

iv) Set the GPIO port C bit set/reset high register(bsrrh) to M\_GPIOC\_RED\_LED (pin 15).

c) Loop M\_DLY\_16800000 times and call WdogKickWatchDog to reload watchdog counter in each loop.

##### 10.33.1.7.11 daugwyrterr-RterrError-LLR-011

Requirement ID: H398-LLD-GWY-FNC-4155

The function shall loop forever and perform the following operations when the error type is BIT\_ERR

a) call GpioSetBits with reference to GPIO port B (M\_GPIOB) and GPIO pin 2 (M\_GPIO\_PIN\_2) as parameters to switch on the heartbeat LED.

b) Loop M\_DLY\_8 times and do the following in each loop

i) Loop M\_DLY\_1550000 times to provide delay and call WdogKickWatchDog to reload watchdog counter in each loop.

ii) Set the GPIO port C bit set/reset low register(bsrrl) to M\_GPIOC\_RED\_LED (pin 15).

iii) Loop M\_DLY\_1550000 times to provide delay and call WdogKickWatchDog to reload watchdog counter in each loop.

iv) Set the GPIO port C bit set/reset high register(bsrrh) to M\_GPIOC\_RED\_LED (pin 15).

c) Loop M\_DLY\_16800000 times and call WdogKickWatchDog to reload watchdog counter in each loop.

##### 10.33.1.7.12 daugwyrterr-RterrError-LLR-012

Requirement ID: H398-LLD-GWY-FNC-4156

The function shall loop forever and perform the following operations when the error type is INTR\_ERR

a) call GpioSetBits with reference to GPIO port B (M\_GPIOB) and GPIO pin 2 (M\_GPIO\_PIN\_2) as parameters to switch on the heartbeat LED.

b) Loop M\_DLY\_1550000 times to provide delay and call WdogKickWatchDog to reload watchdog counter in each loop.

c) Set the GPIO port C bit set/reset low register(bsrrl) to M\_GPIOC\_RED\_LED (pin 15).

d) Loop M\_DLY\_1550000 times to provide delay and call WdogKickWatchDog to reload watchdog counter in each loop.

e) Set the GPIO port C bit set/reset high register(bsrrh) to M\_GPIOC\_RED\_LED (pin 15).

##### 10.33.1.7.13 daugwyrterr-RterrError-LLR-013

Requirement ID: H398-LLD-GWY-FNC-4157

The function shall loop forever and perform the following operations when the error type is MEMMORY\_ERR

a) call GpioSetBits with reference to GPIO port B (M\_GPIOB) and GPIO pin 2 (M\_GPIO\_PIN\_2) as parameters to on the heartbeat LED.

b) Loop M\_DLY\_10 times and do the following in each loop

i) Loop M\_DLY\_1550000 times to provide delay and call WdogKickWatchDog to reload watchdog counter in each loop.

ii) Set the GPIO port C bit set/reset low register(bsrrl) to M\_GPIOC\_RED\_LED (pin 15).

iii) Loop M\_DLY\_1550000 times to provide delay and call WdogKickWatchDog to reload watchdog counter in each loop.

iv) Set the GPIO port C bit set/reset high register(bsrrh) to M\_GPIOC\_RED\_LED (pin 15).

1. Loop M\_DLY\_16800000 times and call WdogKickWatchDog to reload watchdog counter in each loop.

##### 10.33.1.7.14 daugwyrterr-RterrError-LLR-010

Requirement ID: H398-LLD-GWY-FNC-4161

The function shall loop forever and perform the following operations when the error type is NVM\_DATA\_CRC\_ERR.

a) call GpioSetBits with reference to GPIO port B (M\_GPIOB) and GPIO pin 2 (M\_GPIO\_PIN\_2) as parameters to switch on the heartbeat LED.

b) Loop M\_DLY\_11 times and do the following in each loop

i) Loop M\_DLY\_1550000 times to provide delay and call WdogKickWatchDog to reload watchdog counter in each loop.

ii) Set the GPIO port C bit set/reset low register(bsrrl) to M\_GPIOC\_RED\_LED (pin 15).

iii) Loop M\_DLY\_1550000 times to provide delay and call WdogKickWatchDog to reload watchdog counter in each loop.

iv) Set the GPIO port C bit set/reset high register(bsrrh) to M\_GPIOC\_RED\_LED (pin 15).

c) Loop M\_DLY\_16800000 times and call WdogKickWatchDog to reload watchdog counter in each loop.

##### 10.33.1.7.15 daugwyrterr-RterrError-LLR-020

Requirement ID: H398-LLD-GWY-FNC-6385

The function shall loop forever and perform the following operations when the error type is CMU\_ERR.

a) call GpioSetBits with reference to GPIO port B (M\_GPIOB) and GPIO pin 2 (M\_GPIO\_PIN\_2) as parameters to switch on the heartbeat LED.

b) Loop M\_DLY\_12 times and do the following in each loop

i) Loop M\_DLY\_1550000 times to provide delay and call WdogKickWatchDog to reload watchdog counter in each loop.

ii) Set the GPIO port C bit set/reset low register(bsrrl) to M\_GPIOC\_RED\_LED (pin 15).

iii) Loop M\_DLY\_1550000 times to provide delay and call WdogKickWatchDog to reload watchdog counter in each loop.

iv) Set the GPIO port C bit set/reset high register(bsrrh) to M\_GPIOC\_RED\_LED (pin 15).

c) Loop M\_DLY\_16800000 times and call WdogKickWatchDog to reload watchdog counter in each loop.

10.33.1.7.16 daugwyrterr-RterrError-LLR-019

Requirement ID: H398-LLD-GWY-FNC-4163

The function shall loop forever and perform the following operations when the error type is other than FLIGHT\_APP\_CRC\_ERR, MCD\_ERR, BIT\_ERR, INTR\_ERR, MEMMORY\_ERR, A825\_INTER\_PBIT\_ERR,NVM\_DATA\_CRC\_ERR,CMU\_ERR

1. call GpioResetBits with reference to GPIO port B (M\_GPIOB) and GPIO pin 2 (M\_GPIO\_PIN\_2) as parameters to switch off the heartbeat LED.
2. Loop M\_DLY\_1550000 times to provide delay and call WdogKickWatchDog to reload watchdog counter in each loop.
3. Set bsrrl of M\_GPIOC to M\_GPIOC\_RED\_LED (pin 15).
4. Loop M\_DLY\_1550000 times to provide delay and call WdogKickWatchDog to reload watchdog counter in each loop.
5. Set bsrrh of M\_GPIOC to M\_GPIOC\_RED\_LED (pin 15).

### 10.33.2 RterrForever

Low Level Design Details about CSU RterrForever will follow in the sub sections.

#### 10.33.2.1 Brief Description

The RterrForever function always returns TRUE.

#### 10.33.2.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.33.2.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.33.2.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.33.2.5 Return Value

T\_BOOL - returns TRUE

#### 10.33.2.6 Other CSUs called by this CSU

None

#### 10.33.2.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to RterrForever

##### 10.33.2.7.1 daugwyrterr-RterrForever-LLR-001

Requirement ID: H398-LLD-GWY-FNC-4172

The function shall return TRUE

### 10.33.3 CheckValidErrorFlash

Low Level Design Details about CSU CheckValidErrorFlash will follow in the sub sections.

#### 10.33.3.1 Brief Description

CheckValidErrorFlash function checks Error Flash for validity

#### 10.33.3.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD(H398-003-012-GWY).

#### 10.33.3.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.33.3.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.33.3.5 Return Value

T\_BOOL - return validity

#### 10.33.3.6 Other CSUs called by this CSU

None

#### 10.33.3.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to CheckValidErrorFlash.

##### 10.33.3.7.1 daugwyrterr-CheckValidErrorFlash-LLR-001

Requirement ID: H398-LLD-GWY-FNC-4181

The function shall set CRC Control register(cr of M\_CRC) with M\_CRC\_CR\_VALUE\_BIT\_ONE.

##### 10.33.3.7.2 daugwyrterr-CheckValidErrorFlash-LLR-002

Requirement ID: H398-LLD-GWY-FNC-4182

The function shall store the error log in CRC Data register, for all error stored in Flash (M\_MEMMAP\_FIRST\_ERR\_REPORT\_ADDR) from M\_ZERO to unit size of T\_RTERR\_REPORTSTRUCT - M\_ONE

(Note: unit size of T\_RTERR\_REPORTSTRUCT is caluculated as ((size of(T\_RTERR\_REPORTSTRUCT) / M\_FOUR) - M\_ONE);)

##### 10.33.3.7.3 daugwyrterr-CheckValidErrorFlash-LLR-003

Requirement ID: H398-LLD-GWY-FNC-4183

The function shall store CRC Data register value in check sum variable.

##### 10.33.3.7.4 daugwyrterr-CheckValidErrorFlash-LLR-004

Requirement ID: H398-LLD-GWY-FNC-4184

The function shall return FALSE when calculated check sum is not equal to CRC in first error report address of Flash(M\_MEMMAP\_FIRST\_ERR\_REPORT\_ADDR).

### 10.33.4 UpdateNVRam

Low Level Design Details about CSU UpdateNVRam will follow in the sub sections.

#### 10.33.4.1 Brief Description

UpdateNVRam function checks for update NVRAM.

#### 10.33.4.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD(H398-003-012-GWY).

#### 10.33.4.3 List of global variables accessed and modified

Accessed : xram\_data\_buffer,Logic\_currenttime

Modified : xram\_data\_buffer

#### 10.33.4.4 Parameter list (Input/Output)

Inputs : T\_ERROR\_TYPE ucerr – error type

T\_SINT32 sub - sub field to represent XRAM error status

Outputs : None

#### 10.33.4.5 Return Value

None

#### 10.33.4.6 Other CSUs called by this CSU

None

#### 10.33.4.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to UpdateNVRam.

##### 10.33.4.7.1 daugwyrterr-UpdateNVRam-LLR-001

Requirement ID: H398-LLD-GWY-FNC-4193

The function shall loop for M\_TWENTY and perform the following

When M\_ZERO is equal to error count of Xram\_data\_buffer bitwise OR with sub id of error item is equal to sub.

1. Increment the number of faults of Xram\_data\_buffer when the M\_ZERO is equal to error count of Xram\_data\_buffer.
2. Error code of faults of Xram\_data\_buffer to error type
3. Sub id(sub\_id) of faults of Xram\_data\_buffer to sub
4. Error count increment of faults of Xram\_data\_buffer.

### 10.33.5 WriteErrorToFlash

Low Level Design Details about CSU WriteErrorToFlash will follow in the sub sections.

#### 10.33.5.1 Brief Description

WriteErrorToFlash function writes the Error to Flash.

#### 10.33.5.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD(H398-003-012-GWY)

#### 10.33.5.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.33.5.4 Parameter list (Input/Output)

Inputs : T\_SINT32 ErrCode - error number

T\_SINT32 ErrSubNum - Sub error number

Outputs : None

#### 10.33.5.5 Return Value

None.

#### 10.33.5.6 Other CSUs called by this CSU

CheckValidErrorFlash

EraseErrSection

ProgramWord

#### 10.33.5.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to WriteErrorToFlash

##### 10.33.5.7.1 daugwyrterr-WriteErrorToFlash-LLR-001

Requirement ID: H398-LLD-GWY-FNC-4202

The function shall set IWDG Key register(kr of M\_IWDG) with M\_KR\_REGISTER\_BIT\_MASK

and the watchdog reset is prevented.

##### 10.33.5.7.2 daugwyrterr-WriteErrorToFlash-LLR-002

Requirement ID: H398-LLD-GWY-FNC-4203

The function shall do the following.

A temporary report buffer with index to M\_MEMMAP\_FIRST\_ERR\_REPORT\_ADDR with index from M\_ZERO to size of T\_RTERR\_REPORT\_STRUCT, when CheckValidErrorFlash function is not of M\_ZERO.

##### 10.33.5.7.3 daugwyrterr-WriteErrorToFlash-LLR-003

Requirement ID: H398-LLD-GWY-FNC-4204

The function shall reset all members (except errorcode, which is set to M\_MINUS\_ONE) of temporary report structure, when CheckValidErrorFlash is M\_ZERO.

##### 10.33.5.7.4 daugwyrterr-WriteErrorToFlash-LLR-004

Requirement ID: H398-LLD-GWY-FNC-4205

The function shall increment the total count of temporary report structure by one and

for all the error list the function shall do the following when error code of temporary

report structure is equal to M\_MINUS\_ONE

-store error code of temporary report structure with ErrCode

-store sub id of temporary report structure with ErrSubNum

-store error count of temporary report structure respectively with M\_ONE

##### 10.33.5.7.5 daugwyrterr-WriteErrorToFlash-LLR-005

Requirement ID: H398-LLD-GWY-FNC-4206

For all the error list, the function shall increment error count of temporary report structure by one,

when following conditions are satisfied:

- error code of temporary report structure is equal to ErrCode

- error code of sub id of temporary report structure is equal to ErrSubNum

respectively.

- increment the error count of temporary report structure when the error count of temporary report structure is less than M\_ERROR\_COUNT\_MAX\_VALUE.

##### 10.33.5.7.6 daugwyrterr-WriteErrorToFlash-LLR-006

Requirement ID: H398-LLD-GWY-FNC-4207

The function shall set CRC Control register(cr of M\_CRC) with M\_CRC\_CR\_VALUE\_BIT\_ONE

##### 10.33.5.7.7 daugwyrterr-WriteErrorToFlash-LLR-007

Requirement ID: H398-LLD-GWY-FNC-4208

The function shall store the error log in Flash in CRC Data register from M\_ZERO to unit size of T\_RTERR\_REPORT\_STRUCT- M\_ONE.

(Note: unit size of T\_RTERR\_REPORT\_STRUCT is caluculated as ((size of(T\_RTERR\_REPORT\_STRUCT) / M\_FOUR) - M\_ONE);)

##### 10.33.5.7.8 daugwyrterr-WriteErrorToFlash-LLR-008

Requirement ID: H398-LLD-GWY-FNC-4209

The function shall store CRC Data register value in local check sum variable and stores these local check sum value to crc value of temporary report structure.

##### 10.33.5.7.9 daugwyrterr-WriteErrorToFlash-LLR-009

Requirement ID: H398-LLD-GWY-FNC-4210

The function shall configure FLASH key register 1 and 2 with M\_FLASH\_KEY1

and M\_FLASH\_KEY2, to unlock FLASH control register, when cr of M\_FLASH bitwise AND with M\_FLASH\_CR\_LOCK control register is not of RESET.

##### 10.33.5.7.10 daugwyrterr-WriteErrorToFlash-LLR-010

Requirement ID: H398-LLD-GWY-FNC-4211

The function shall configure FLASH status register with M\_FLASH\_STATUS\_REGISTER\_VAL

##### 10.33.5.7.11 daugwyrterr-WriteErrorToFlash-LLR-011

Requirement ID: H398-LLD-GWY-FNC-4212

The function shall set the destination address as M\_MEMMAP\_FIRST\_ERR\_REPORT\_ADDR and register pointer to M\_MEMMAP\_FIRST\_ERR\_REGISTERS and copy all contents from index M\_ZERO to M\_CALIBRATION\_DATA\_COUNT of the register pointer to register buffer

##### 10.33.5.7.12 daugwyrterr-WriteErrorToFlash-LLR-012

Requirement ID: H398-LLD-GWY-FNC-4213

The function shall return with reference to temporary report structure when erasing the Error Section in Memory (EraseErrSection()) is equal to M\_ZERO.

##### 10.33.5.7.13 daugwyrterr-WriteErrorToFlash-LLR-013

Requirement ID: H398-LLD-GWY-FNC-4214

The function shall stop programming the data, when return value of ProgramWord called with parameters local destination(M\_MEMMAP\_FIRST\_ERR\_REGISTERS) and source pointer(address of register buffer) is equal to M\_ZERO, for calibration data from M\_ZERO to product of (two hundred (M\_CALIBRATION\_DATA\_COUNT) and four (M\_FOUR)).

##### 10.33.5.7.14 daugwyrterr-WriteErrorToFlash-LLR-014

Requirement ID: H398-LLD-GWY-FNC-4215

The function shall increment the local destination by M\_FOUR and source pointer by M\_ONE for calibration data from M\_ZERO to product of (two hundred (M\_CALIBRATION\_DATA\_COUNT) and four (M\_FOUR))

##### 10.33.5.7.15 daugwyrterr-WriteErrorToFlash-LLR-015

Requirement ID: H398-LLD-GWY-FNC-4216

The function shall stop programming the data,

when return value of ProgramWord called with parameters local destination(M\_MEMMAP\_FIRST\_ERR\_REPORT\_ADDR) and source pointer(address of temporary report structure) is FALSE for calibration data from M\_ZERO to size of temporary report structure

##### 10.33.5.7.16 daugwyrterr-WriteErrorToFlash-LLR-016

Requirement ID: H398-LLD-GWY-FNC-4217

The function shall increment the local destination by M\_FOUR and source pointer by M\_ONE for calibration data from M\_ZERO size of temporary report structure.

##### 10.33.5.7.17 daugwyrterr-WriteErrorToFlash-LLR-017

Requirement ID: H398-LLD-GWY-FNC-4218

The function shall lock the FLASH control register once the Flash Memory operations are done.

### 10.33.6 EraseErrSection

Low Level Design Details about CSU EraseErrSection will follow in the sub sections.

#### 10.33.6.1 Brief Description

EraseErrSection function writes Error to flash.

#### 10.33.6.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD(H398-003-012-GWY).

#### 10.33.6.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.33.6.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.33.6.5 Return Value

T\_BOOL - Return the Erase Status

#### 10.33.6.6 Other CSUs called by this CSU

None

#### 10.33.6.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to EraseErrSection.

##### 10.33.6.7.1 daugwyrterr-EraseErrSection-LLR-001

Requirement ID: H398-LLD-GWY-FNC-4227

The function shall wait till loop counter is less than M\_PROGRAM\_WORD\_MAX\_COUNT AND ongoing programs

in FLASH status register is not zero.

##### 10.33.6.7.2 daugwyrterr-EraseErrSection-LLR-002

Requirement ID: H398-LLD-GWY-FNC-4228

The function shall erase the sector, configure and start Flash memory operations

by setting FLASH\_CR register as below when loop counter is less than M\_PROGRAM\_WORD\_MAX\_COUNT:

- reset Flash CR register by M\_CR\_PSIZE\_MASK, to select program size x8

- set Flash CR register with temporary size, to select program size x32

- reset Flash CR register sector number bits to erase sector0 (M\_ERASE\_OPERATION\_MASK)

- activate Sector Erase for SECTOR\_4(M\_FLASH\_CR\_SER | M\_FLASH\_SECTOR\_4)

- set Flash CR register with M\_FLASH\_CR\_STRT.

##### 10.33.6.7.3 daugwyrterr-EraseErrSection-LLR-003

Requirement ID: H398-LLD-GWY-FNC-4229

The function shall wait till loop counter is less than M\_PROGRAM\_WORD\_MAX\_COUNT AND ongoing programs in FLASH status register is not zero when loop counter is less than 10000 (M\_PROGRAM\_WORD\_MAX\_COUNT)

##### 10.33.6.7.4 daugwyrterr-EraseErrSection-LLR-004

Requirement ID: H398-LLD-GWY-FNC-4230

The function shall disable Sector Erase(SER Bit) and reset Flash CR register sector number bits to erase sector0(M\_ERASE\_OPERATION\_MASK) when loop counter is less than M\_PROGRAM\_WORD\_MAX\_COUNT.

##### 10.33.6.7.5 daugwyrterr-EraseErrSection-LLR-005

Requirement ID: H398-LLD-GWY-FNC-4231

The function shall return TRUE when loop counter is less than M\_PROGRAM\_WORD\_MAX\_COUNT.

### 10.33.7 ProgramWord

Low Level Design Details about CSU ProgramWord will follow in the sub sections.

#### 10.33.7.1 Brief Description

ProgramWord function program writes new program data to Flash.

#### 10.33.7.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD(H398-003-012-GWY).

#### 10.33.7.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.33.7.4 Parameter list (Input/Output)

Inputs : T\_UINT32 data - Program word data

Outputs : T\_UINT32 address - Program word address

#### 10.33.7.5 Return Value

T\_BOOL - return TRUE when program operation is completed

- return FALSE when program operation is not completed

#### 10.33.7.6 Other CSUs called by this CSU

None

#### 10.33.7.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to ProgramWord.

##### 10.33.7.7.1 daugwyrterr-ProgramWord-LLR-001

Requirement ID: H398-LLD-GWY-FNC-4240

The function shall wait till loop counter is less than M\_PROGRAM\_WORD\_MAX\_COUNT and ongoing programs

in FLASH status register is not equal to zero.

##### 10.33.7.7.2 daugwyrterr-ProgramWord-LLR-002

Requirement ID: H398-LLD-GWY-FNC-4241

The function shall configure CR register to program the new data, when loop counter is less than M\_PROGRAM\_WORD\_MAX\_COUNT

-reset Flash CR register by M\_CR\_PSIZE\_MASK, to select program size x8

-set Flash CR register with M\_FLASH\_PSIZE\_WORD, to select program size x32

-set Flash CR register with M\_FLASH\_CR\_PG, to activate flash programming, store the data in passed address

##### 10.33.7.7.3 daugwyrterr-ProgramWord-LLR-003

Requirement ID: H398-LLD-GWY-FNC-4242

The function shall wait till loop counter is less than M\_PROGRAM\_WORD\_MAX\_COUNT and ongoing programs in

FLASH status register is not equal to zero when loop counter is less than 10000 (M\_PROGRAM\_WORD\_MAX\_COUNT).

##### 10.33.7.7.4 daugwyrterr-ProgramWord-LLR-004

Requirement ID: H398-LLD-GWY-FNC-4243

The function shall reset Programming (M\_FLASH\_CR\_PG) bit of FLASH control register when loop counter is less than M\_PROGRAM\_WORD\_MAX\_COUNT.

##### 10.33.7.7.5 daugwyrterr-ProgramWord-LLR-005

Requirement ID: H398-LLD-GWY-FNC-4244

The function shall return TRUE when program operation is completed i.e. loop counter is less than M\_PROGRAM\_WORD\_MAX\_COUNT.

## 10.34 daugwysound

The daugwysound CSC implements sound related routines.

### 10.34.1 SoundSynthesize

Low Level Design Details about CSU SoundSynthesize will follow in the sub sections.

#### 10.34.1.1 Brief Description

The SoundSynthesize synthesizes the audio.

#### 10.34.1.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.34.1.3 List of global variables accessed and modified

Accessed : Tx\_a429\_bnr

Modified : Ssound

#### 10.34.1.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.34.1.5 Return Value

T\_UINT16 - Return synthesized sound

#### 10.34.1.6 Other CSUs called by this CSU

None

#### 10.34.1.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to SoundSynthesize

##### 10.34.1.7.1 daugwysound-SoundSynthesize-LLR-001

Requirement ID: H398-LLD-GWY-FNC-4275

The function shall set output variable with M\_TWO\_ZERO\_FOUR\_EIGHT

##### 10.34.1.7.2 daugwysound-SoundSynthesize-LLR-002

Requirement ID: H398-LLD-GWY-FNC-4276

The function shall loop from M\_ZERO until MAX\_AUDIO and set fGain of synthesized sound for all the loop indices with volume gain of index (data of a429 transmission binary (Tx\_a429\_bnr) of index LABEL\_0100)

##### 10.34.1.7.3 daugwysound-SoundSynthesize-LLR-003

Requirement ID: H398-LLD-GWY-FNC-4277

The function shall loop from M\_ZERO until MAX\_AUDIO and add output variable with (product of (T\_FLOAT32) and (pi16Rec of psRec of synthesized sound with index (loop index) is added to u16Step of synthesized sound with index (loop index))) multiplied to product of M\_MAX\_VOLUME and fGain of synthesized sound with index (loop index)) when,

1. u8Active of synthesized sound with index (loop index) is not M\_ZERO
2. eWave of synthesized sound with index (loop index) is SOUND\_RECORDING

##### 10.34.1.7.4 daugwysound-SoundSynthesize-LLR-004

Requirement ID: H398-LLD-GWY-FNC-4278

The function shall loop from M\_ZERO until MAX\_AUDIO, set u16Step of synthesized sound for index (loop index) with M\_ZERO and set play variable of index (loop index) with FALSE when,

1. u8Active of synthesized sound with index (loop index) is not M\_ZERO
2. eWave of synthesized sound with index (loop index) is SOUND\_RECORDING
3. pre-increment of u16Step of synthesized sound with index (loop index) is greater than or equal to u16Length of psRec of synthesized sound with index (loop index)

##### 10.34.1.7.5 daugwysound-SoundSynthesize-LLR-005

Requirement ID: H398-LLD-GWY-FNC-4279

The function shall loop from M\_ZERO until MAX\_AUDIO and set u8Active of synthesized sound for index (loop index) with FALSE when,

1. u8Active of synthesized sound with index (loop index) is not M\_ZERO
2. eWave of synthesized sound with index (loop index) is SOUND\_RECORDING
3. pre-increment of u16Step of synthesized sound with index (loop index) is greater than or equal to u16Length of psRec of synthesized sound with index (loop index)
4. u8Single of synthesized sound with index (loop index) is not M\_ZERO

##### 10.34.1.7.6 daugwysound-SoundSynthesize-LLR-006

Requirement ID: H398-LLD-GWY-FNC-4280

The function shall loop from M\_ZERO until MAX\_AUDIO and do nothing when,

1. u8Active of synthesized sound with index (loop index) is not M\_ZERO
2. eWave of synthesized sound with index (loop index) is other than SOUND\_RECORDING

##### 10.34.1.7.7 daugwysound-SoundSynthesize-LLR-007

Requirement ID: H398-LLD-GWY-FNC-4281

The function shall loop from M\_ZERO until MAX\_AUDIO and update fTime of synthesized sound for index (loop index) by adding the previous value of fTime and (product of (division of M\_FP\_ONE by M\_TMR\_7\_INTR\_FREQ) and M\_FP\_TWO\_FIFTY\_FIVE) multiplied to (u16Freq of synthesized sound for index (loop index)) when,u8Active of synthesized sound with index (loop index) is not M\_ZERO

##### 10.34.1.7.8 daugwysound-SoundSynthesize-LLR-008

Requirement ID: H398-LLD-GWY-FNC-4282

The function shall loop from M\_ZERO until MAX\_AUDIO and decrement fTime of synthesized sound for index (loop index) by M\_FP\_TWO\_FIFTY\_FIVE when,

1. u8Active of synthesized sound with index (loop index) is not M\_ZERO
2. fTime of synthesized sound for index (loop index) is greater than M\_FP\_TWO\_FIFTY\_FIVE

##### 10.34.1.7.9 daugwysound-SoundSynthesize-LLR-009

Requirement ID: H398-LLD-GWY-FNC-4283

The function shall set output variable with M\_FOUR\_ZERO\_NINE\_FIVE when output variable is greater than M\_FOUR\_ZERO\_NINE\_FIVE

##### 10.34.1.7.10 daugwysound-SoundSynthesize-LLR-010

Requirement ID: H398-LLD-GWY-FNC-4284

The function shall set output variable with M\_ZERO when output variable is less than M\_ZERO

##### 10.34.1.7.11 daugwysound-SoundSynthesize-LLR-011

Requirement ID: H398-LLD-GWY-FNC-4285

The function shall return output variable.

### 10.34.2 AudioTest

Low Level Design Details about CSU AudioTest will follow in the sub sections.

#### 10.34.2.1 Brief Description

The AudioTest performs the Audio sound test

#### 10.34.2.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.34.2.3 List of global variables accessed and modified

Accessed : Tx\_a429\_bnr, Op\_mode

Modified : Sound\_test\_mode,Ssound

#### 10.34.2.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.34.2.5 Return Value

None

#### 10.34.2.6 Other CSUs called by this CSU

DUDiscreteInputsStatus

#### 10.34.2.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to AudioTest

##### 10.34.2.7.1 daugwysound-AudioTest-LLR-001

Requirement ID: H398-LLD-GWY-FNC-4294

The function shall do the following when init value is equal to FALSE and

return value of the function call DUDiscreteInputsStatus with parameters (IP\_LABEL\_0270,M\_TEN, ALL\_DU) is not FALSE:

1. pre-increment the counter variable by one

##### 10.34.2.7.2 daugwysound-AudioTest-LLR-002

Requirement ID: H398-LLD-GWY-FNC-4295

The function shall do the following when init value is equal to FALSE and

return value of the function call DUDiscreteInputsStatus with parameters (IP\_LABEL\_0270,M\_TEN, ALL\_DU) is FALSE:

a) Set the counter variable to M\_ZERO

##### 10.34.2.7.3 daugwysound-AudioTest-LLR-003

Requirement ID: H398-LLD-GWY-FNC-4296

The function shall do the following when init value is equal to FALSE and

(status of a429 transmission binary with index (LABEL\_040) is equal to OK AND data of a429 transmission binary with index (LABEL\_040) is less than M\_TWENTY AND left\_engine\_out of operation mode AND right\_engine\_out of operation mode AND counter variable is greater than M\_SOUND\_RATE) returns TRUE:

1. set the init value with TRUE
2. set Sound\_test\_mode with TRUE
3. reset the counter variable
4. set flag variable with GREETING\_412
5. set play variable of index (GREETING\_412) with TRUE

##### 10.34.2.7.4 daugwysound-AudioTest-LLR-004

Requirement ID: H398-LLD-GWY-FNC-4297

The function shall do the following when init value is equal to FALSE and

(status of a429 transmission binary with index (LABEL\_040) is equal to OK AND data of a429 transmission binary with index (LABEL\_040) is less than twenty AND left\_engine\_out of operation mode AND right\_engine\_out of operation mode AND counter variable is greater than M\_SOUND\_RATE) returns FALSE:

1. set the init value with FALSE
2. set Sound\_test\_mode with FALSE
3. set flag variable with GREETING\_412
4. set play variable of index (GREETING\_412) with FALSE

##### 10.34.2.7.5 daugwysound-AudioTest-LLR-005

Requirement ID: H398-LLD-GWY-FNC-4298

The function shall do the following when counter disable is greater than (product of M\_SOUND\_RATE\_FACTOR\_P\_FIVE and M\_SOUND\_RATE) and (DUDiscreteInputsStatus with parameters IP\_LABEL\_0270,M\_TEN,ALL\_DU) returns TRUE.

1. set counter disable with M\_ZERO
2. set init value with FALSE
3. set Sound\_test\_mode with FALSE

##### 10.34.2.7.6 daugwysound-AudioTest-LLR-006

Requirement ID: H398-LLD-GWY-FNC-4299

The function shall do the following from M\_ZERO until MAX\_AUDIO minus 1 when counter disable is greater than (product of M\_SOUND\_RATE\_FACTOR\_P\_FIVE and M\_SOUND\_RATE) and (DUDiscreteInputsStatus with parameters IP\_LABEL\_0270,M\_TEN,ALL\_DU) returns TRUE

1. set play variable of index (index variable) with FALSE
2. set u8Active of synthesized sound of index (index variable) with FALSE

##### 10.34.2.7.7 daugwysound-AudioTest-LLR-007

Requirement ID: H398-LLD-GWY-FNC-4300

The function shall pre-increment the counter disable by one when ((DUDiscreteInputsStatus with parameters IP\_LABEL\_0270,M\_TEN,ALL\_DU returns FALSE) AND (init value is equal to TRUE)) returns TRUE

##### 10.34.2.7.8 daugwysound-AudioTest-LLR-008

Requirement ID: H398-LLD-GWY-FNC-4301

The function shall reset the counter disable when (NOT of(((DUDiscreteInputsStatus with parameters IP\_LABEL\_0270,M\_TEN,ALL\_DU) AND (init value is equal to TRUE)) returns FALSE

##### 10.34.2.7.9 daugwysound-AudioTest-LLR-009

Requirement ID: H398-LLD-GWY-FNC-4302

The function shall set u8Active of synthesized sound of index GREETING\_412 with TRUE when,

1. init value is equal to TRUE
2. flag variable is GREETING\_412
3. play variable of index GREETING\_412 is TRUE

##### 10.34.2.7.10 daugwysound-AudioTest-LLR-010

Requirement ID: H398-LLD-GWY-FNC-4303

The function shall do the following

1. set u8Active of synthesized sound of index GREETING\_412 with FALSE
2. set play variable of index LOW\_ROTOR\_RPM\_ALERT with TRUE
3. set counter sequence with M\_ZERO
4. set flag variable with LOW\_ROTOR\_RPM\_ALERT

when,

1. init value is equal to TRUE
2. flag variable is GREETING\_412
3. play variable of index GREETING\_412 is FALSE

##### 10.34.2.7.11 daugwysound-AudioTest-LLR-011

Requirement ID: H398-LLD-GWY-FNC-4304

The function shall pre-increment the counter sequence when,

1. init value is equal to TRUE
2. flag variable is LOW\_ROTOR\_RPM\_ALERT

##### 10.34.2.7.12 daugwysound-AudioTest-LLR-012

Requirement ID: H398-LLD-GWY-FNC-4305

The function shall set u8Active of synthesized sound of index LOW\_ROTOR\_RPM\_ALERT with TRUE when,

1. init value is equal to TRUE
2. flag variable is LOW\_ROTOR\_RPM\_ALERT
3. counter sequence is greater than or equal to (product of M\_AUDIO\_ALERTS\_INTERSPACED\_VAL and M\_SOUND\_RATE)
4. play variable with index LOW\_ROTOR\_RPM\_ALERT is TRUE

##### 10.34.2.7.13 daugwysound-AudioTest-LLR-013

Requirement ID: H398-LLD-GWY-FNC-4306

The function shall do the following

1. set u8Active of synthesized sound of index LOW\_ROTOR\_RPM\_ALERT with FALSE
2. set play variable of index VNE\_WARNING with TRUE
3. reset counter sequence
4. set flag variable with VNE\_WARNING

when,

1. init value is equal to TRUE
2. flag variable is LOW\_ROTOR\_RPM\_ALERT
3. counter sequence is greater than or equal to (product of M\_AUDIO\_ALERTS\_INTERSPACED\_VAL and M\_SOUND\_RATE)
4. play variable with index LOW\_ROTOR\_RPM\_ALERT is FALSE

##### 10.34.2.7.14 daugwysound-AudioTest-LLR-014

Requirement ID: H398-LLD-GWY-FNC-4307

The function shall reset sweep count and pre-increment counter sequence when,

1. init value is equal to TRUE
2. flag variable is VNE\_WARNING

##### 10.34.2.7.15 daugwysound-AudioTest-LLR-015

Requirement ID: H398-LLD-GWY-FNC-4308

The function shall set u8Active of synthesized sound of index VNE\_WARNING with TRUE when,

1. init value is equal to TRUE
2. flag variable is VNE\_WARNING
3. counter sequence is greater than or equal to (product of M\_AUDIO\_ALERTS\_INTERSPACED\_VAL and M\_SOUND\_RATE)
4. play variable with index (VNE\_WARNING) is equal to TRUE

##### 10.34.2.7.16 daugwysound-AudioTest-LLR-016

Requirement ID: H398-LLD-GWY-FNC-4309

The function shall do the following

1. set u8Active of synthesized sound of index VNE\_WARNING with FALSE
2. set play variable of index MASTER\_CAUTION\_WARNING\_TONE with TRUE
3. reset counter sequence
4. set flag variable with MASTER\_CAUTION\_WARNING\_TONE

when,

1. init value is equal to TRUE
2. flag variable is VNE\_WARNING
3. counter sequence is greater than or equal to (product of M\_AUDIO\_ALERTS\_INTERSPACED\_VAL and M\_SOUND\_RATE)
4. play variable with index (VNE\_WARNING) is equal to FALSE
5. counter sequence is greater than or equal to (product of M\_SOUND\_RATE\_TWO\_P\_TWO\_FIVE and M\_SOUND\_RATE)

##### 10.34.2.7.17 daugwysound-AudioTest-LLR-017

Requirement ID: H398-LLD-GWY-FNC-4310

The function shall pre-increment the counter sequence

when,

1. init value is equal to TRUE
2. flag variable is MASTER\_CAUTION\_WARNING\_TONE

##### 10.34.2.7.18 daugwysound-AudioTest-LLR-018

Requirement ID: H398-LLD-GWY-FNC-4311

The function shall set u8Active of synthesized sound of index MASTER\_CAUTION\_WARNING\_TONE with TRUE when,

1. init value is equal to TRUE
2. flag variable is MASTER\_CAUTION\_WARNING\_TONE
3. counter sequence is greater than or equal to (product of M\_AUDIO\_ALERTS\_INTERSPACED\_VAL and M\_SOUND\_RATE)
4. play variable with index MASTER\_CAUTION\_WARNING\_TONE is TRUE

##### 10.34.2.7.19 daugwysound-AudioTest-LLR-019

Requirement ID: H398-LLD-GWY-FNC-4312

The function shall set the following

1. set u8Active of synthesized sound of index MASTER\_CAUTION\_WARNING\_TONE with FALSE
2. set play variable of index AUTOPILOT\_DECOUPLE with TRUE
3. reset counter sequence
4. set flag variable with AUTOPILOT\_DECOUPLE

when,

1. init value is equal to TRUE
2. flag variable is MASTER\_CAUTION\_WARNING\_TONE
3. counter sequence is greater than or equal to (product of M\_AUDIO\_ALERTS\_INTERSPACED\_VAL and M\_SOUND\_RATE)
4. play variable with index MASTER\_CAUTION\_WARNING\_TONE is FALSE

##### 10.34.2.7.20 daugwysound-AudioTest-LLR-020

Requirement ID: H398-LLD-GWY-FNC-4313

The function shall pre-increment the counter sequence

when,

1. init value is equal to TRUE
2. flag variable is AUTOPILOT\_DECOUPLE

##### 10.34.2.7.21 daugwysound-AudioTest-LLR-021

Requirement ID: H398-LLD-GWY-FNC-4314

The function shall set u8Active of synthesized sound of index AUTOPILOT\_DECOUPLE with TRUE when,

1. init value is equal to TRUE
2. flag variable is AUTOPILOT\_DECOUPLE
3. counter sequence is greater than or equal to (product of M\_AUDIO\_ALERTS\_INTERSPACED\_VAL and M\_SOUND\_RATE)
4. play variable with index AUTOPILOT\_DECOUPLE is TRUE

##### 10.34.2.7.22 daugwysound-AudioTest-LLR-022

Requirement ID: H398-LLD-GWY-FNC-4315

The function shall set the following

1. set u8Active of synthesized sound of index AUTOPILOT\_DECOUPLE with FALSE
2. set play variable of index CHECK\_ALTITUDE\_ALERT with TRUE
3. reset counter sequence
4. set flag variable with CHECK\_ALTITUDE\_ALERT

when,

1. init value is equal to TRUE
2. flag variable is AUTOPILOT\_DECOUPLE
3. counter sequence is greater than or equal to (product of M\_AUDIO\_ALERTS\_INTERSPACED\_VAL and M\_SOUND\_RATE)
4. play variable with index AUTOPILOT\_DECOUPLE is FALSE

##### 10.34.2.7.23 daugwysound-AudioTest-LLR-023

Requirement ID: H398-LLD-GWY-FNC-4316

The function shall pre-increment the counter sequence

when,

1. init value is equal to TRUE
2. flag variable is CHECK\_ALTITUDE\_ALERT

##### 10.34.2.7.24 daugwysound-AudioTest-LLR-024

Requirement ID: H398-LLD-GWY-FNC-4317

The function shall set u8Active of synthesized sound of index CHECK\_ALTITUDE\_ALERT with TRUE when,

1. init value is equal to TRUE
2. flag variable is CHECK\_ALTITUDE\_ALERT
3. counter sequence is greater than or equal to (product of M\_AUDIO\_ALERTS\_INTERSPACED\_VAL and M\_SOUND\_RATE)
4. play variable with index CHECK\_ALTITUDE\_ALERT is TRUE

##### 10.34.2.7.25 daugwysound-AudioTest-LLR-025

Requirement ID: H398-LLD-GWY-FNC-4318

The function shall increment sweep count by one when,

1. init value is equal to TRUE
2. flag variable is CHECK\_ALTITUDE\_ALERT
3. counter sequence is greater than or equal to (product of M\_AUDIO\_ALERTS\_INTERSPACED\_VAL and M\_SOUND\_RATE)
4. play variable with index CHECK\_ALTITUDE\_ALERT is FALSE

##### 10.34.2.7.26 daugwysound-AudioTest-LLR-026

Requirement ID: H398-LLD-GWY-FNC-4319

The function shall set the following

1. set play variable of index CHECK\_ALTITUDE\_ALERT with TRUE
2. set u8Active of synthesized sound of index CHECK\_ALTITUDE\_ALERT with TRUE

when,

1. init value is equal to TRUE
2. flag variable is CHECK\_ALTITUDE\_ALERT
3. counter sequence is greater than or equal to (product of M\_AUDIO\_ALERTS\_INTERSPACED\_VAL and M\_SOUND\_RATE)
4. play variable with index CHECK\_ALTITUDE\_ALERT is FALSE
5. sweep count is one

##### 10.34.2.7.27 daugwysound-AudioTest-LLR-027

Requirement ID: H398-LLD-GWY-FNC-4320

The function shall set the following

1. set u8Active of synthesized sound of index CHECK\_ALTITUDE\_ALERT with FALSE
2. set play variable of index ALTS\_ALERT with TRUE
3. reset counter sequence
4. set flag variable with ALTS\_ALERT

when,

1. init value is equal to TRUE
2. flag variable is CHECK\_ALTITUDE\_ALERT
3. counter sequence is greater than or equal to (product of M\_AUDIO\_ALERTS\_INTERSPACED\_VAL and M\_SOUND\_RATE)
4. play variable with index CHECK\_ALTITUDE\_ALERT is FALSE
5. sweep count is not one

##### 10.34.2.7.28 daugwysound-AudioTest-LLR-028

Requirement ID: H398-LLD-GWY-FNC-4321

The function shall reset sweep count and pre-increment counter sequence

when,

1. init value is equal to TRUE
2. flag variable is ALTS\_ALERT

##### 10.34.2.7.29 daugwysound-AudioTest-LLR-029

Requirement ID: H398-LLD-GWY-FNC-4322

The function shall set u8Active of synthesized sound of index ALTS\_ALERT with TRUE when,

1. init value is equal to TRUE
2. flag variable is ALTS\_ALERT
3. counter sequence is greater than or equal to (product of M\_AUDIO\_ALERTS\_INTERSPACED\_VAL and M\_SOUND\_RATE)
4. play variable with index ALTS\_ALERT is TRUE

##### 10.34.2.7.30 daugwysound-AudioTest-LLR-030

Requirement ID: H398-LLD-GWY-FNC-4323

The function shall set the following

1. set u8Active of synthesized sound of index ALTS\_ALERT with FALSE
2. set play variable of index ONE\_HUNDRED\_FEET with TRUE
3. reset counter sequence
4. set flag variable with ONE\_HUNDRED\_FEET

when,

1. init value is equal to TRUE
2. flag variable is ALTS\_ALERT
3. counter sequence is greater than or equal to (product of M\_AUDIO\_ALERTS\_INTERSPACED\_VAL and M\_SOUND\_RATE)
4. play variable with index ALTS\_ALERT is FALSE

##### 10.34.2.7.31 daugwysound-AudioTest-LLR-031

Requirement ID: H398-LLD-GWY-FNC-4324

The function shall pre-increment counter sequence

when,

1. init value is equal to TRUE
2. flag variable is ONE\_HUNDRED\_FEET

##### 10.34.2.7.32 daugwysound-AudioTest-LLR-032

Requirement ID: H398-LLD-GWY-FNC-4325

The function shall set u8Active of synthesized sound of index ONE\_HUNDRED\_FEET with TRUE when,

1. init value is equal to TRUE
2. flag variable is ONE\_HUNDRED\_FEET
3. counter sequence is greater than or equal to (product of M\_AUDIO\_ALERTS\_INTERSPACED\_VAL and M\_SOUND\_RATE)
4. play variable with index ONE\_HUNDRED\_FEET is TRUE

##### 10.34.2.7.33 daugwysound-AudioTest-LLR-033

Requirement ID: H398-LLD-GWY-FNC-4326

The function shall set the following

1. set u8Active of synthesized sound of index ONE\_HUNDRED\_FEET with FALSE
2. set play variable of index ONE\_HUNDERED\_FIFTY\_FEET with TRUE
3. reset counter sequence
4. set flag variable with ONE\_HUNDERED\_FIFTY\_FEET

when,

1. init value is equal to TRUE
2. flag variable is ONE\_HUNDRED\_FEET
3. counter sequence is greater than or equal to (product of M\_AUDIO\_ALERTS\_INTERSPACED\_VAL and M\_SOUND\_RATE)
4. play variable with index ONE\_HUNDRED\_FEET is FALSE

##### 10.34.2.7.34 daugwysound-AudioTest-LLR-034

Requirement ID: H398-LLD-GWY-FNC-4327

The function shall pre-increment counter sequence

when,

1. init value is equal to TRUE
2. flag variable is ONE\_HUNDERED\_FIFTY\_FEET

##### 10.34.2.7.35 daugwysound-AudioTest-LLR-035

Requirement ID: H398-LLD-GWY-FNC-4328

The function shall set u8Active of synthesized sound of index ONE\_HUNDERED\_FIFTY\_FEET with TRUE when,

1. init value is equal to TRUE
2. flag variable is ONE\_HUNDERED\_FIFTY\_FEET
3. counter sequence is greater than or equal to (product of M\_AUDIO\_ALERTS\_INTERSPACED\_VAL and M\_SOUND\_RATE)
4. play variable with index ONE\_HUNDERED\_FIFTY\_FEET is TRUE

##### 10.34.2.7.36 daugwysound-AudioTest-LLR-036

Requirement ID: H398-LLD-GWY-FNC-4329

The function shall set the following

1. set u8Active of synthesized sound of index ONE\_HUNDERED\_FIFTY\_FEET with FALSE
2. reset counter sequence
3. set flag variable with GREETING\_412
4. set init variable with FALSE

when,

1. init value is equal to TRUE
2. flag variable is ONE\_HUNDERED\_FIFTY\_FEET
3. counter sequence is greater than or equal to (product of M\_AUDIO\_ALERTS\_INTERSPACED\_VAL and M\_SOUND\_RATE)
4. play variable with index ONE\_HUNDERED\_FIFTY\_FEET is FALSE

##### 10.34.2.7.37 daugwysound-AudioTest-LLR-037

Requirement ID: H398-LLD-GWY-FNC-4330

The function shall do nothing

when,

1. init value is equal to TRUE
2. flag variable is other than GREETING\_412, LOW\_ROTOR\_RPM\_ALERT, VNE\_WARNING, MASTER\_CAUTION\_WARNING\_TONE, AUTOPILOT\_DECOUPLE, CHECK\_ALTITUDE\_ALERT, ALTS\_ALERT, ONE\_HUNDRED\_FEET and ONE\_HUNDERED\_FIFTY\_FEET

### 10.34.3 SoundTask

Low Level Design Details about CSU SoundTask will follow in the sub sections.

#### 10.34.3.1 Brief Description

The SoundTask performs the sound task

#### 10.34.3.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.34.3.3 List of global variables accessed and modified

Accessed : Appdata\_appdata, Sound\_status, Sound\_test\_mode, Cmdownload\_lp\_cmd\_data, App\_emode,App\_strap, Timesplay

Modified : Sound\_status, Ssound , Timesplay

#### 10.34.3.4 Parameter list (Input/Output)

Inputs : Void \* pData – data pointer

Outputs : None

#### 10.34.3.5 Return Value

None

#### 10.34.3.6 Other CSUs called by this CSU

OsSemPend

SetAudioInhibit

AudioTest

#### 10.34.3.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to SoundTask

##### 10.34.3.7.1 daugwysound-SoundTask-LLR-001

Requirement ID: H398-LLD-GWY-FNC-4339

The function shall loop from audio Index equal to M\_ZERO until MAX\_AUDIO minus 1 and set the following:

1. Set u16Length of synthesized audio for index (audio Index) with size of audio for index (audio Index) of Appdata\_appdata
2. Set pi16Rec of synthesized audio for index (audio Index) with data of audio for index (audio Index) of Appdata\_appdata

##### 10.34.3.7.2 daugwysound-SoundTask-LLR-002

Requirement ID: H398-LLD-GWY-FNC-4340

The function shall loop from audio Index equal to M\_ZERO until MAX\_AUDIO minus 1 and set the following:

1. Set eWave of synthesized sound for index (audio Index) with SOUND\_RECORDING
2. Set fGain of synthesized sound for index (audio Index) with M\_GAIN\_P\_ONE\_TWO\_FIVE\_F
3. Set u8Active of synthesized sound for index (audio Index) with FALSE

##### 10.34.3.7.3 daugwysound-SoundTask-LLR-003

Requirement ID: H398-LLD-GWY-FNC-4341

The function shall loop from audio Index equal to M\_ZERO until MAX\_AUDIO minus 1 and set the following when audio Index is either LOW\_ROTOR\_RPM\_ALERT or VNE\_WARNING:

1. Set u8Single of synthesized sound for index (audio Index) with FALSE

##### 10.34.3.7.4 daugwysound-SoundTask-LLR-004

Requirement ID: H398-LLD-GWY-FNC-4342

The function shall loop from audio Index equal to M\_ZERO until MAX\_AUDIO minus 1 and set the following when audio Index is neither LOW\_ROTOR\_RPM\_ALERT nor VNE\_WARNING:

1. Set u8Single of synthesized sound for index (audio Index) with TRUE

##### 10.34.3.7.5 daugwysound-SoundTask-LLR-005

Requirement ID: H398-LLD-GWY-FNC-4343

The function shall loop from audio Index equal to M\_ZERO until MAX\_AUDIO minus 1 and set the following:

1. Set u16Step of synthesized sound for index (audio Index) with M\_ZERO
2. Set psRec of synthesized sound for index (audio Index) with reference of audio of index (audio Index)

##### 10.34.3.7.6 daugwysound-SoundTask-LLR-006

Requirement ID: H398-LLD-GWY-FNC-4344

The function shall loop forever and do the following:

1. Call OsSemPend with parameters (Semaphore Sound Task, M\_ZERO, reference of error status)
2. Do nothing when ((App\_emode is not equal to APP\_NORMAL) OR NOT of App\_strap ) returns TRUE.

##### 10.34.3.7.7 daugwysound-SoundTask-LLR-007

Requirement ID: H398-LLD-GWY-FNC-4345

The function shall loop forever and do the following when App\_emode is equal to APP\_NORMAL OR NOT of App\_strap and the function is entered:

1. Call AudioTest

##### 10.34.3.7.8 daugwysound-SoundTask-LLR-008

Requirement ID: H398-LLD-GWY-FNC-4346

The function shall loop forever and do the following when App\_emode is equal to APP\_NORMAL OR NOT of App\_strap and the function is entered:

1. Loop from index value M\_ZERO until MAX\_AUDIO minus 1 when Sound\_test\_mode is M\_ZERO
2. Set play variable with Sound\_status for every index of loop and set Sound\_status for every index value with FALSE when Sound\_test\_mode is M\_ZERO, Mute of audio\_cfg of Cmdownload\_lp\_cmd\_data for every index is M\_ZERO and Sound\_status for every index is not M\_ZERO

##### 10.34.3.7.9 daugwysound-SoundTask-LLR-009

Requirement ID: H398-LLD-GWY-FNC-4347

The function shall loop forever and set the following for every index until index value is MAX\_AUDIO minus 1 when App\_emode is equal to APP\_NORMAL OR NOT of App\_strap and the function is entered:

1. Decrement the Times play by one
2. Set play variable with TRUE
3. Set Sound\_status with FALSE

When,

1. Sound\_test\_mode is M\_ZERO
2. Mute of audio\_cfg of Cmdownload\_lp\_cmd\_data for every index is M\_ZERO
3. Sound\_status for every index is M\_ZERO
4. Times play for every index is greater than zero AND NOT of play variable with every index is not M\_ZERO

##### 10.34.3.7.10 daugwysound-SoundTask-LLR-010

Requirement ID: H398-LLD-GWY-FNC-4348

The function shall loop forever and set the following for every index until index value is MAX\_AUDIO minus 1 when App\_emode is equal to APP\_NORMAL OR NOT of App\_strap and the function is entered:

1. Set Sound\_status with FALSE
2. Set Times play with M\_ZERO
3. Set play variable with FALSE

When,

1. Sound\_test\_mode is M\_ZERO
2. Mute of audio\_cfg of Cmdownload\_lp\_cmd\_data for every index is not M\_ZERO

##### 10.34.3.7.11 daugwysound-SoundTask-LLR-011

Requirement ID: H398-LLD-GWY-FNC-4349

The function shall loop forever and set the following for every index from M\_ONE until index value is MAX\_AUDIO minus 1 when App\_emode OR NOT of App\_strap is equal to APP\_NORMAL and the function is entered:

1. Set u8Active of sound with TRUE to activate the audio

When,

1. Sound\_test\_mode is M\_ZERO
2. play variable for every index is TRUE
3. play variable with index (subtraction of index by M\_ONE) is FALSE

##### 10.34.3.7.12 daugwysound-SoundTask-LLR-012

Requirement ID: H398-LLD-GWY-FNC-4350

The function shall loop forever and set the following for every index from M\_ONE until index value is MAX\_AUDIO minus 1 when App\_emode is equal to APP\_NORMAL OR NOT of App\_strap and the function is entered:

1. Set u8Active of sound with FALSE for every next index (i.e addition of index with one until MAX\_AUDIO minus 1) to turn off higher precedence Audio

When,

1. Sound\_test\_mode is M\_ZERO
2. play variable for every index is TRUE
3. play variable with index (subtraction of index by M\_ONE) is FALSE

##### 10.34.3.7.13 daugwysound-SoundTask-LLR-013

Requirement ID: H398-LLD-GWY-FNC-4351

The function shall loop forever and stop the execution from the current condition for every index from M\_ONE until index value is MAX\_AUDIO minus 1

When,

1. App\_emode is equal to APP\_NORMAL OR NOT of App\_strap
2. the function is entered
3. Sound\_test\_mode is M\_ZERO
4. play variable for every index is TRUE

##### 10.34.3.7.14 daugwysound-SoundTask-LLR-014

Requirement ID: H398-LLD-GWY-FNC-4352

The function shall loop forever and do the following for every index from M\_ONE until index value is MAX\_AUDIO minus 1 when App\_emode is equal to APP\_NORMAL OR NOT of App\_strap and the function is entered:

1. Set u8Active of sound with FALSE

When,

1. Sound\_test\_mode is M\_ZERO
2. play variable for every index is not TRUE

### 10.34.4 SoundInit

Low Level Design Details about CSU SOUND\_Init will follow in the sub sections.

#### 10.34.4.1 Brief Description

The SoundInit initilaizes the task signaling paramters .

#### 10.34.4.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.34.4.3 List of global variables accessed and modified

Accessed : SoundTaskStk

Modified : SoundTaskStk

#### 10.34.4.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.34.4.5 Return Value

None

#### 10.34.4.6 Other CSUs called by this CSU

TbaseTaskSignaling

OsSemCreate

OsTaskCreate

SoundTask

RterrError

RterrForever

#### 10.34.4.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to SoundInit

##### 10.34.4.7.1 daugwysound-SoundInit-LLR-001

Requirement ID: H398-LLD-GWY-FNC-4364

The function shall do the following to install task signaling parameters

1. Call OsSemCreate with parameter M\_ZERO and set the return value to Semaphore Sound Task
2. Call TbaseTaskSignaling with parameters M\_SOUND\_TASK\_TICKS and Semaphore Sound Task

##### 10.34.4.7.2 daugwysound-SoundInit-LLR-002

Requirement ID: H398-LLD-GWY-FNC-4365

The function shall loop do the following from stack counter is equal to M\_ZERO until M\_SOUND\_TASK\_STK\_SIZE minus 1

1. Set Soundtaskstk of index (stack counter) with M\_CBIT\_TASK\_STK\_VAL

##### 10.34.4.7.3 daugwysound-SoundInit-LLR-003

Requirement ID: H398-LLD-GWY-FNC-4366

The function shall do the following :

1. Call OsTaskCreate with parameters (function pointer SoundTask, M\_HW\_NULL,reference of Soundtaskstk with index M\_SOUND\_TASK\_STK\_SIZE, M\_SOUND\_TASK\_PRIO)
2. Call RterrError with parameters (BIT\_ERR,M\_THREE, function pointer RterrForever) when function call OsTaskCreate returns other than M\_OS\_NO\_ERR
3. Do nothing when function call OsTaskCreate returns M\_OS\_NO\_ERR

## 10.35 daugwytach

This daugwytach CSC contains implementation of routines to read tachometer.

### 10.35.1 TachRead

Low Level Design Details about CSU TachRead will follow in the sub sections.

#### 10.35.1.1 Brief Description

The function TachRead shall Calculate the sampled tach reading for the selected channel.

#### 10.35.1.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD(H398-003-012-GWY).

#### 10.35.1.3 List of global variables accessed and modified

Accessed : U32\_critical\_sr

Modified : U32\_critical\_sr

#### 10.35.1.4 Parameter list (Input/Output)

Inputs : T\_UINT8 u8\_chan - Tach channel no (TACH\_CHAN\_1 to TACH\_CHAN\_8)

Outputs : None

#### 10.35.1.5 Return Value

T\_FLOAT32 - Return the tach channel sampling reading value

#### 10.35.1.6 Other CSUs called by this CSU

SaveStatusReg,

RestoreStatusReg

#### 10.35.1.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TachRead.

##### 10.35.1.7.1 daugwytach-TachRead-LLR-001

Requirement ID: H398-LLD-GWY-FNC-4376

The function shall Send a request to uCOS to disable all the interrupt by calling the function SaveStatusReg () and the obtained value of function is updated in U32\_critical\_sr.

##### 10.35.1.7.2 daugwytach-TachRead-LLR-002

Requirement ID: H398-LLD-GWY-FNC-4377

The function shall set big buffer to psBigBuffer of As\_tach\_chan with index u8Chan, loop from loop counter is M\_ZERO to M\_TACH\_BIG\_BUFFER\_SIZE minus 1 and do the following:

- Set big buffer with reference of sample big buffer with indices (u8Chan, subtraction of (M\_TACH\_BIG\_BUFFER\_SIZE and M\_ONE)) when pre-decrement of big buffer is less than sample big buffer with index u8Chan, otherwise do nothing.

- Update the count value by adding it with u16Counts of big buffer

- Update the samples value by adding it with u16Samples of big buffer

- Stop the execution from current condition when count value is greater than or equal to u32Resolution of sample tach channel with index u8Chan AND loop counter is greater than minimum boundary value, otherwise do nothing.

##### 10.35.1.7.3 daugwytach-TachRead-LLR-003

Requirement ID: H398-LLD-GWY-FNC-4378

The function shall Send a request to uCOS to enable all the interrupt by calling the function RestoreStatusReg where U32\_critical\_sr is passed as parameter of this function .

##### 10.35.1.7.4 daugwytach-TachRead-LLR-004

Requirement ID: H398-LLD-GWY-FNC-4379

The function shall set the tach value as M\_ZERO when count value is less than u32Resolution of sample tach channel with index u8Chan, otherwise do the following and then return the tach value:

Set the tach value as (division of (multiplication of samples value with f32Scaler of sample tach channel with index u8Chan) by (multiplication of count value with M\_TACH\_US\_PER\_COUNT)), when count value is not zero

### 10.35.2 TachDriver

Low Level Design Details about CSU TachDriver will follow in the sub sections.

#### 10.35.2.1 Brief Description

The function TachDriver shall install the tach driver (Load the data from DMA buffer to Tach buffer).

#### 10.35.2.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD(H398-003-012-GWY).

#### 10.35.2.3 List of global variables accessed and modified

Accessed : Appdata\_appdata

Modified : None

#### 10.35.2.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.35.2.5 Return Value

None

#### 10.35.2.6 Other CSUs called by this CSU

TimClearITPendingBit

#### 10.35.2.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TachDriver.

##### 10.35.2.7.1 daugwytach-TachDriver-LLR-001

Requirement ID: H398-LLD-GWY-FNC-4388

The function shall call 'TimClearITPendingBit' with parameter (M\_TIM12, M\_TIM\_IT\_UPDATE) to Clear M\_TIM12 update interrupt pending bit

##### 10.35.2.7.2 daugwytach-TachDriver-LLR-002

Requirement ID: H398-LLD-GWY-FNC-4389

The function shall Read all the sensor reading by loop through all the tach channels (M\_ZERO to numofTach of Appdata\_appdata minus 1) and do the following:

- when the channel status (u8chanstate) of tach with index as loop counter of Appdata\_appdata is not equal to ENABLED do nothing

##### 10.35.2.7.3 daugwytach-TachDriver-LLR-003

Requirement ID: H398-LLD-GWY-FNC-6255

The function shall Read all the sensor reading by loop through all the tach channels (M\_ZERO to numofTach of Appdata\_appdata minus 1) and do the following:

⦁ Set next buffer location to m0ar of psDMA of Tach channel with index as loop counter added to (M\_TACH\_DMA\_BUFFER\_SIZE subtracted by ndtr of psDMA of Tach channel with index as loop counter).

⦁ Pre-decrement the next buffer location when it is lesser than DMA buffer to read the Tach channel with index as loop counter do the following:

⦁ Set next buffer location to reference of DMA buffer to read the Tach channel with indices as loop counter and (M\_TACH\_DMA\_BUFFER\_SIZE subtracted by M\_ONE).

⦁ Otherwise Do nothing.

##### 10.35.2.7.4 daugwytach-TachDriver-LLR-004

Requirement ID: H398-LLD-GWY-FNC-6256

The function shall Read all the sensor reading by loop through all the tach channels (M\_ZERO to numofTach of Appdata\_appdata minus 1) and do the following:

- when the channel status (u8chanstate) of tach with index as loop counter of Appdata\_appdata is equal to ENABLED

- Do the following when u8ChanStatus of DMA buffer to read the Tach channel with index as loop counter is equal to ENABLE:

⦁ Set u16Counts of psBigBuffer of DMA buffer to read the Tach channel with index as loop counter to (de-reference of next buffer location subtracted by dereference of pu16First of DMA buffer to read the Tach channel with index as loop counter) when de-reference of next buffer location is greater than or equal to dereference of pu16First of DMA buffer to read the Tach channel with index as loop counter.

⦁ Otherwise do the following:

⦁ Set u16Counts of psBigBuffer of DMA buffer to read the Tach channel with index as loop counter to (M\_WRAPPING\_FOR\_TIME\_COUNT subtracted by dereference of pu16First of DMA buffer to read the Tach channel with index as loop counter added to de-reference of next buffer location added to M\_ONE)

##### 10.35.2.7.5 daugwytach-TachDriver-LLR-005

Requirement ID: H398-LLD-GWY-FNC-6257

The function shall Read all the sensor reading by loop through all the tach channels (M\_ZERO to numofTach of Appdata\_appdata minus 1) and do the following:

- when the channel status (u8chanstate) of tach with index as loop counter of Appdata\_appdata is equal to ENABLED

- Do the following when u8ChanStatus of DMA buffer to read the Tach channel with index as loop counter is equal to ENABLE:

⦁ Set u16Samples of psBigBuffer of DMA buffer to read the Tach channel with index as loop counter to (next buffer location subtracted by pu16First of DMA buffer to read the Tach channel with index as loop counter) when next buffer location is greater than or equal to pu16First of DMA buffer to read the Tach channel with index as loop counter.

⦁ Otherwise do the following:

⦁ Set u16Samples of psBigBuffer of DMA buffer to read the Tach channel with index as loop counter to (next buffer location added to M\_TACH\_DMA\_BUFFER\_SIZE subtracted by pu16First of DMA buffer to read the Tach channel with index as loop counter).

##### 10.35.2.7.6 daugwytach-TachDriver-LLR-006

Requirement ID: H398-LLD-GWY-FNC-6258

The function shall Read all the sensor reading by loop through all the tach channels (M\_ZERO to numofTach of Appdata\_appdata minus 1) and do the following:

- when the channel status (u8chanstate) of tach with index as loop counter of Appdata\_appdata is equal to ENABLED

- Do the following when u8ChanStatus of DMA buffer to read the Tach channel with index as loop counter is not equal to ENABLE

⦁ Set u16Counts of psBigBuffer of DMA buffer to read the Tach channel with index as loop counter to M\_ZERO.

⦁ Set u16Samples of psBigBuffer of DMA buffer to read the Tach channel with index as loop counter to M\_ZERO.

##### 10.35.2.7.7 daugwytach-TachDriver-LLR-007

Requirement ID: H398-LLD-GWY-FNC-6259

The function shall Read all the sensor reading by loop through all the tach channels (M\_ZERO to numofTach of Appdata\_appdata minus 1) and do the following:

- when the channel status (u8chanstate) of tach with index as loop counter of Appdata\_appdata is equal to ENABLED

- Do the following when u8ChanStatus of DMA buffer to read the Tach channel with index as loop counter is not equal to ENABLE:

* Do the following when next buffer location is equal to pu16First of DMA buffer to read the Tach channel with index as loop counter:

⦁ Set u8ChanStatus of DMA buffer to read the Tach channel with index as loop counter to FALSE when pre-increment of u16Timeout of DMA buffer to read the Tach channel with index as loop counter is greater than or equal to M\_TACH\_DRIVER\_RATE.

⦁ Otherwise do nothing.

##### 10.35.2.7.8 daugwytach-TachDriver-LLR-008

Requirement ID: H398-LLD-GWY-FNC-6260

The function shall Read all the sensor reading by loop through all the tach channels (M\_ZERO to numofTach of Appdata\_appdata minus 1) and do the following:

- when the channel status (u8chanstate) of tach with index as loop counter of Appdata\_appdata is equal to ENABLED

- Do the following when u8ChanStatus of DMA buffer to read the Tach channel with index as loop counter is not equal to ENABLE

⦁ Set u16Timeout of DMA buffer to read the Tach channel with index as loop counter to M\_ZERO.

⦁ Set u8ChanStatus of DMA buffer to read the Tach channel with index as loop counter to TRUE.

##### 10.35.2.7.9 daugwytach-TachDriver-LLR-009

Requirement ID: H398-LLD-GWY-FNC-6261

The function shall Read all the sensor reading by loop through all the tach channels (M\_ZERO to numofTach of Appdata\_appdata minus 1) and do the following:

- when the channel status (u8chanstate) of tach with index as loop counter of Appdata\_appdata is equal to ENABLED

- Do the following when u8ChanStatus of DMA buffer to read the Tach channel with index as loop counter is not equal to ENABLE

* Set pu16First of DMA buffer to read the Tach channel with index as loop counter to next buffer location.

⦁ Set psBigBuffer of DMA buffer to read the Tach channel with index as loop counter to Big buffer to read the tach channel with index as loop counter when pre-incremented psBigBuffer of DMA buffer to read the Tach channel with index as loop counter is greater than or equal to reference of Big buffer to read the tach channel with indices as loop counter and M\_TACH\_BIG\_BUFFER\_SIZE.

⦁ Otherwise do nothing.

### 10.35.3 TachInit

Low Level Design Details about CSU TachInit will follow in the sub sections.

#### 10.35.3.1 Brief Description

The function TachInit shall initialize the tach channel and DMA buffer for the channel to start communication.

#### 10.35.3.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD(H398-003-012-GWY).

#### 10.35.3.3 List of global variables accessed and modified

Accessed : Appdata\_appdata

Modified : Appdata\_appdata

#### 10.35.3.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.35.3.5 Return Value

None

#### 10.35.3.6 Other CSUs called by this CSU

TmrInitTim12

InitTachChan

TachDriver

#### 10.35.3.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TachInit.

##### 10.35.3.7.1 daugwytach-TachInit-LLR-001

Requirement ID: H398-LLD-GWY-FNC-4398

The function shall Read all the sensor reading by loop through all the tach channels (M\_ZERO to numofTach of Appdata\_appdata) and do nothing when the u8ChanState of tach of Appdata\_appdata is not ENABLED for all the tach channels

##### 10.35.3.7.2 daugwytach-TachInit-LLR-002

Requirement ID: H398-LLD-GWY-FNC-4399

The function shall Read all the sensor reading by loop through all the tach channels (M\_ZERO to numofTach of Appdata\_appdata) and do the following when the u8ChanState of tach of Appdata\_appdata is ENABLED for all the tach channels and u16Scaled of tach of Appdata\_appdata for all the tach channels are not equal to M\_ZERO

* Calculate the maximum input frequency as division of (product of f32RefFreq of tach of Appdata\_appdata and (u16MaxRange of tach of Appdata\_appdata)) by u16Scaled of tach of Appdata\_appdata

##### 10.35.3.7.3 daugwytach-TachInit-LLR-004

Requirement ID: H398-LLD-GWY-FNC-4401

The function shall Read all the sensor reading by loop through all the tach channels (M\_ZERO to numofTach of Appdata\_appdata), loop from frequency division is M\_ONE to M\_TACH\_MAX\_DIV minus 1,update the frequency division to (bit shift of frequency division to left by M\_ONE) for each iteration and stop the execution from the current condition when

* u8ChanState of tach of Appdata\_appdata is ENABLED for all the tach channels
* frequency division is not M\_ZERO
* (((division of maximum input frequency by frequency division) is equal to M\_TACH\_5\_KHZ by calling the macro M\_FP\_COMPARE) OR (division of maximum input frequency by frequency division) is lesser than M\_TACH\_5\_KHZ ))

##### 10.35.3.7.4 daugwytach-TachInit-LLR-005

Requirement ID: H398-LLD-GWY-FNC-4402

The function shall Read all the sensor reading by loop through all the tach channels (M\_ZERO to numofTach of Appdata\_appdata), loop from frequency division is M\_ONE to M\_TACH\_MAX\_DIV-1,update the frequency division to (bit shift of frequency division to left by M\_ONE) for each iteration and do nothing when

* u8ChanState of tach of Appdata\_appdata is ENABLED for all the tach channels
* frequency division is not M\_ZERO
* ((division of maximum input frequency by frequency division) is greater than M\_TACH\_5\_KHZ)

##### 10.35.3.7.5 daugwytach-TachInit-LLR-006

Requirement ID: H398-LLD-GWY-FNC-4403

The function shall Read all the sensor reading by loop through all the tach channels (M\_ZERO to numofTach of Appdata\_appdata) and do the following

* Calculate the run time scaler by setting f32Scaler of tach channel with division of (product of u16Scaled of tach of Appdata\_appdata for all the tach channels and M\_SEC\_TO\_USEC\_UNIT) by (division of f32RefFreq of tach of Appdata\_appdata for all the tach channels by u8freq\_div)

when

* u8ChanState of tach of Appdata\_appdata is ENABLED for all the tach channels
* f32RefFreq of tach of Appdata\_appdata for all the tach channels is NOT equal to M\_ZERO by calling the macro M\_FP\_COMPARE.

##### 10.35.3.7.6 daugwytach-TachInit-LLR-007

Requirement ID: H398-LLD-GWY-FNC-4404

The function shall Read all the sensor reading by loop through all the tach channels (M\_ZERO to numofTach of Appdata\_appdata) and do the following

* Calculate resolution for a conversion by setting u32Resolution of tach channel with (product of M\_SAMPLING\_PERIOD and u16MaxRange of tach of Appdata\_appdata for all the channels)
* Initialize tach channel by calling function InitTachChan using tach channel as pointer with parameters (frequency division, buffer dma for all the tach channels, M\_TACH\_DMA\_BUFFER\_SIZE)

when

* u8ChanState of tach of Appdata\_appdata is ENABLED for all the tach channels

##### 10.35.3.7.7 daugwytach-TachInit-LLR-008

Requirement ID: H398-LLD-GWY-FNC-4405

The function shall Call the function ‘TmrInitTim12’ with parameter (dereference to Tach Driver) to Initialize timer 12 to drive tach channels

## 10.36 daugwytbase

The daugwytbase CSC contains implementation of timebase routines for signaling semaphores.

### 10.36.1 TbaseTaskSignaling

Low Level Design Details about CSU TbaseTaskSignaling will follow in the sub sections.

#### 10.36.1.1 Brief Description

The TbaseTaskSignaling function installs the following task signaling parameters into the timebase to be serviced

- Task ticks.

- Semaphore of the task

#### 10.36.1.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.36.1.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.36.1.4 Parameter list (Input/Output)

Inputs : T\_UINT16 u16\_task\_ticks Task ticks to delay this thread

T\_OS\_EVENT \* ps\_semaphore pointer to Semaphore to be posted for thread

Outputs : None

#### 10.36.1.5 Return Value

None

#### 10.36.1.6 Other CSUs called by this CSU

None

#### 10.36.1.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TbaseTaskSignaling

##### 10.36.1.7.1 daugwytbase-TbaseTaskSignaling-LLR-001

Requirement ID: H398-LLD-GWY-FNC-4415

The function shall return when the total number of installed tasks has reached the maximum limit of M\_OS\_MAX\_TASKS.

##### 10.36.1.7.2 daugwytbase-TbaseTaskSignaling-LLR-002

Requirement ID: H398-LLD-GWY-FNC-4416

The function shall reset the Tick Counter of the Timebase Task List to M\_ZERO

##### 10.36.1.7.3 daugwytbase-TbaseTaskSignaling-LLR-003

Requirement ID: H398-LLD-GWY-FNC-4417

The function shall save u16\_task\_ticks to the task ticks of the Timebase Task List

##### 10.36.1.7.4 daugwytbase-TbaseTaskSignaling-LLR-004

Requirement ID: H398-LLD-GWY-FNC-4418

The function shall save ps\_semaphore to the Semaphore of the Timebase Task List

##### 10.36.1.7.5 daugwytbase-TbaseTaskSignaling-LLR-005

Requirement ID: H398-LLD-GWY-FNC-4419

The function shall increment number of installed tasks by one

### 10.36.2 TbaseIntrHandler

Low Level Design Details about CSU TbaseIntrHandler will follow in the sub sections

#### 10.36.2.1 Brief Description

The TbaseIntrHandler function is the interrupt handler for timebase interrupt

#### 10.36.2.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY)

#### 10.36.2.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.36.2.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.36.2.5 Return Value

None

#### 10.36.2.6 Other CSUs called by this CSU

OsIntEnter

OsTimeTick

OsSemPost

OsIntExit

#### 10.36.2.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TbaseIntrHandler

##### 10.36.2.7.1 daugwytbase-TbaseIntrHandler-LLR-001

Requirement ID: H398-LLD-GWY-FNC-4428

The function shall inform the Kernel that, it is entering the ISR by calling OsIntEnter

##### 10.36.2.7.2 daugwytbase-TbaseIntrHandler-LLR-002

Requirement ID: H398-LLD-GWY-FNC-4429

The function shall update the time tick by calling OsTimeTick

##### 10.36.2.7.3 daugwytbase-TbaseIntrHandler-LLR-003

Requirement ID: H398-LLD-GWY-FNC-4430

The function shall loop through the Timebase Task List and perform the following for each task in the list when the task tick counter (incremented by 1) is greater than or equal to the task ticks of the corresponding task.

a) Reset the task tick counter to M\_ZERO.

b) Signal the Semaphore of the task by calling OsSemPost with Semaphore of the corresponding task as parameter, otherwise do nothing.

##### 10.36.2.7.4 daugwytbase-TbaseIntrHandler-LLR-004

Requirement ID: H398-LLD-GWY-FNC-4431

The function shall inform the Micro C OS, that it is leaving the ISR by calling OsIntExit

### 10.36.3 TbaseInit

Low Level Design Details about CSU TbaseInit will follow in the sub sections

#### 10.36.3.1 Brief Description

The TbaseInit function shall initialize the timebase

#### 10.36.3.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.36.3.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.36.3.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.36.3.5 Return Value

None

#### 10.36.3.6 Other CSUs called by this CSU

SysTickConfig

#### 10.36.3.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TbaseInit

##### 10.36.3.7.1 daugwytbase-TbaseInit-LLR-001

Requirement ID: H398-LLD-GWY-FNC-4440

The function shall initialize the System Tick Interrupt period by calling SysTickConfig with number of ticks between two interrupts as parameter (M\_HW\_SYSTEM\_CLOCK divided by M\_OS\_TICKS\_PER\_SEC).

## 10.37 daugwytmr

This file defines implementation of timer routines to trigger internal ADC

### 10.37.1 Timer7Intr

Low Level Design Details about CSU Timer7Intr will follow in the sub sections

#### 10.37.1.1 Brief Description

The function Timer7Intr shall initialize timer 7 interrupt

#### 10.37.1.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD(H398-003-012-GWY)

#### 10.37.1.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.37.1.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.37.1.5 Return Value

None

#### 10.37.1.6 Other CSUs called by this CSU

TimClearITPendingBit

SoundSynthesize

#### 10.37.1.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to Timer7Intr.

##### 10.37.1.7.1 daugwytmr-Timer7Intr-LLR-001

Requirement ID: H398-LLD-GWY-FNC-4450

The function shall call ‘TimClearITPendingBit’ with parameter

(M\_TIM7 , M\_TIM\_IT\_UPDATE) to clear the update interrupts and holds the data register (i.e dhr12r1)of M\_DAC by calling the function SoundSynthesize.

### 10.37.2 TmrInitDac

Low Level Design Details about CSU TmrInitDac will follow in the sub sections.

#### 10.37.2.1 Brief Description

The TmrInitDac function uses timer 6 to trigger internal ADC conversion

#### 10.37.2.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.37.2.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.37.2.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.37.2.5 Return Value

None

#### 10.37.2.6 Other CSUs called by this CSU

RccApb1PeriphClockCmd,

IntrInstall,

NvicInit,

TimPrescalerConfig,

TimSetAutoReload,

TimCmd,

TimItConfig,

Timer7Intr

#### 10.37.2.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TmrInitDac

##### 10.37.2.7.1 daugwytmr-TmrInitDac-LLR-001

Requirement ID: H398-LLD-GWY-FNC-4459

The function shall enable the Low Speed APB(APB1) Peripheral Clock by calling 'RccApb1PeriphClockCmd' with parameters M\_RCC\_APB1PERIPH\_TIM7, ENABLE.

##### 10.37.2.7.2 daugwytmr-TmrInitDac-LLR-002

Requirement ID: H398-LLD-GWY-FNC-4460

The function shall loads the ADC period interrupt into vector table by calling IntrInstall with parameters

INTR\_TIM\_7, Timer7Intr.

##### 10.37.2.7.3 daugwytmr-TmrInitDac-LLR-003

Requirement ID: H398-LLD-GWY-FNC-4461

The function shall initialize NVIC peripheral by calling 'NvicInit' with reference to nvic init structure as

a parameter with the members initialized as follows

a) nvic\_irq\_channel is set with TIM7\_IRQN,

b) nvic\_irq\_channel\_preempt\_pri is set with M\_ZERO,

c) nvic\_irq\_channel\_subpriority is set with M\_ZERO and

d) nvic\_irq\_channel\_cmd is set with ENABLE.

##### 10.37.2.7.4 daugwytmr-TmrInitDac-LLR-004

Requirement ID: H398-LLD-GWY-FNC-4462

The function shall configure prescaler for Timer 6 by calling 'TimPrescalerConfig' with parameters M\_TIM7, M\_TMR\_7\_SCALE subtract with M\_ONE and M\_TIM\_PSCRELOADMODE\_UPDATE.

##### 10.37.2.7.5 daugwytmr-TmrInitDac-LLR-005

Requirement ID: H398-LLD-GWY-FNC-4463

The function shall set the timer 6 counter reset value by function TimSetAutoReload with parameters M\_TIM7, M\_TMR\_7\_PERIOD minus M\_ONE.

##### 10.37.2.7.6 daugwytmr-TmrInitDac-LLR-006

Requirement ID: H398-LLD-GWY-FNC-4464

The function shall enable the timer 6 counter value by function TimCmd with parameters M\_TIM7, ENABLE.

##### 10.37.2.7.7 daugwytmr-TmrInitDac-LLR-007

Requirement ID: H398-LLD-GWY-FNC-4465

The function shall enable the UIF interrupt request by calling ‘TimItConfig’ with parameters M\_TIM7, M\_TIM\_IT\_UPDATE and ENABLE.

### 10.37.3 TmrInitTim6

Low Level Design Details about CSU TmrInitTim6 will follow in the sub sections.

#### 10.37.3.1 Brief Description

The TmrInitTim6 function uses Timer 6 to trigger internal ADC conversion.

#### 10.37.3.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.37.3.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.37.3.4 Parameter list (Input/Output)

Inputs : T\_INTR\_FN pfn\_trigger\_internal\_adc - Reference to function to be set as ISR in the vector for interrupt INTR\_TIM\_6\_ADC.

Outputs : None

#### 10.37.3.5 Return Value

None

#### 10.37.3.6 Other CSUs called by this CSU

RccApb1PeriphClockCmd,

IntrInstall,

NvicInit,

TimPrescalerConfig,

TimSetAutoReload,

Timcmd,

TimItConfig.

#### 10.37.3.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TmrInitTim6

##### 10.37.3.7.1 daugwytmr-TmrInitTim6-LLR-001

Requirement ID: H398-LLD-GWY-FNC-4474

The function shall enable the Low Speed APB(APB1) Peripheral Clock by calling 'RccApb1PeriphClockCmd' with parameters M\_RCC\_APB1PERIPH\_TIM6, ENABLE.

##### 10.37.3.7.2 daugwytmr-TmrInitTim6-LLR-002

Requirement ID: H398-LLD-GWY-FNC-4475

The function shall enable load the reference to the pfn\_trigger\_internal\_adc into the vector table for Timer 6 DAC interrupt by calling 'IntrInstall' with parameters INTR\_TIM\_6\_ADC and reference to pfn\_trigger\_internal\_adc.

##### 10.37.3.7.3 daugwytmr-TmrInitTim6-LLR-003

Requirement ID: H398-LLD-GWY-FNC-4476

The function shall initialize NVIC Peripheral by calling 'NvicInit' with reference to nvic init structure as a parameter with the members initialized as follows

a) nvic\_irq\_channel is set with TIM6\_DAC\_IRQN,

b) nvic\_irq\_channel\_preempt\_pri is set with M\_TEN,

c) nvic\_irq\_channel\_subpriority is set with M\_NVIC\_IRQ\_CHANNEL\_SUBPRIORITY0 and

d) nvic\_irq\_channel\_cmd is set with ENABLE

##### 10.37.3.7.4 daugwytmr-TmrInitTim6-LLR-004

Requirement ID: H398-LLD-GWY-FNC-4477

The function shall configure prescaler for timer 6 by calling 'TimPrescalerConfig' with parameters M\_TIM6, M\_TMR\_6\_SCALE minus M\_ONE and M\_TIM\_PSCRELOADMODE\_UPDATE

##### 10.37.3.7.5 daugwytmr-TmrInitTim6-LLR-005

Requirement ID: H398-LLD-GWY-FNC-4478

The function shall set the Timer 6 reset register value by calling 'TimSetAutoReload' with parameters M\_TIM6, and M\_TMR\_6\_PERIOD minus M\_ONE.

##### 10.37.3.7.6 daugwytmr-TmrInitTim6-LLR-006

Requirement ID: H398-LLD-GWY-FNC-4479

The function shall enable the Timer 6 counter by calling 'Timcmd' with parameters M\_TIM6, and ENABLE.

##### 10.37.3.7.7 daugwytmr-TmrInitTim6-LLR-007

Requirement ID: H398-LLD-GWY-FNC-4480

The function shall enable the UIE interrupt by calling 'TimItConfig' with parameters M\_TIM6, M\_TIM\_IT\_UPDATE and ENABLE

### 10.37.4 TmrInitTim12

Low Level Design Details about CSU TmrInitTim12 will follow in the sub sections.

#### 10.37.4.1 Brief Description

The function TmrInitTim12 shall initialize timer 12 for tach driver.

#### 10.37.4.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD(H398-003-012-GWY).

#### 10.37.4.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.37.4.4 Parameter list (Input/Output)

Inputs : T\_INTR\_FN pfnTachDriver -Reference of the function to Tach Driver

Outputs : None

#### 10.37.4.5 Return Value

None

#### 10.37.4.6 Other CSUs called by this CSU

RccApb1PeriphClockCmd,

IntrInstall,

NvicInit,

TimPrescalerConfig,

TimSetAutoReload,

TimCmd,

TimItConfig

#### 10.37.4.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TmrInitTim12.

##### 10.37.4.7.1 daugwytmr-TmrInitTim12-LLR-001

Requirement ID: H398-LLD-GWY-FNC-4489

The function shall call 'RccApb1PeriphClockCmd' with parameter

(M\_RCC\_APB1PERIPH\_TIM12, ENABLE) to enable the TIM 12 clock

##### 10.37.4.7.2 daugwytmr-TmrInitTim12-LLR-002

Requirement ID: H398-LLD-GWY-FNC-4490

The function shall call 'IntrInstall' with parameter (INTR\_TIM\_8\_BRK\_TIM\_12 , pfnTachDriver) to Install tach driver interrupt into vector table in RAM.

##### 10.37.4.7.3 daugwytmr-TmrInitTim12-LLR-003

Requirement ID: H398-LLD-GWY-FNC-4491

The function shall Call function NvicInit with parameter (Reference to NVIC init structure) ,to Initializes the NVIC peripheral (TIM 12 global Interrupt) to the value given below

- Set nvic\_irq\_channel to TIM8\_BRK\_TIM12\_IRQN

- Set nvic\_irq\_channel\_preempt\_pri to NVIC\_PRIORITY\_LEVEL\_4

- Set nvic\_irq\_channel\_subpriority to NVIC\_PRIORITY\_LEVEL\_0

- Set nvic\_irq\_channel\_cmd to ENABLE

##### 10.37.4.7.4 daugwytmr-TmrInitTim12-LLR-004

Requirement ID: H398-LLD-GWY-FNC-4492

The function shall call 'TimPrescalerConfig' with parameter (M\_TIM12, M\_TMR\_12\_SCALE minus M\_ONE, M\_TIM\_PSCRELOADMODE\_UPDATE) to Configure prescaler for timer 12.

##### 10.37.4.7.5 daugwytmr-TmrInitTim12-LLR-005

Requirement ID: H398-LLD-GWY-FNC-4493

The function shall call 'TimSetAutoReload' with parameter (M\_TIM12, M\_TMR\_12\_PERIOD minus M\_ONE) to Set timer 12 counter reset value.

##### 10.37.4.7.6 daugwytmr-TmrInitTim12-LLR-006

Requirement ID: H398-LLD-GWY-FNC-4494

The function shall call 'TimCmd' with parameter (M\_TIM12, ENABLE) to enable timer 12 counter.

##### 10.37.4.7.7 daugwytmr-TmrInitTim12-LLR-007

Requirement ID: H398-LLD-GWY-FNC-4495

The function shall call 'TimItConfig' with parameter (M\_TIM12, M\_TIM\_IT\_UPDATE, ENABLE) to enable the UIF Interrupt Request for timer 12.

### 10.37.5 GetDivConfig

Low Level Design Details about CSU GetDivConfig will follow in the sub sections.

#### 10.37.5.1 Brief Description

The function GetDivConfig shall Set the frequency to Capture Prescaler.

#### 10.37.5.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD(H398-003-012-GWY).

#### 10.37.5.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.37.5.4 Parameter list (Input/Output)

Inputs : T\_UINT8 u8FreqDiv - frequency divider

Outputs : None

#### 10.37.5.5 Return Value

T\_UINT16 ->Return frequency to Capture Prescaler

#### 10.37.5.6 Other CSUs called by this CSU

None

#### 10.37.5.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to GetDivConfig.

##### 10.37.5.7.1 daugwytmr-GetDivConfig-LLR-001

Requirement ID: H398-LLD-GWY-FNC-4504

The function shall return 'M\_TIM\_ICPSC\_DIV2' when the input frequency divider is equal to M\_TMR\_FREQ\_DIV\_2.

##### 10.37.5.7.2 daugwytmr-GetDivConfig-LLR-002

Requirement ID: H398-LLD-GWY-FNC-4505

The function shall return 'M\_TIM\_ICPSC\_DIV4' when the input frequency divider is M\_TMR\_FREQ\_DIV\_4.

##### 10.37.5.7.3 daugwytmr-GetDivConfig-LLR-003

Requirement ID: H398-LLD-GWY-FNC-4506

The function shall return 'M\_TIM\_ICPSC\_DIV8' when the input frequency divider is M\_TMR\_FREQ\_DIV\_8.

##### 10.37.5.7.4 daugwytmr-GetDivConfig-LLR-004

Requirement ID: H398-LLD-GWY-FNC-4507

The function shall return 'M\_TIM\_ICPSC\_DIV1' when the input frequency divider is other than M\_TMR\_FREQ\_DIV\_2, M\_TMR\_FREQ\_DIV\_4, M\_TMR\_FREQ\_DIV\_8.

### 10.37.6 TmrInitTach1

Low Level Design Details about CSU TmrInitTach1 will follow in the sub sections.

#### 10.37.6.1 Brief Description

The function TmrInitTach1 shall Initialize timer 8 channel 1 for tach 1, DMA channel and Tach 1 GPIO pins for communication.

#### 10.37.6.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD(H398-003-012-GWY).

#### 10.37.6.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.37.6.4 Parameter list (Input/Output)

Inputs : T\_UINT8 u8FreqDiv -> Frequency divider of the Tach channel

T\_UINT16 \*pu16BufferDMA -> Reference to the DMA buffer

T\_UINT16 u16BufferSize -> Size of the buffer

Outputs : None

#### 10.37.6.5 Return Value

None

#### 10.37.6.6 Other CSUs called by this CSU

RccAhb1PeriphClockCmd,

RccApb2PeriphClockCmd,

GpioPinAFConfig

GpioInit,

DmaDeInit,

DmaInit,

DmaCmd,

TimICInit,

TimDmaCmd,

TimPrescalerConfig,

TimCmd

GetDivConfig

#### 10.37.6.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TmrInitTach1.

##### 10.37.6.7.1 daugwytmr-TmrInitTach1-LLR-001

Requirement ID: H398-LLD-GWY-FNC-4516

The function shall call 'RccAhb1PeriphClockCmd' with parameter

(M\_RCC\_AHB1PERIPH\_DMA2, ENABLE) to Enable DMA2 clock

##### 10.37.6.7.2 daugwytmr-TmrInitTach1-LLR-002

Requirement ID: H398-LLD-GWY-FNC-4517

The function shall call 'RccApb2PeriphClockCmd' with parameter

(M\_RCC\_APB2PERIPH\_TIM8, ENABLE) to Enable TIM8 clock

##### 10.37.6.7.3 daugwytmr-TmrInitTach1-LLR-003

Requirement ID: H398-LLD-GWY-FNC-4518

The function shall call 'RccAhb1PeriphClockCmd' with parameter

(M\_RCC\_AHB1PERIPH\_GPIOC, ENABLE) to Enable GPIOC clock

##### 10.37.6.7.4 daugwytmr-TmrInitTach1-LLR-004

Requirement ID: H398-LLD-GWY-FNC-4519

The function shall call 'GpioPinAFConfig' with parameter (M\_GPIOC, M\_GPIO\_PINSOURCE6, M\_GPIO\_AF\_TIM8) to Configure pin function for tach 1.

##### 10.37.6.7.5 daugwytmr-TmrInitTach1-LLR-005

Requirement ID: H398-LLD-GWY-FNC-4520

The function shall Call function 'GpioInit' with parameter (M\_GPIOC, Reference to GPIO Init structure), to Configure tach 1 input GPIO pin to the value given below

- Set the gpio\_pin of Init structure to M\_GPIOC\_TACH\_1

- Set the gpio\_mode of Init structure to GPIO\_MODE\_AF

- Set the gpio\_speed of Init structure to GPIO\_SPEED\_50MHZ

- Set the gpio\_pupd of Init structure to GPIO\_PUPD\_NOPULL

##### 10.37.6.7.6 daugwytmr-TmrInitTach1-LLR-006

Requirement ID: H398-LLD-GWY-FNC-4521

The function shall call 'DmaDeInit' with parameter (M\_DMA2\_STREAM2) to Reset M\_DMA2 stream 2.

##### 10.37.6.7.7 daugwytmr-TmrInitTach1-LLR-007

Requirement ID: H398-LLD-GWY-FNC-4522

The function shall Call function 'DmaInit' with parameter (M\_DMA2\_STREAM2, Reference to DMA Init structure), to Configure DMA2 stream 2 channel 7 to the value given below

- Set the dma\_channel of Init structure to M\_DMA\_CHANNEL\_7

- Set the dma\_peripheral\_baseaddr of Init structure to address of capture/compare register 1 of timer 8(M\_TIM8)

- Set the dma\_memory0\_baseaddr of Init structure to Reference to DMA Buffer

- Set the dma\_dir of Init structure to M\_DMA\_DIR\_PERIPHERAL\_TO\_MEMORY

- Set the dma\_buffersize of Init structure to DMA Buffer Size

- Set the dma\_peripheral\_inc of Init structure to M\_DMA\_PERIPHERALINC\_DISABLE

- Set the dma\_memory\_inc of Init structure to M\_DMA\_MEMORYINC\_ENABLE

- Set the dma\_peripheral\_datasize of Init structure to M\_DMA\_PERIPH\_DATASIZE\_HALFWORD

- Set the dma\_memory\_datasize of Init structure to M\_DMA\_MEMORY\_DATASIZE\_HALFWORD

- Set the dma\_mode of Init structure to M\_DMA\_MODE\_CIRCULAR

- Set the dma\_priority of Init structure to M\_DMA\_PRIORITY\_HIGH

- Set the dma\_fifo\_mode of Init structure to M\_DMA\_FIFOMODE\_DISABLE

- Set the dma\_fifo\_threshold of Init structure to M\_DMA\_FIFOTHRESHOLD\_FULL

- Set the dma\_memory\_burst of Init structure to M\_DMA\_MEMORYBURST\_SINGLE

- Set the dma\_peripheral\_burst of Init structure to M\_DMA\_PERIPHERALBURST\_SINGLE

##### 10.37.6.7.8 daugwytmr-TmrInitTach1-LLR-008

Requirement ID: H398-LLD-GWY-FNC-4523

The function shall call 'DmaCmd' with parameter (M\_DMA2\_STREAM2, ENABLE) to Enable DMA2 stream 2.

##### 10.37.6.7.9 daugwytmr-TmrInitTach1-LLR-009

Requirement ID: H398-LLD-GWY-FNC-4524

The function shall Call function TimICInit with parameter (M\_TIM8, Reference to TIM Input Capture Init structure), to Configure channel 1 input of timer 8 to the value given below

- Set tim\_channel to M\_TIM\_CHANNEL\_1

- Set tim\_ic\_polarity to M\_TIM\_ICPOLARITY\_RISING

- Set tim\_ic\_selection to M\_TIM\_ICSELECTION\_DIRECTTI

- Set tim\_ic\_prescaler to the frequency to Capture Prescaler

(that is return value of the function GetDivConfig called with the parameter u8FreqDiv)

- Set tim\_ic\_filter to M\_TIM\_ICFILTER\_15

##### 10.37.6.7.10 daugwytmr-TmrInitTach1-LLR-010

Requirement ID: H398-LLD-GWY-FNC-4525

The function shall call 'TimDmaCmd' with parameter (M\_TIM8, M\_TIM\_DMA\_CC1, ENABLE) to enable DMA trigger for channel 1

##### 10.37.6.7.11 daugwytmr-TmrInitTach1-LLR-011

Requirement ID: H398-LLD-GWY-FNC-4526

The function shall call 'TimPrescalerConfig' with parameter (M\_TIM8, M\_TMR\_8\_SCALE minus M\_ONE, M\_TIM\_PSCRELOADMODE\_UPDATE) to Configure prescaler for timer 8.

##### 10.37.6.7.12 daugwytmr-TmrInitTach1-LLR-012

Requirement ID: H398-LLD-GWY-FNC-4527

The function shall call 'TimCmd' with parameter (M\_TIM8, ENABLE) to enable timer 8 counter

### 10.37.7 TmrInitTach2

Low Level Design Details about CSU TmrInitTach2 will follow in the sub sections.

#### 10.37.7.1 Brief Description

The function TmrInitTach2 shall initialize timer 8 channels 2 for tach 2, DMA channel and Tach 2 GPIO pins for communication.

#### 10.37.7.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD(H398-003-012-GWY).

#### 10.37.7.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.37.7.4 Parameter list (Input/Output)

Inputs : T\_UINT8 u8FreqDiv -> Frequency divider of the Tach channel

T\_UINT16 \*pu16bufferDMA -> Reference to the DMA buffer

T\_UINT16 u16BufferSize -> Size of the buffer

Outputs : None

#### 10.37.7.5 Return Value

None

#### 10.37.7.6 Other CSUs called by this CSU

RccAhb1PeriphClockCmd,

RccApb2PeriphClockCmd,

GpioPinAFConfig,

GpioInit,

DmaDeInit,

DmaInit,

DmaCmd,

TimICInit,

TimDmaCmd,

TimPrescalerConfig,

TimCmd

#### 10.37.7.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TmrInitTach2.

##### 10.37.7.7.1 daugwytmr-TmrInitTach2-LLR-001

Requirement ID: H398-LLD-GWY-FNC-4536

The function shall call 'RccAhb1PeriphClockCmd' with parameter (M\_RCC\_AHB1PERIPH\_DMA2, ENABLE) to Enable DMA2 clock.

##### 10.37.7.7.2 daugwytmr-TmrInitTach2-LLR-002

Requirement ID: H398-LLD-GWY-FNC-4537

The function shall call 'RccApb2PeriphClockCmd' with parameter

(M\_RCC\_APB2PERIPH\_TIM8, ENABLE) to Enable TIM8 clock

##### 10.37.7.7.3 daugwytmr-TmrInitTach2-LLR-003

Requirement ID: H398-LLD-GWY-FNC-4538

The function shall call 'RccAhb1PeriphClockCmd' with parameter

(M\_RCC\_AHB1PERIPH\_GPIOC, ENABLE) to Enable GPIOC clock.

##### 10.37.7.7.4 daugwytmr-TmrInitTach2-LLR-004

Requirement ID: H398-LLD-GWY-FNC-4539

The function shall call 'GpioPinAFConfig' with parameter (M\_GPIOC, M\_GPIO\_PINSOURCE7, M\_GPIO\_AF\_TIM8) to Configure pin function for tach 2.

##### 10.37.7.7.5 daugwytmr-TmrInitTach2-LLR-005

Requirement ID: H398-LLD-GWY-FNC-4540

The function shall Call function 'GpioInit' with parameter (M\_GPIOC ,Reference to GPIO Init structure) , to Configure tach 2 input GPIO pin to the value given below

- Set the gpio\_pin of Init structure to M\_GPIOC\_TACH\_2

- Set the gpio\_mode of Init structure to GPIO\_MODE\_AF

- Set the gpio\_speed of Init structure to GPIO\_SPEED\_50MHZ

- Set the gpio\_pupd of Init structure to GPIO\_PUPD\_NOPULL

##### 10.37.7.7.6 daugwytmr-TmrInitTach2-LLR-006

Requirement ID: H398-LLD-GWY-FNC-4541

The function shall call 'DmaDeInit' with parameter (M\_DMA2\_STREAM3) to Reset DMA2 stream 3.

##### 10.37.7.7.7 daugwytmr-TmrInitTach2-LLR-007

Requirement ID: H398-LLD-GWY-FNC-4542

The function shall Call function 'DmaInit' with parameter (M\_DMA2\_STREAM3 ,Reference to DMA Init structure) , to Configure DMA2 stream 3 channel 7 to the value given below

- Set the dma\_channel of Init structure to M\_DMA\_CHANNEL\_7

- Set the dma\_peripheral\_baseaddr of Init structure to address of capture/compare register 2 of timer 8(M\_TIM8)

- Set the dma\_memory0\_baseaddr of Init structure to Buffer DMA

- Set the dma\_dir of Init structure to M\_DMA\_DIR\_PERIPHERAL\_TO\_MEMORY

- Set the dma\_buffersize of Init structure to Buffer Size

- Set the dma\_peripheral\_inc of Init structure to M\_DMA\_PERIPHERALINC\_DISABLE

- Set the dma\_memory\_inc of Init structure to M\_DMA\_MEMORYINC\_ENABLE

- Set the dma\_peripheral\_datasize of Init structure to M\_DMA\_PERIPH\_DATASIZE\_HALFWORD

- Set the dma\_memory\_datasize of Init structure to M\_DMA\_MEMORY\_DATASIZE\_HALFWORD

- Set the dma\_mode of Init structure to M\_DMA\_MODE\_CIRCULAR

- Set the dma\_priority of Init structure to M\_DMA\_PRIORITY\_HIGH

- Set the dma\_fifo\_mode of Init structure to M\_DMA\_FIFOMODE\_DISABLE

- Set the dma\_fifo\_threshold of Init structure to M\_DMA\_FIFOTHRESHOLD\_FULL

- Set the dma\_memory\_burst of Init structure to M\_DMA\_MEMORYBURST\_SINGLE

- Set the dma\_peripheral\_burst of Init structure to M\_DMA\_PERIPHERALBURST\_SINGLE

##### 10.37.7.7.8 daugwytmr-TmrInitTach2-LLR-008

Requirement ID: H398-LLD-GWY-FNC-4543

The function shall call 'DmaCmd' with parameter (M\_DMA2\_STREAM3, ENABLE) to Enable DMA2 stream 3.

##### 10.37.7.7.9 daugwytmr-TmrInitTach2-LLR-009

Requirement ID: H398-LLD-GWY-FNC-4544

The function shall Call function TimICInit with parameter (M\_TIM8, Reference to TIM Input Capture Init structure), to Configure channel 2 input of timer 8 to the value given below

- Set tim\_channel to M\_TIM\_CHANNEL\_2

- Set tim\_ic\_polarity to M\_TIM\_ICPOLARITY\_RISING

- Set tim\_ic\_selection to M\_TIM\_ICSELECTION\_DIRECTTI

-Set tim\_ic\_prescaler to the frequency to Capture Prescaler

(that is return value of the function GetDivConfig called with the parameter u8FreqDiv)

- Set tim\_ic\_filter to M\_TIM\_ICFILTER\_15

##### 10.37.7.7.10 daugwytmr-TmrInitTach2-LLR-010

Requirement ID: H398-LLD-GWY-FNC-4545

The function shall call 'TimDmaCmd' with parameter (M\_TIM8, M\_TIM\_DMA\_CC2, ENABLE) to enable DMA trigger for channel 2.

##### 10.37.7.7.11 daugwytmr-TmrInitTach2-LLR-011

Requirement ID: H398-LLD-GWY-FNC-4546

The function shall call 'TimPrescalerConfig' with parameter (M\_TIM8,M\_TMR\_8\_SCALE minus M\_ONE M\_TIM\_PSCRELOADMODE\_UPDATE) to Configure prescaler for timer 8.

##### 10.37.7.7.12 daugwytmr-TmrInitTach2-LLR-012

Requirement ID: H398-LLD-GWY-FNC-4547

The function shall call 'TimCmd' with parameter (M\_TIM8, ENABLE) to enable timer 8 counter

### 10.37.8 TmrInitTach3

Low Level Design Details about CSU TmrInitTach3 will follow in the sub sections.

#### 10.37.8.1 Brief Description

The function TmrInitTach3 shall initialize timer 8 channels 3 for tach 3, DMA channel and Tach 3 GPIO pins for communication

#### 10.37.8.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD(H398-003-012-GWY).

#### 10.37.8.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.37.8.4 Parameter list (Input/Output)

Inputs : T\_UINT8 u8FreqDiv -> Frequency divider of the Tach channel

T\_UINT16 \*pu16BufferDMA -> Reference to the DMA buffer

T\_UINT16 u16BufferSize -> Size of the buffer

Outputs : None

11.36.8.5 Return Value

None

#### 10.37.8.5 Other CSUs called by this CSU

RccAhb1PeriphClockCmd,

RccApb2PeriphClockCmd,

GpioPinAFConfig,

GpioInit,

DmaDeInit,

DmaInit,

DmaCmd,

TimICInit,

TimDmaCmd,

TimPrescalerConfig,

TimCmd

#### 10.37.8.6 Description of list of LLRs allocated

The following section will list the LLRs allocated to TmrInitTach3

##### 10.37.8.6.1 daugwytmr-TmrInitTach3-LLR-001

Requirement ID: H398-LLD-GWY-FNC-4555

The function shall call 'RccAhb1PeriphClockCmd' with parameter

(M\_RCC\_AHB1PERIPH\_DMA2, ENABLE) to Enable DMA2 clock

##### 10.37.8.6.2 daugwytmr-TmrInitTach3-LLR-002

Requirement ID: H398-LLD-GWY-FNC-4556

The function shall call 'RccApb2PeriphClockCmd' with parameter

(M\_RCC\_APB2PERIPH\_TIM8, ENABLE) to Enable TIM8 clock.

##### 10.37.8.6.3 daugwytmr-TmrInitTach3-LLR-003

Requirement ID: H398-LLD-GWY-FNC-4557

The function shall call 'RccAhb1PeriphClockCmd' with parameter

(M\_RCC\_AHB1PERIPH\_GPIOC, ENABLE) to Enable GPIOC clock.

##### 10.37.8.6.4 daugwytmr-TmrInitTach3-LLR-004

Requirement ID: H398-LLD-GWY-FNC-4558

The function shall call 'GpioPinAFConfig' with parameter (M\_GPIOC, M\_GPIO\_PINSOURCE8, M\_GPIO\_AF\_TIM8) to Configure pin function for tach 3

##### 10.37.8.6.5 daugwytmr-TmrInitTach3-LLR-005

Requirement ID: H398-LLD-GWY-FNC-4559

The function shall Call function 'GpioInit' with parameter (M\_GPIOC ,Reference to GPIO Init structure) , to Configure tach 3 input GPIO pin to the value given below

- Set the gpio\_pin of Init structure to M\_GPIOC\_TACH\_3

- Set the gpio\_mode of Init structure to GPIO\_MODE\_AF

- Set the gpio\_speed of Init structure to GPIO\_SPEED\_50MHZ

- Set the gpio\_pupd of Init structure to GPIO\_PUPD\_NOPULL

##### 10.37.8.6.6 daugwytmr-TmrInitTach3-LLR-006

Requirement ID: H398-LLD-GWY-FNC-4560

The function shall call 'DmaDeInit' with parameter (M\_DMA2\_STREAM4) to Reset DMA2 stream 4

##### 10.37.8.6.7 daugwytmr-TmrInitTach3-LLR-007

Requirement ID: H398-LLD-GWY-FNC-4561

The function shall Call function 'DmaInit' with parameter (M\_DMA2\_STREAM4, Reference to DMA Init structure), to Configure DMA2 stream 4 channel 7 to the value given below

- Set the dma\_channel of Init structure to M\_DMA\_CHANNEL\_7

- Set the dma\_peripheral\_baseaddr of Init structure to address of capture/compare register 3 of timer 8(M\_TIM8)

- Set the dma\_memory0\_baseaddr of Init structure to Buffer DMA

- Set the dma\_dir of Init structure to M\_DMA\_DIR\_PERIPHERAL\_TO\_MEMORY

- Set the dma\_buffersize of Init structure to Buffer Size

- Set the dma\_peripheral\_inc of Init structure to M\_DMA\_PERIPHERALINC\_DISABLE

- Set the dma\_memory\_inc of Init structure to M\_DMA\_MEMORYINC\_ENABLE

- Set the dma\_peripheral\_datasize of Init structure to M\_DMA\_PERIPH\_DATASIZE\_HALFWORD

- Set the dma\_memory\_datasize of Init structure to M\_DMA\_MEMORY\_DATASIZE\_HALFWORD

- Set the dma\_mode of Init structure to M\_DMA\_MODE\_CIRCULAR

- Set the dma\_priority of Init structure to M\_DMA\_PRIORITY\_HIGH

- Set the dma\_fifo\_mode of Init structure to M\_DMA\_FIFOMODE\_DISABLE

- Set the dma\_fifo\_threshold of Init structure to M\_DMA\_FIFOTHRESHOLD\_FULL

- Set the dma\_memory\_burst of Init structure to M\_DMA\_MEMORYBURST\_SINGLE

- Set the dma\_peripheral\_burst of Init structure to M\_DMA\_PERIPHERALBURST\_SINGLE

##### 10.37.8.6.8 daugwytmr-TmrInitTach3-LLR-008

Requirement ID: H398-LLD-GWY-FNC-4562

The function shall call 'DmaCmd' with parameter (M\_DMA2\_STREAM4, ENABLE) to Enable DMA2 stream 4

##### 10.37.8.6.9 daugwytmr-TmrInitTach3-LLR-009

Requirement ID: H398-LLD-GWY-FNC-4563

The function shall Call function TimICInit with parameter (M\_TIM8, Reference to TIM Input Capture Init structure) to Configure channel 3 input of timer 8 to the value given below

- Set tim\_channel to M\_TIM\_CHANNEL\_3

- Set tim\_ic\_polarity to M\_TIM\_ICPOLARITY\_RISING

- Set tim\_ic\_selection to M\_TIM\_ICSELECTION\_DIRECTTI

- Set tim\_ic\_prescaler to the frequency to Capture Prescaler

(that is return value of the function GetDivConfig called with the parameter u8FreqDiv)

- Set tim\_ic\_filter to M\_TIM\_ICFILTER\_15

##### 10.37.8.6.10 daugwytmr-TmrInitTach3-LLR-010

Requirement ID: H398-LLD-GWY-FNC-4564

The function shall call 'TimDmaCmd' with parameter (M\_TIM8, M\_TIM\_DMA\_CC3, ENABLE) to enable DMA trigger for channel 3

##### 10.37.8.6.11 daugwytmr-TmrInitTach3-LLR-011

Requirement ID: H398-LLD-GWY-FNC-4565

The function shall call 'TimPrescalerConfig' with parameter (M\_TIM8, M\_TMR\_8\_SCALE minus M\_ONE, M\_TIM\_PSCRELOADMODE\_UPDATE) to Configure prescaler for timer 8

##### 10.37.8.6.12 daugwytmr-TmrInitTach3-LLR-012

Requirement ID: H398-LLD-GWY-FNC-4566

The function shall call 'TimCmd' with parameter (M\_TIM8, ENABLE) to enable timer 8 counter

### 10.37.9 TmrInitTach4

Low Level Design Details about CSU TmrInitTach4 will follow in the sub sections.

#### 10.37.9.1 Brief Description

The function TmrInitTach4 shall initialize timer 8 channels 4 for tach 4, DMA channel and Tach 4 GPIO pins for communication.

#### 10.37.9.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD(H398-003-012-GWY).

#### 10.37.9.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.37.9.4 Parameter list (Input/Output)

Inputs : T\_UINT8 u8FreqDiv -> Frequency divider of the Tach channel

T\_UINT16 \*pu16BufferDMA -> Reference to the DMA buffer

T\_UINT16 u16BufferSize -> Size of the buffer

Outputs : None

#### 10.37.9.5 Return Value

None

#### 10.37.9.6 Other CSUs called by this CSU

RccAhb1PeriphClockCmd,

RccApb2PeriphClockCmd,

GpioPinAFConfig,

GpioInit,

DmaDeInit,

DmaInit,

DmaCmd,

TimICInit,

TimDmaCmd,

TimPrescalerConfig,

TimCmd

#### 10.37.9.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TmrInitTach4.

##### 10.37.9.7.1 daugwytmr-TmrInitTach4-LLR-001

Requirement ID: H398-LLD-GWY-FNC-4575

The function shall call 'RccAhb1PeriphClockCmd' with parameter

(M\_RCC\_AHB1PERIPH\_DMA2, ENABLE) to Enable DMA2 clock.

##### 10.37.9.7.2 daugwytmr-TmrInitTach4-LLR-002

Requirement ID: H398-LLD-GWY-FNC-4576

The function shall call 'RccApb2PeriphClockCmd' with parameters (M\_RCC\_APB2PERIPH\_TIM8, ENABLE) to enable TIM8 clock

##### 10.37.9.7.3 daugwytmr-TmrInitTach4-LLR-003

Requirement ID: H398-LLD-GWY-FNC-4577

The function shall call 'RccAhb1PeriphClockCmd' with parameter

(M\_RCC\_AHB1PERIPH\_GPIOC, ENABLE) to Enable GPIOC clock.

##### 10.37.9.7.4 daugwytmr-TmrInitTach4-LLR-004

Requirement ID: H398-LLD-GWY-FNC-4578

The function shall call 'GpioPinAFConfig' with parameter (M\_GPIOC, M\_GPIO\_PINSOURCE9, M\_GPIO\_AF\_TIM8) to Configure pin function for tach 4.

##### 10.37.9.7.5 daugwytmr-TmrInitTach4-LLR-005

Requirement ID: H398-LLD-GWY-FNC-4579

The function shall Call function 'GpioInit' with parameter (M\_GPIOC, Reference to GPIO Init structure) to Configure tach 4 input GPIO pin to the value given below

- Set the gpio\_pin of Init structure to M\_GPIOC\_TACH\_4

- Set the gpio\_mode of Init structure to GPIO\_MODE\_AF

- Set the gpio\_speed of Init structure to GPIO\_SPEED\_50MHZ

- Set the gpio\_pupd of Init structure to GPIO\_PUPD\_NOPULL

##### 10.37.9.7.6 daugwytmr-TmrInitTach4-LLR-006

Requirement ID: H398-LLD-GWY-FNC-4580

The function shall call 'DmaDeInit' with parameter (M\_DMA2\_STREAM7) to Reset M\_DMA2 stream 7

##### 10.37.9.7.7 daugwytmr-TmrInitTach4-LLR-007

Requirement ID: H398-LLD-GWY-FNC-4581

The function shall Call function 'DmaInit' with parameter (M\_DMA2\_STREAM7, Reference to DMA Init structure) to Configure DMA2 stream 7 channel 7 to the value given below

- Set the dma\_channel of Init structure to M\_DMA\_CHANNEL\_7

- Set the dma\_peripheral\_baseaddr of Init structure to address of capture/compare register 4 of timer 8(M\_TIM8)

- Set the dma\_memory0\_baseaddr of Init structure to Buffer DMA

- Set the dma\_dir of Init structure to M\_DMA\_DIR\_PERIPHERAL\_TO\_MEMORY

- Set the dma\_buffersize of Init structure to Buffer Size

- Set the dma\_peripheral\_inc of Init structure to M\_DMA\_PERIPHERALINC\_DISABLE

- Set the dma\_memory\_inc of Init structure to M\_DMA\_MEMORYINC\_ENABLE

- Set the dma\_peripheral\_datasize of Init structure to M\_DMA\_PERIPH\_DATASIZE\_HALFWORD

- Set the dma\_memory\_datasize of Init structure to M\_DMA\_MEMORY\_DATASIZE\_HALFWORD

- Set the dma\_mode of Init structure to M\_DMA\_MODE\_CIRCULAR

- Set the dma\_priority of Init structure to M\_DMA\_PRIORITY\_HIGH

- Set the dma\_fifo\_mode of Init structure to M\_DMA\_FIFOMODE\_DISABLE

- Set the dma\_fifo\_threshold of Init structure to M\_DMA\_FIFOTHRESHOLD\_FULL

- Set the dma\_memory\_burst of Init structure to M\_DMA\_MEMORYBURST\_SINGLE

- Set the dma\_peripheral\_burst of Init structure to M\_DMA\_PERIPHERALBURST\_SINGLE

##### 10.37.9.7.8 daugwytmr-TmrInitTach4-LLR-008

Requirement ID: H398-LLD-GWY-FNC-4582

The function shall call 'DmaCmd' with parameter (M\_DMA2\_STREAM7, ENABLE) to Enable DMA2 stream 7

##### 10.37.9.7.9 daugwytmr-TmrInitTach4-LLR-009

Requirement ID: H398-LLD-GWY-FNC-4583

The function shall Call function TimICInit with parameter (M\_TIM8, Reference to TIM Input Capture Init structure), to Configure channel 4 input of timer 8 to the value given below

- Set tim\_channel to M\_TIM\_CHANNEL\_4

- Set tim\_ic\_polarity to M\_TIM\_ICPOLARITY\_RISING

- Set tim\_ic\_selection to M\_TIM\_ICSELECTION\_DIRECTTI

- Set tim\_ic\_prescaler to the frequency to Capture Prescaler

(that is return value of the function GetDivConfig called with the parameter u8FreqDiv)

- Set tim\_ic\_filter to M\_TIM\_ICFILTER\_15

##### 10.37.9.7.10 daugwytmr-TmrInitTach4-LLR-010

Requirement ID: H398-LLD-GWY-FNC-4584

The function shall call 'TimDmaCmd' with parameter (M\_TIM8, M\_TIM\_DMA\_CC4, ENABLE) to enable DMA trigger for channel 4.

##### 10.37.9.7.11 daugwytmr-TmrInitTach4-LLR-011

Requirement ID: H398-LLD-GWY-FNC-4585

The function shall call 'TimPrescalerConfig' with parameter (M\_TIM8, M\_TMR\_8\_SCALE minus M\_ONE, M\_TIM\_PSCRELOADMODE\_UPDATE) to Configure prescaler for timer 8

##### 10.37.9.7.12 daugwytmr-TmrInitTach4-LLR-012

Requirement ID: H398-LLD-GWY-FNC-4586

The function shall call 'TimCmd' with parameter (M\_TIM8, ENABLE) to enable timer 8 counter

### 10.37.10 TmrInitTach5

Low Level Design Details about CSU TmrInitTach5 will follow in the sub sections.

#### 10.37.10.1 Brief Description

The function TmrInitTach5 shall initialize timer 4 channels 1 for tach 5, DMA channel and Tach 5 GPIO pins for communication.

#### 10.37.10.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD(H398-003-012-GWY).

#### 10.37.10.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.37.10.4 Parameter list (Input/Output)

Inputs : T\_UINT8 u8FreqDiv -> Frequency divider of the Tach channel

T\_UINT16 \*pu16BufferDMA -> Reference to the DMA buffer

T\_UINT16 u16BufferSize -> Size of the buffer

Outputs : None

#### 10.37.10.5 Return Value

None

#### 10.37.10.6 Other CSUs called by this CSU

RccAhb1PeriphClockCmd,

RccApb1PeriphClockCmd,

GpioPinAFConfig,

GpioInit,

DmaDeInit,

DmaInit,

DmaCmd,

TimICInit,

TimDmaCmd,

TimPrescalerConfig,

TimCmd.

#### 10.37.10.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TmrInitTach5.

##### 10.37.10.7.1 daugwytmr-TmrInitTach5-LLR-001

Requirement ID: H398-LLD-GWY-FNC-4595

The function shall call 'RccAhb1PeriphClockCmd' with parameter

(M\_RCC\_AHB1PERIPH\_DMA1, ENABLE) to Enable DMA1 clock.

##### 10.37.10.7.2 daugwytmr-TmrInitTach5-LLR-002

Requirement ID: H398-LLD-GWY-FNC-4596

The function shall call 'RccApb1PeriphClockCmd' with parameters (M\_RCC\_APB1PERIPH\_TIM4, ENABLE) to enable TIM4 clock

##### 10.37.10.7.3 daugwytmr-TmrInitTach5-LLR-003

Requirement ID: H398-LLD-GWY-FNC-4597

The function shall call 'RccAhb1PeriphClockCmd' with parameter

(M\_RCC\_AHB1PERIPH\_GPIOB, ENABLE) to Enable GPIOB clock.

##### 10.37.10.7.4 daugwytmr-TmrInitTach5-LLR-004

Requirement ID: H398-LLD-GWY-FNC-4598

The function shall call 'GpioPinAFConfig' with parameter (M\_GPIOB, M\_GPIO\_PINSOURCE6, M\_GPIO\_AF\_TIM4) to Configure pin function for tach 5.

##### 10.37.10.7.5 daugwytmr-TmrInitTach5-LLR-005

Requirement ID: H398-LLD-GWY-FNC-4599

The function shall Call function 'GpioInit' with parameter (M\_GPIOB, Reference to GPIO Init structure) to Configure tach 5 input GPIO pin to the value given below

- Set the gpio\_pin of Init structure to M\_GPIOB\_TACH\_5

- Set the gpio\_mode of Init structure to GPIO\_MODE\_AF

- Set the gpio\_speed of Init structure to GPIO\_SPEED\_50MHZ

- Set the gpio\_pupd of Init structure to GPIO\_PUPD\_NOPULL

##### 10.37.10.7.6 daugwytmr-TmrInitTach5-LLR-006

Requirement ID: H398-LLD-GWY-FNC-4600

The function shall call 'DmaDeInit' with parameter (M\_DMA1\_STREAM0) to Reset M\_DMA1 stream 0.

##### 10.37.10.7.7 daugwytmr-TmrInitTach5-LLR-007

Requirement ID: H398-LLD-GWY-FNC-4601

The function shall Call function 'DmaInit' with parameter (M\_DMA1\_STREAM0, Reference to DMA Init structure) to Configure DMA1 stream 0 channel 2 to the value given below

- Set the dma\_channel of Init structure to M\_DMA\_CHANNEL\_2

- Set the dma\_peripheral\_baseaddr of Init structure to address of capture/compare register 1 of timer 4(M\_TIM4)

- Set the dma\_memory0\_baseaddr of Init structure to Buffer DMA

- Set the dma\_dir of Init structure to M\_DMA\_DIR\_PERIPHERAL\_TO\_MEMORY

- Set the dma\_buffersize of Init structure to Buffer Size

- Set the dma\_peripheral\_inc of Init structure to M\_DMA\_PERIPHERALINC\_DISABLE

- Set the dma\_memory\_inc of Init structure to M\_DMA\_MEMORYINC\_ENABLE

- Set the dma\_peripheral\_datasize of Init structure to M\_DMA\_PERIPH\_DATASIZE\_HALFWORD

- Set the dma\_memory\_datasize of Init structure to M\_DMA\_MEMORY\_DATASIZE\_HALFWORD

- Set the dma\_mode of Init structure to M\_DMA\_MODE\_CIRCULAR

- Set the dma\_priority of Init structure to M\_DMA\_PRIORITY\_HIGH

- Set the dma\_fifo\_mode of Init structure to M\_DMA\_FIFOMODE\_DISABLE

- Set the dma\_fifo\_threshold of Init structure to M\_DMA\_FIFOTHRESHOLD\_FULL

- Set the dma\_memory\_burst of Init structure to M\_DMA\_MEMORYBURST\_SINGLE

- Set the dma\_peripheral\_burst of Init structure to M\_DMA\_PERIPHERALBURST\_SINGLE

##### 10.37.10.7.8 daugwytmr-TmrInitTach5-LLR-008

Requirement ID: H398-LLD-GWY-FNC-4602

The function shall call 'DmaCmd' with parameter (M\_DMA1\_STREAM0, ENABLE) to Enable DMA1 stream 0.

##### 10.37.10.7.9 daugwytmr-TmrInitTach5-LLR-009

Requirement ID: H398-LLD-GWY-FNC-4603

The function shall Call function TimICInit with parameter (M\_TIM4, Reference to TIM Input Capture Init structure), to Configure channel 1 input of timer 4 to the value given below

- Set tim\_channel to M\_TIM\_CHANNEL\_1

- Set tim\_ic\_polarity to M\_TIM\_ICPOLARITY\_RISING

- Set tim\_ic\_selection to M\_TIM\_ICSELECTION\_DIRECTTI

- Set tim\_ic\_prescaler to the frequency to Capture Prescaler

(that is return value of the function GetDivConfig called with the parameter u8FreqDiv)

- Set tim\_ic\_filter to M\_TIM\_ICFILTER\_15

##### 10.37.10.7.10 daugwytmr-TmrInitTach5-LLR-010

Requirement ID: H398-LLD-GWY-FNC-4604

The function shall call 'TimDmaCmd' with parameter (M\_TIM4, M\_TIM\_DMA\_CC1, ENABLE) to enable DMA trigger for channel 1.

##### 10.37.10.7.11 daugwytmr-TmrInitTach5-LLR-011

Requirement ID: H398-LLD-GWY-FNC-4605

The function shall call 'TimPrescalerConfig' with parameter (M\_TIM4, M\_TMR\_4\_SCALE minus M\_ONE, M\_TIM\_PSCRELOADMODE\_UPDATE) to Configure prescaler for timer 4.

##### 10.37.10.7.12 daugwytmr-TmrInitTach5-LLR-012

Requirement ID: H398-LLD-GWY-FNC-4606

The function shall call 'TimCmd' with parameter (M\_TIM4, ENABLE) to enable timer 4 counter.

## 10.38 daugwyucos

The daugwyucos CSC defines the implementation of uC/OS routines for the real-time kernel.

### 10.38.1 OsInit

Low Level Design Details about CSU OsInit will follow in the sub sections.

#### 10.38.1.1 Brief Description

This is the uC/OS initialization function.

The OsInit function initializes the Os ready to run task list, TCB priority table list, list of free TCBs, list of free Event Control Blocks to default values. The function creates the OsTaskIdle function.

(Note: OSInit must be called before OSStart).

#### 10.38.1.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.38.1.3 List of global variables accessed and modified

Accessed : Os\_task\_idle\_stk

Modified : Os\_tcb\_high\_rdy,

Os\_tcb\_cur,

Os\_running

Os\_task\_idle\_stk

Os\_tcb\_prio\_tbl

#### 10.38.1.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.38.1.5 Return Value

None

#### 10.38.1.6 Other CSUs called by this CSU

OsTaskCreate

OsTaskIdle

#### 10.38.1.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to OsInit.

##### 10.38.1.7.1 daugwyucos-OsInit-LLR-001

Requirement ID: H398-LLD-GWY-FNC-4616

The function shall initialize the following:

- Current value of System Time to M\_ZERO

- Pointer to highest priority TCB ready to run (Os\_tcb\_high\_rdy) to M\_NULL.

- Pointer to currently running TCB (Os\_tcb\_cur) to M\_NULL.

- Pointer to list of TCBs to M\_NULL

- Interrupt nesting level, Multitasking lock nesting level to M\_ZERO.

- Flag indicating that Kernel is running(Os\_running) to FALSE.

- Counter of number of context switches to M\_ZERO.

-Ready list group to M\_ZERO.

- All the indices (until index is M\_RDY\_LST\_SIZE minus 1) of the table of tasks which are ready to run to M\_ZERO.

- All the indices (until index is M\_MAX\_TCB\_PRIO minus 1) of the Os\_tcb\_prio\_tbl to M\_NULL.

- os\_tcb\_next of each index of OS TCB list to the next element of OS TCB list (except the last element M\_OS\_MAX\_TASKS).

- os\_tcb\_next of last index of OS TCB list to M\_NULL.

- OS TCB free list to the base address of OS TCB list.

- Set the os\_eventptr of each index of 'Event Control Blocks list' to the next element of Event Control Blocks list (except the last element (M\_OS\_MAX\_EVENTS minus M\_ONE)).

- os\_eventptr of last index of Event Control Blocks list to M\_NULL.

- list of free EVENT control blocks to the base address of Event Control Blocks list.

##### 10.38.1.7.2 daugwyucos-OsInit-LLR-002

Requirement ID: H398-LLD-GWY-FNC-4617

The function shall initialize Os\_task\_idle\_stk with M\_CBIT\_TASK\_STK\_VAL by looping through M\_ZERO to M\_OS\_IDLE\_TASK\_STK\_SIZE.

##### 10.38.1.7.3 daugwyucos-OsInit-LLR-003

Requirement ID: H398-LLD-GWY-FNC-4618

The function shall create the Os Idle Task by calling 'OsTaskCreate' with parameters

a) function OsTaskIdle,

b) M\_NULL as task entry point,

c) Reference to top of stack of Os task idle stack (i.e. Os\_task\_idle\_stk)

d) M\_OS\_LOWEST\_PRIO as task priority.

### 10.38.2 OsTaskIdle

Low Level Design Details about CSU OsTaskIdle will follow in the sub sections.

#### 10.38.2.1 Brief Description

The OsTaskIdle function keeps track of the CPU idle time and reset the watch dog counter.

#### 10.38.2.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.38.2.3 List of global variables accessed and modified

Accessed : U32\_critical\_sr

Modified : U32\_critical\_sr

#### 10.38.2.4 Parameter list (Input/Output)

Inputs : void \* p\_data Not used always executed with null reference

Outputs : void \* p\_data Not used always executed with null reference

#### 10.38.2.5 Return Value

None

#### 10.38.2.6 Other CSUs called by this CSU

SaveStatusReg

RestoreStatusReg

WdogKickWatchDog

#### 10.38.2.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to OsTaskIdle.

##### 10.38.2.7.1 daugwyucos-OsTaskIdle-LLR-001

Requirement ID: H398-LLD-GWY-FNC-4627

The function shall do the following:

1. Set p\_data to p\_data added to M\_ZERO.
2. infinitely loop and perform the following operations

* Call ‘SaveStatusReg’ (M\_OS\_ENTER\_CRITICAL) to disable interrupts before incrementing the idle counter and set return value of ‘SaveStatusReg’ to U32\_critical\_sr.
* Call ‘WdogKickWatchDog’ to reset the watchdog timer.
* Call ‘RestoreStatusReg’ (M\_OS\_EXIT\_CRITICAL) with parameter U32\_critical\_sr to restore the Priority Mask Register after the idle counter increment is completed.

### 10.38.3 OsStart

Low Level Design Details about CSU OsStart will follow in the sub sections.

#### 10.38.3.1 Brief Description

The function OsStart starts the multitasking process, allowing uC/OS to manage

the tasks that have been created.

NOTE:

a) Before OsStart () is called, OsInit () has to be called and at least one task has to be created.

b) OSStart sets Os\_tcb\_high\_rdy to point to the Os\_tcb of the highest priority task.

#### 10.38.3.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.38.3.3 List of global variables accessed and modified

Accessed : Os\_tcb\_high\_rdy, Os\_tcb\_prio\_tbl

Modified : Os\_tcb\_high\_rdy, Os\_tcb\_cur, Os\_running

#### 10.38.3.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.38.3.5 Return Value

None

#### 10.38.3.6 Other CSUs called by this CSU

OsStartHighRdy

#### 10.38.3.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to OsStart.

##### 10.38.3.7.1 daugwyucos-OsStart-LLR-001

Requirement ID: H398-LLD-GWY-FNC-4636

The function shall

a) Sets second set of highest priority task priority number to priority resolution table with index as ready list group.

b) Sets first set of highest priority task priority number to priority resolution table with index as table of tasks which are ready to run.

c)Calculates priority value as ((second set of highest priority task priority number bit shifted to left by M\_THREE) added to the first set of highest priority task priority number)

d)Sets the Os\_tcb\_high\_rdy to point to the highest priority (determined in point c) task ready to run from the TCB priority table (i.e. Os\_tcb\_prio\_tbl).

##### 10.38.3.7.2 daugwyucos-OsStart-LLR-002

Requirement ID: H398-LLD-GWY-FNC-4637

The function shall set the pointer Os\_tcb\_cur (currently running TCB) to the highest priority TCB ready to run (Os\_tcb\_high\_rdy).

##### 10.38.3.7.3 daugwyucos-OsStart-LLR-003

Requirement ID: H398-LLD-GWY-FNC-4638

The function shall set the Os\_running to TRUE.

##### 10.38.3.7.4 daugwyucos-OsStart-LLR-004

Requirement ID: H398-LLD-GWY-FNC-4639

The function shall call the function 'OsStartHighRdy' to start running the highest priority task.

### 10.38.4 OsSched

Low Level Design Details about CSU OsSched will follow in the sub sections.

#### 10.38.4.1 Brief Description

The function OsSched does the Task-level scheduling.

#### 10.38.4.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.38.4.3 List of global variables accessed and modified

Accessed : Os\_tcb\_high\_rdy, Os\_tcb\_cur,U32\_critical\_sr, Os\_tcb\_prio\_tbl,

Modified : Os\_tcb\_high\_rdy,U32\_critical\_sr

#### 10.38.4.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.38.4.5 Return Value

None

#### 10.38.4.6 Other CSUs called by this CSU

SaveStatusReg,

RestoreStatusReg,

OSCtxSw

#### 10.38.4.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to OsSched.

##### 10.38.4.7.1 daugwyucos-OsSched-LLR-001

Requirement ID: H398-LLD-GWY-FNC-4648

The function shall call ‘SaveStatusReg’ (M\_OS\_ENTER\_CRITICAL) to disable interrupts before execution of the scheduler begins and stores return value of function SaveStatusReg’ in U32\_critical\_sr.

##### 10.38.4.7.2 daugwyucos-OsSched-LLR-002

Requirement ID: H398-LLD-GWY-FNC-4649

The function shall do the following when the task scheduling is enabled and not the ISR level i.e. check when either Interrupt nesting level bitwise OR with Multitasking lock nesting levels are zero.

- Set the high priority task index to the priority resolution table with index ready list of task

- Sets the 'Os\_tcb\_high\_rdy' to the highest priority task ready to run from the Os tcb priority table (i.e. Os\_tcb\_prio\_tbl with index as (highest priority task bit shifted to left by M\_THREE) added to the priority resolution table with index (table of tasks which are ready to run with index (highest priority task))).

##### 10.38.4.7.3 daugwyucos-OsSched-LLR-003

Requirement ID: H398-LLD-GWY-FNC-4650

The function shall do the following when either Interrupt nesting level or Multitasking lock nesting levels are zero:

a) When the task to be scheduled next 'Os\_tcb\_high\_rdy' is not the current running task 'Os\_tcb\_cur' do the following:

i) Increment context switch counter by one

ii) call OsCtxSw (by using M\_OS\_TASK\_SW) to perform context switch.

b) Otherwise do nothing.

##### 10.38.4.7.4 daugwyucos-OsSched-LLR-004

Requirement ID: H398-LLD-GWY-FNC-4651

The function shall do nothing when the Interrupt nesting level bitwise OR with Multitasking lock nesting levels are not equal to zero.

##### 10.38.4.7.5 daugwyucos-OsSched-LLR-005

Requirement ID: H398-LLD-GWY-FNC-4652

The function shall call the ‘RestoreStatusReg’ (M\_OS\_EXIT\_CRITICAL) with parameter U32\_critical\_sr to restore the priority mask register.

### 10.38.5 OsTcbInit

Low Level Design Details about CSU OsTcbInit will follow in the sub sections.

#### 10.38.5.1 Brief Description

This function Initializes the OS\_TCB - Each time a task is created, OsTcbInit ()

is called by OSTaskCreate ().

#### 10.38.5.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY)

#### 10.38.5.3 List of global variables accessed and modified

Accessed : U32\_critical\_sr,Os\_tcb\_prio\_tbl

Modified : U32\_critical\_sr

#### 10.38.5.4 Parameter list (Input/Output)

Inputs : T\_UBYTE prio Task priority

void \* stck Pointer to the top of stack

Outputs : None

#### 10.38.5.5 Return Value

T\_UBYTE - Return the status

M\_OS\_NO\_ERR - TCB created successfully.

M\_OS\_NO\_MORE\_TCB - No more free TCB in the TCB list

#### 10.38.5.6 Other CSUs called by this CSU

SaveStatusReg,

RestoreStatusReg

#### 10.38.5.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to OsTcbInit.

##### 10.38.5.7.1 daugwyucos-OsTcbInit-LLR-001

Requirement ID: H398-LLD-GWY-FNC-4661

The function shall call ‘SaveStatusReg’ (M\_OS\_ENTER\_CRITICAL) to disable interrupts before creation of TCB, return value of ‘SaveStatusReg’ gets stored in U32\_critical\_sr

##### 10.38.5.7.2 daugwyucos-OsTcbInit-LLR-002

Requirement ID: H398-LLD-GWY-FNC-4662

The function shall set a free TCB from the free TCB Os tcb free list.

##### 10.38.5.7.3 daugwyucos-OsTcbInit-LLR-003

Requirement ID: H398-LLD-GWY-FNC-4663

The function shall perform following operations when free TCB is obtained successfully from the free TCB list Os tcb free list (i.e not equal to M\_NULL).

a) Update the free TCB list with the pointer to the next TCB (os\_tcb\_next) in the free TCB list

b) Call ‘RestoreStatusReg’ (M\_OS\_EXIT\_CRITICAL) to restore the priority mask register with parameter U32\_critical\_sr

c) set the following members of the free TCB list

- pointer to current stack(os\_tcb\_stkptr) to stack pointer 'stck'.

- task priority (os\_tcb\_prio) to the current task prioirty 'prio'.

- task status (os\_tcb\_stat) to M\_OS\_STAT\_RDY.

- task delay (os\_tcb\_dly) to M\_ZERO.

- os\_tcb\_y to current task priority 'prio' right shifted by M\_THREE.

- os\_tcb\_bity to value of Mapping table with index as os\_tcb\_y of free TCB list structure.

- os\_tcb\_x to current task priority 'prio' bitwise AND with M\_HEX\_SEVEN.

- os\_tcb\_bitx to value of Mapping table with index as os\_tcb\_x of free TCB list structure.

- os\_tcb\_eventptr to M\_NULL.

- call ‘SaveStatusReg’ (M\_OS\_ENTER\_CRITICAL) to disable interrupts, return valueof ‘SaveStatusReg’ gets stored in U32\_critical\_sr.

d) set Os\_tcb\_prio\_tbl with index 'prio' to free TCB

e) set the free TCB member

- os\_tcb\_next to pointer to next TCB in the Pointer to list of TCBs.

- os\_tcb\_prev to M\_NULL

f) Set os\_tcb\_prev of Pointer to list of TCBs to free TCB list when Pointer to list of TCBs has valid data(i.e not equal to M\_NULL), otherwise do nothing.

g) Set the Pointer to list of TCBs to free TCB list.

h) Make the task ready to run by updating the following values

- Ready list group Bitwise OR with os\_tcb\_bity of free TCB

- Table of tasks which are ready to run with index os\_tcb\_y of free TCB to Table of tasks which are ready to run Bitwise OR with os\_tcb\_bitx of free TCB

##### 10.38.5.7.4 daugwyucos-OsTcbInit-LLR-004

Requirement ID: H398-LLD-GWY-FNC-4664

The function shall call ‘RestoreStatusReg’ (M\_OS\_EXIT\_CRITICAL) with parameter U32\_critical\_sr to restore the priority mask register after the successful creation of TCB is completed.

##### 10.38.5.7.5 daugwyucos-OsTcbInit-LLR-005

Requirement ID: H398-LLD-GWY-FNC-4665

The function shall return M\_OS\_NO\_ERR on successful creation of TCB.

##### 10.38.5.7.6 daugwyucos-OsTcbInit-LLR-006

Requirement ID: H398-LLD-GWY-FNC-4666

The function shall call ‘RestoreStatusReg’ (M\_OS\_EXIT\_CRITICAL) with parameter U32\_critical\_sr to restore the priority mask register when the list of free TCB is equal to M\_NULL.

##### 10.38.5.7.7 daugwyucos-OsTcbInit-LLR-007

Requirement ID: H398-LLD-GWY-FNC-4667

The function shall return M\_OS\_NO\_MORE\_TCB when the list of free TCB is equal to M\_NULL.

### 10.38.6 OsIntEnter

Low Level Design Details about CSU OsIntEnter will follow in the sub sections.

#### 10.38.6.1 Brief Description

The function OsIntEnter increments ISR nesting level.

#### 10.38.6.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.38.6.3 List of global variables accessed and modified

Accessed : U32\_critical\_sr

Modified : U32\_critical\_sr

#### 10.38.6.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.38.6.5 Return Value

None

#### 10.38.6.6 Other CSUs called by this CSU

SaveStatusReg,

RestoreStatusReg

#### 10.38.6.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to OsIntEnter.

##### 10.38.6.7.1 daugwyucos-OsIntEnter-LLR-001

Requirement ID: H398-LLD-GWY-FNC-4676

The function shall do the following on entering an ISR

- Send a request to uC/OS to disable all the interrupts by calling the function ‘SaveStatusReg’ (M\_OS\_ENTER\_CRITICAL) and return value of ‘SaveStatusReg’ gets stored in U32\_critical\_sr.

- Increment ISR nesting level counter by one.

- Send a request to uC/OS to restore the priority mask register by calling ‘RestoreStatusReg’ (M\_OS\_EXIT\_CRITICAL) with parameter U32\_critical\_sr.

### 10.38.7 OsIntExit

Low Level Design Details about CSU OsIntExit will follow in the sub sections.

#### 10.38.7.1 Brief Description

The function OsIntExit decrements ISR nesting level and perform interrupt level context switch.

#### 10.38.7.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.38.7.3 List of global variables accessed and modified

Accessed : Os\_tcb\_prio\_tbl

Os\_tcb\_high\_rdy

Os\_tcb\_cur

U32\_critical\_sr

Modified : Os\_tcb\_high\_rdy

U32\_critical\_sr

#### 10.38.7.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.38.7.5 Return Value

None

#### 10.38.7.6 Other CSUs called by this CSU

OsIntCtxSw,

SaveStatusReg,

RestoreStatusReg

#### 10.38.7.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to OsIntExit

##### 10.38.7.7.1 daugwyucos-OsIntExit-LLR-001

Requirement ID: H398-LLD-GWY-FNC-4685

The function shall call ‘SaveStatusReg’ (M\_OS\_ENTER\_CRITICAL) to disable interrupts before execution of the function and return value of ‘SaveStatusReg’ gets stored in U32\_critical\_sr.

##### 10.38.7.7.2 daugwyucos-OsIntExit-LLR-002

Requirement ID: H398-LLD-GWY-FNC-4686

The function shall do the following when the ISR is complete and not locked i.e. check when either ((the pre decremented Interrupt nesting level) OR Multitasking lock nesting level) are M\_ZERO.

- Set OS init exit to the priority of the highest priority task that is ready to run from the priority resolution table

- Set the 'Os\_tcb\_high\_rdy' to the highest priority task ready to run from the 'Os\_tcb\_prio\_tbl' with index ((OS init exit bit shifted to left by M\_THREE) added to priority resolution table with index (table of task which are ready to run with index (OS init exit)))

##### 10.38.7.7.3 daugwyucos-OsIntExit-LLR-003

Requirement ID: H398-LLD-GWY-FNC-4687

The function shall perform the following operations when the ISR is complete and not locked check when either ((the pre decremented Interrupt nesting level) bitwise OR Multitasking lock nesting level)):

a) When the task to be run next 'Os\_tcb\_high\_rdy' is not the current running task 'Os\_tcb\_cur’.

i) Increment context switch counter by one.

ii) call ‘OsIntCtxSw’ to perform interrupt level context switch.

b) Otherwise do nothing.

##### 10.38.7.7.4 daugwyucos-OsIntExit-LLR-004

Requirement ID: H398-LLD-GWY-FNC-4688

The function shall do nothing when the pre decremented Interrupt nesting level bitwise OR Multitasking lock nesting level are not equal to zero.

##### 10.38.7.7.5 daugwyucos-OsIntExit-LLR-005

Requirement ID: H398-LLD-GWY-FNC-4689

The function shall call the ‘RestoreStatusReg’ (M\_OS\_EXIT\_CRITICAL) with parameter U32\_critical\_sr to restore the priority mask register.

### 10.38.8 OsTimeDly

Low Level Design Details about CSU OsTimeDly will follow in the sub sections.

#### 10.38.8.1 Brief Description

The function OsTimeDly allows the calling task to delay for a specific number of clock ticks.

#### 10.38.8.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.38.8.3 List of global variables accessed and modified

Accessed : Os\_tcb\_cur

U32\_critical\_sr

Modified : Os\_tcb\_cur

U32\_critical\_sr

#### 10.38.8.4 Parameter list (Input/Output)

Inputs : T\_UWORD ticks No of Delay clock ticks

Outputs : None

#### 10.38.8.5 Return Value

None

#### 10.38.8.6 Other CSUs called by this CSU

SaveStatusReg,

RestoreStatusReg,

OsSched

#### 10.38.8.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to OsTimeDly.

##### 10.38.8.7.1 daugwyucos-OsTimeDly-LLR-001

Requirement ID: H398-LLD-GWY-FNC-4698

The function shall do the following when 'ticks' is greater than M\_ZERO

- Send a request to uC/OS to disable all the interrupts by calling function ‘SaveStatusReg’ (M\_OS\_ENTER\_CRITICAL) and return value of ‘SaveStatusReg’ gets stored in U32\_critical\_sr.

- Set Table of task which are ready to run with index os\_tcb\_y of Os\_tcb\_cur to (Table of task which are ready to run with index os\_tcb\_y of Os\_tcb\_cur bitwise AND with (Negation of (os\_tcb\_bitx of Os\_tcb\_cur)))

- Set Os ready group to (os ready group bitwise AND with (Negation of os\_tcb\_bity of Os\_tcb\_cur))

when table of task which are ready to run with index os\_tcb\_y of Os\_tcb\_cur is M\_ZERO, otherwise do nothing.

- set os\_tcb\_dly of the TCB current task (Os\_tcb\_cur) to 'ticks' to indicate that the task is delayed.

- Send a request to uC/OS to enable all the interrupts by calling function ‘RestoreStatusReg’ (M\_OS\_EXIT\_CRITICAL) with parameter U32\_critical\_sr.

- Call the function 'OsSched' to run the next high priority task as this task has been delayed.

##### 10.38.8.7.2 daugwyucos-OsTimeDly-LLR-002

Requirement ID: H398-LLD-GWY-FNC-4699

The function shall do nothing when 'ticks' is less than or equal to zero

### 10.38.9 OsTimeTick

Low Level Design Details about CSU OsTimeTick will follow in the sub sections

#### 10.38.9.1 Brief Description

The function OsTimeTick processes the clock tick for the task or the ISR. Check all the task to see when they are either waiting for time to expire (Call to OsTimeDly ()) or waiting for events to occur until they timeout

#### 10.38.9.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.38.9.3 List of global variables accessed and modified

Accessed : U32\_critical\_sr

Modified : U32\_critical\_sr

#### 10.38.9.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.38.9.5 Return Value

None

#### 10.38.9.6 Other CSUs called by this CSU

SaveStatusReg,

RestoreStatusReg

#### 10.38.9.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to OsTimeTick.

##### 10.38.9.7.1 daugwyucos-OsTimeTick-LLR-001

Requirement ID: H398-LLD-GWY-FNC-4708

The function shall loop through all TCBs in TCB list till the task priority for the TCB is M\_OS\_LOWEST\_PRIO and does the following:

* Send a request to uC/OS to disable all the interrupts by calling ‘SaveStatusReg’ (M\_OS\_ENTER\_CRITICAL) and return value of ‘SaveStatusReg’ gets stored in U32\_critical\_sr.
* Do the following when number of ticks to os\_tcb\_dly for the TCB is not Zero:

1. Make task Ready to Run (i.e Set Os ready group to (os ready group bitwise OR with os\_tcb\_bity of TCB) and set table of task which are ready to run with index (os\_tcb\_y of TCB) to (table of task which are ready to run with index (os\_tcb\_y of TCB) bitwise OR with os\_tcb\_bitx of TCB)) when number of ticks os\_tcb\_dly for the TCB is Zero and decremented value of number of ticks os\_tcb\_dly for the TCB is M\_ZERO and (Negation of (task status os\_tcb\_stat for the TCB Bitwise AND with (M\_OS\_STAT\_SUSPEND))) is not equal to FALSE.
2. Set the number of ticks os\_tcb\_dly for the TCB to M\_ONE when number of ticks os\_tcb\_dly for the TCB is Zero and decremented value of number of ticks os\_tcb\_dly for the TCB is M\_ZERO and (Negation of (task status os\_tcb\_stat for the TCB Bitwise AND with (M\_OS\_STAT\_SUSPEND))) is not equal to FALSE.
3. Do nothing when decremented number of ticks os\_tcb\_dly for the TCB is not equal to M\_ZERO.

* Do nothing when number of ticks to os\_tcb\_dly for the TCB is M\_ZERO.
* Send a request to uC/OS to enable all the interrupts by calling ‘RestoreStatusReg’ (M\_OS\_EXIT\_CRITICAL) with parameter U32\_critical\_sr.
* Point the current TCB to next TCB os\_tcb\_next

##### 10.38.9.7.2 daugwyucos-OsTimeTick-LLR-002

Requirement ID: H398-LLD-GWY-FNC-4709

The function shall send a request to uC/OS to disable all the interrupts by calling ‘SaveStatusReg’ (M\_OS\_ENTER\_CRITICAL) and return value of ‘SaveStatusReg’ gets stored in U32\_critical\_sr

##### 10.38.9.7.3 daugwyucos-OsTimeTick-LLR-003

Requirement ID: H398-LLD-GWY-FNC-4710

The function shall increment the Current value of system time by one

##### 10.38.9.7.4 daugwyucos-OsTimeTick-LLR-004

Requirement ID: H398-LLD-GWY-FNC-4711

The function shall send a request to uC/OS to enable all the interrupts by calling ‘RestoreStatusReg’ (M\_OS\_EXIT\_CRITICAL) with parameter U32\_critical\_sr when os\_eventgrp of ECB p\_event equal to Zero AND os\_eventcnt of ECB p\_event is greater than M\_MAX\_SEMPHORE

### 10.38.10 OsSemCreate

Low Level Design Details about CSU OsSemCreate will follow in the sub sections

#### 10.38.10.1 Brief Description

The function OsSemCreate creates and initialize Semaphore

#### 10.38.10.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.38.10.3 List of global variables accessed and modified

Accessed : U32\_critical\_sr

Modified : U32\_critical\_sr

#### 10.38.10.4 Parameter list (Input/Output)

Inputs : T\_UWORD count The initial value of the semaphore

Outputs : None

#### 10.38.10.5 Return Value

T\_OS\_EVENT\* Return Pointer to the event control block (ECB) associated with

the created semaphore.

#### 10.38.10.6 Other CSUs called by this CSU

SaveStatusReg,

RestoreStatusReg

#### 10.38.10.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to OsSemCreate

##### 10.38.10.7.1 daugwyucos-OsSemCreate-LLR-001

Requirement ID: H398-LLD-GWY-FNC-4720

The function shall call ‘SaveStatusReg’ (M\_OS\_ENTER\_CRITICAL) to disable interrupts and return value of ‘SaveStatusReg’ gets stored in U32\_critical\_sr

##### 10.38.10.7.2 daugwyucos-OsSemCreate-LLR-002

Requirement ID: H398-LLD-GWY-FNC-4721

The function shall store an ECB from free list of ECBs

##### 10.38.10.7.3 daugwyucos-OsSemCreate-LLR-003

Requirement ID: H398-LLD-GWY-FNC-4722

The function shall set the list of free ECB to next free list of ECB (os\_eventptr) when the the list of free ECB is not M\_NULL, otherwise do nothing.

##### 10.38.10.7.4 daugwyucos-OsSemCreate-LLR-004

Requirement ID: H398-LLD-GWY-FNC-4723

The function shall enable all the interrupts by calling ‘RestoreStatusReg’ (M\_OS\_EXIT\_CRITICAL) with parameter U32\_critical\_sr

##### 10.38.10.7.5 daugwyucos-OsSemCreate-LLR-005

Requirement ID: H398-LLD-GWY-FNC-4724

The function shall do the following when the reference to the selected ECB is not equal to M\_NULL, otherwise return M\_NULL.

- Set the desired initial count (count) for the Semaphore into os\_eventcnt of obtained ECB.

- Clear the os\_eventgrp by setting it to M\_HEX2\_ZERO

-Loop through (until M\_MAX\_TSK\_WAIT\_EVNT minus 1) to clear os\_event\_tbl of the obtained ECB by setting it to M\_HEX2\_ZERO.

- Return with the reference to the obtained ECB.

### 10.38.11 OsSemPend

Low Level Design Details about CSU OsSemPend will follow in the sub sections

#### 10.38.11.1 Brief Description

The function OsSemPend waits on a semaphore.

#### 10.38.11.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.38.11.3 List of global variables accessed and modified

Accessed : U32\_critical\_sr

Os\_tcb\_cur

Modified : U32\_critical\_sr

Os\_tcb\_cur

#### 10.38.11.4 Parameter list (Input/Output)

Inputs: T\_OS\_EVENT \*p\_event Pointer to the desired semaphore's ECB

T\_UWORD timeout Time in clock ticks to wait for the resource.

If 0, the task will wait until the resource becomes available or the

event occurs.

Outputs: T\_OS\_EVENT \*p\_event Pointer to the desired semaphore's ECB

T\_UBYTE \*error Pointer to error message.

- Set to M\_OS\_NO\_ERR if the semaphore is available.

- Set to M\_OS\_TIMEOUT if the semaphore is not signaled

within the specified timeout

#### 10.38.11.5 Return Value

None

#### 10.38.11.6 Other CSUs called by this CSU

SaveStatusReg,

RestoreStatusReg,

OsSched

#### 10.38.11.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to OsSemPend.

##### 10.38.11.7.1 daugwyucos-OsSemPend-LLR-001

Requirement ID: H398-LLD-GWY-FNC-4733

The function shall call ‘SaveStatusReg’ (M\_OS\_ENTER\_CRITICAL) to disable interrupts and return value of ‘SaveStatusReg’ gets stored in U32\_critical\_sr

##### 10.38.11.7.2 daugwyucos-OsSemPend-LLR-002

Requirement ID: H398-LLD-GWY-FNC-4734

The function shall do the following when the event counter of the current TCB (os\_eventcnt of p\_event) is greater than Zero (when semaphore is positive, resource is available).

- Decrement the Semaphore counter os\_eventcnt of the current TCB(p\_event) by one

- Send a request to uC/OS to enable all the interrupts by calling ‘RestoreStatusReg’ (M\_OS\_EXIT\_CRITICAL) with parameter U32\_critical\_sr.

- Set the pointer/ derefrence error message to M\_OS\_NO\_ERR.

##### 10.38.11.7.3 daugwyucos-OsSemPend-LLR-003

Requirement ID: H398-LLD-GWY-FNC-4735

The function shall do the following when the event counter (os\_eventcnt) of the current TCB p\_event is less than or equal to Zero (when semaphore is negative, resource is not available):

* Set os\_tcb\_stat of Os\_tcb\_cur to os\_tcb\_stat Os\_tcb\_cur bitwise OR with M\_OS\_STAT\_SEM.
* Set the number of ticks os\_tcb\_dly of currently running TCB Os\_tcb\_cur to value of timeout.
* Set TCB event pointer os\_tcb\_eventptr of currently running TCB Os\_tcb\_cur to ECB location of the Semaphore(i.e.p\_event)
* Set Table of task which are ready to run with index os\_tcb\_y of Os\_tcb\_cur to (Table of task which are ready to run with index os\_tcb\_y of Os\_tcb\_cur bitwise AND with (Negation of (os\_tcb\_bitx of Os\_tcb\_cur))).
* Set Os ready group to (Os ready group bitwise AND (Negation of os\_tcb\_bity of Os\_tcb\_cur)) when table of task which are ready to run with index os\_tcb\_y of Os\_tcb\_cur is M\_ZERO, otherwise do nothing.

##### 10.38.11.7.4 daugwyucos-OsSemPend-LLR-004

Requirement ID: H398-LLD-GWY-FNC-6262

The function shall do the following:

a) Set os\_event\_tbl of ECB p\_event with index (os\_tcb\_y of Os\_tcb\_cur) to (os\_event\_tbl of ECB p\_event with index (os\_tcb\_y of Os\_tcb\_cur) bitwise OR with os\_tcb\_bitx of Os\_tcb\_cur).

b) Set os\_eventgrpof ECB p\_event to (os\_eventgrp of ECB p\_event bitwise OR with os\_tcb\_bity of Os\_tcb\_cur).

⦁ Send a request to uC/OS to enable all the interrupts by calling ‘RestoreStatusReg’ (M\_OS\_EXIT\_CRITICAL) with parameter U32\_critical\_sr.

⦁ Call the function 'OsSched' to schedule the next highest priority task.

⦁ Send a request to uC/OS to disable all the interrupts by calling ‘SaveStatusReg’ (M\_OS\_ENTER\_CRITICAL) and return value of ‘SaveStatusReg’ gets stored in U32\_critical\_sr.

##### 10.38.11.7.5 daugwyucos-OsSemPend-LLR-005

Requirement ID: H398-LLD-GWY-FNC-6263

The function shall do the following when os\_tcb\_stat of Os\_tcb\_cur bitwise AND with M\_OS\_STAT\_SEM is not equal to FALSE:

a) Set os\_event\_tbl of p\_event with index os\_tcb\_y of Os\_tcb\_cur to (os\_event\_tbl of p\_event with index os\_tcb\_y of Os\_tcb\_cur bitwise AND with (Negation of (os\_tcb\_bitx of Os\_tcb\_cur))).

b) Set os\_eventgrp of p\_event to (os\_eventgrp of p\_event Bitwise AND with (Negation of (os\_tcb\_bity of Os\_tcb\_cur))) when os\_event\_tbl of p\_event is M\_ZERO, otherwise do nothing.

c) Set Task status os\_tcb\_stat of current running TCB Os\_tcb\_cur to 'M\_OS\_STAT\_RDY'.

d) Set the Event pointer os\_tcb\_eventptr of currently running TCB Os\_tcb\_cur to M\_NULL.

e) Send a request to uC/OS to enable all the interrupts by calling ‘RestoreStatusReg’ (M\_OS\_EXIT\_CRITICAL) with parameter U32\_critical\_sr.

f) Set the error message pointer (i.e.- \*error) to M\_OS\_TIMEOUT

##### 10.38.11.7.6 daugwyucos-OsSemPend-LLR-006

Requirement ID: H398-LLD-GWY-FNC-6264

The function shall do the following when os\_tcb\_stat of Os\_tcb\_cur bitwise AND with M\_OS\_STAT\_SEM is equal to FALSE:

a) Set the Event pointer os\_tcb\_eventptr of current running TCB Os\_tcb\_cur to M\_NULL.

b) Send a request to uC/OS to enable all the interrupts by calling ‘RestoreStatusReg’ (M\_OS\_EXIT\_CRITICAL) with parameter U32\_critical\_sr.

c) Set the error message to M\_OS\_NO\_ERR

### 10.38.12 OsSemPost

Low Level Design Details about CSU OsSemPost will follow in the sub sections

#### 10.38.12.1 Brief Description

The function OsSemPost posts to a Semaphore: Release the resource.

#### 10.38.12.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.38.12.3 List of global variables accessed and modified

Accessed : U32\_critical\_sr, Os\_tcb\_prio\_tbl

Modified : U32\_critical\_sr

#### 10.38.12.4 Parameter list (Input/Output)

Inputs : T\_OS\_EVENT \*p\_event Pointer to the desired semaphore's ECB

Outputs : T\_OS\_EVENT \*p\_event Pointer to the desired semaphore's ECB

#### 10.38.12.5 Return Value

T\_UBYTE - returns the error code

M\_OS\_NO\_ERR -a) Semaphore posted successfully.

b) Semaphore value within limit and no task waiting for Semaphore.

M\_OS\_SEM\_OVF -overflow of Semaphore value

#### 10.38.12.6 Other CSUs called by this CSU

OsSched,

SaveStatusReg,

RestoreStatusReg

#### 10.38.12.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to OsSemPost

##### 10.38.12.7.1 daugwyucos-OsSemPost-LLR-001

Requirement ID: H398-LLD-GWY-FNC-4744

The function shall call ‘SaveStatusReg’ (M\_OS\_ENTER\_CRITICAL) to disable interrupts and return value of ‘SaveStatusReg’ gets stored in U32\_critical\_sr

##### 10.38.12.7.2 daugwyucos-OsSemPost-LLR-002

Requirement ID: H398-LLD-GWY-FNC-4745

The function shall do the following when any of the task is waiting on Semaphore (i.e. os\_eventgrp of ECB p\_event is not equal to FALSE):

* Set second set of priority index to priority resolution table with index os\_eventgrp of p\_event.
* Set bit position for second set of priority index to mapping table to map bit position to bit mask with index second set of priority index.
* Set first set of priority index to priority resolution table with index os\_event\_tbl with index as second set of priority index of p\_event.
* Set bit position for first set of priority index to mapping table to map bit position to bit mask with index as first set of priority index.
* Get the index of highest priority task pending on event as ((second set of priority index bit shifted to left by M\_THREE) added to first set of priority index).
* Set os\_event\_tbl of p\_event with index second set of priority index to (os\_event\_tbl of p\_event with index second set of priority index bitwise AND with (Negation of (bit position for first set of priority index))).
* Set os\_eventgrp of p\_event to (os\_eventgrp of p\_event Bitwise AND with (Negation of (bit position for second set of priority index))) when os\_event\_tbl of p\_event is M\_ZERO, otherwise do nothing.
* Point the current TCB to Os\_tcb\_prio\_tbl with index as highest priority task index.
* Set os\_tcb\_dly of current task TCB to M\_ZERO.
* Set Event pointer os\_tcb\_eventptr of the current task TCB to M\_NULL.
* Set os\_tcb\_stat of the current task TCB to (os\_tcb\_stat of the current task TCB bitwise AND with (Negation of (M\_OS\_STAT\_SEM))).
* Set Ready list of group to (Ready list of group bitwise OR with bit position for second set of priority index) and table of task which are ready to run with index second set of priority index to (table of task which are ready to run with index second set of priority index bitwise OR with bit position for first set of priority index) when os\_tcb\_stat of current TCB is equal to M\_OS\_STAT\_RDY, otherwise do nothing.
* Send a request to uC/OS to enable all the interrupts by calling ‘RestoreStatusReg’ (M\_OS\_EXIT\_CRITICAL) with parameter U32\_critical\_sr.
* Call the function 'OsSched' to schedule the next highest priority task.
* Return with M\_OS\_NO\_ERR

##### 10.38.12.7.3 daugwyucos-OsSemPost-LLR-003

Requirement ID: H398-LLD-GWY-FNC-4746

The function shall do the following when no task is waiting for Semaphore and the Semaphore value is within the limit. (i.e. os\_eventgrp of ECB p\_event equal to FALSE AND os\_eventcnt of ECB p\_event is less than M\_MAX\_SEMPHORE):

a) Increment the Semaphore count to register event os\_eventcnt of ECB p\_event by one

b) Send a request to uC/OS to enable all the interrupts by calling ‘RestoreStatusReg’ (M\_OS\_EXIT\_CRITICAL) with parameter U32\_critical\_sr.

c) Return with M\_OS\_NO\_ERR

##### 10.38.12.7.4 daugwyucos-OsSemPost-LLR-004

Requirement ID: H398-LLD-GWY-FNC-4747

The function shall do the following when no task is waiting for Semaphore and the Semaphore value is over the limit. (i.e. os\_eventgrp of ECB p\_event equal to Zero AND os\_eventcnt of ECB p\_event is greater than M\_MAX\_SEMPHORE)

a) Send a request to uC/OS to enable all the interrupts by calling ‘RestoreStatusReg’ (M\_OS\_EXIT\_CRITICAL) with parameter U32\_critical\_sr.

b) Return with M\_OS\_SEM\_OVF

## 10.39 daugwywdog

The daugwywdog CSC contains implementation of routines for accessing the watch dog timer.

### 10.39.1 WdogInit

Low Level Design Details about CSU WdogInit will follow in the sub sections.

#### 10.39.1.1 Brief Description

The function calculates the reload value for the Independent Watchdog timer, initializes the prescaler divider to 64, writes reload value to the reload register, enables the Watchdog register and reloads Watchdog Counter.

#### 10.39.1.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.39.1.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 10.39.1.4 Parameter list (Input/Output)

Inputs: None

Outputs: None

#### 10.39.1.5 Return Value

None

#### 10.39.1.6 Other CSUs called by this CSU

IwdgWriteAccessCmd

IwdgSetPrescaler

IwdgSetReload

IwdgEnable

IwdgReloadCounter

#### 10.39.1.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to WdogInit.

##### 10.39.1.7.1 daugwywdog-WdogInit-LLR-001

Requirement ID: H398-LLD-GWY-FNC-6140

The function shall calculate the reload value for the Independent Watchdog timer as follows, the product of the Watchdog Timeout (M\_WDOG\_TIMEOUT\_MS) and Watchdog Frequency (M\_WDOG\_FREQ) divided by the (product of Watchdog prescaler (M\_GET\_PRESCALER\_4\_FOR\_WD left shift of M\_WDOG\_PRESCALER) and M\_WATCHDOG\_FREQ\_KHZ\_SCALE).

##### 10.39.1.7.2 daugwywdog-WdogInit-LLR-002

Requirement ID: H398-LLD-GWY-FNC-6141

The function shall set the calculated reload value to M\_MAX\_RELOAD\_VALUE for reload value greater than M\_MAX\_RELOAD\_VALUE else does nothing.

##### 10.39.1.7.3 daugwywdog-WdogInit-LLR-003

Requirement ID: H398-LLD-GWY-FNC-6142

The function shall call IwdgWriteAccessCmd with parameter M\_IWDG\_WRITEACCESS\_ENABLE to enable write access to the prescaler and reload registers before writing into the registers.

##### 10.39.1.7.4 daugwywdog-WdogInit-LLR-004

Requirement ID: H398-LLD-GWY-FNC-6143

The function shall call IwdgSetPrescaler with parameter M\_WDOG\_PRESCALER to sets IWDG Prescaler value.

##### 10.39.1.7.5 daugwywdog-WdogInit-LLR-005

Requirement ID: H398-LLD-GWY-FNC-6144

The function shall call IwdgSetReload with parameter reload value to set IWDG reload value and then the function shall call IwdgWriteAccessCmd with parameter M\_IWDG\_WRITEACCESS\_DISABLE to disable write access of prescaler and reload registers after write has been completed.

##### 10.39.1.7.6 daugwywdog-WdogInit-LLR-006

Requirement ID: H398-LLD-GWY-FNC-6145

The function shall call IwdgEnable to enable Watchdog Timer.

##### 10.39.1.7.7 daugwywdog-WdogInit-LLR-007

Requirement ID: H398-LLD-GWY-FNC-6146

The function shall call IwdgReloadCounter to reload Watchdog Counter.

### 10.39.2 WdogKickWatchDog

Low Level Design Details about CSU WdogKickWatchDog will follow in the sub sections.

#### 10.39.2.1 Brief Description

The WdogKickWatchDog function reloads Watchdog Counter.

#### 10.39.2.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.39.2.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.39.2.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.39.2.5 Return Value

None

#### 10.39.2.6 Other CSUs called by this CSU

IwdgReloadCounter

#### 10.39.2.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to WdogKickWatchDog.

##### 10.39.2.7.1 daugwywdog-WdogKickWatchDog-LLR-001

Requirement ID: H398-LLD-GWY-FNC-4757

The function shall call IwdgReloadCounter to reload the Watchdog Counter

## 10.40 daugwyxram

The daugwyxram CSC defines implementation of routines for XRAM for use by the DAU

### 10.40.1 XramInit

Low Level Design Details about CSU XramInit will follow in the sub sections

#### 10.40.1.1 Brief Description

The XramInit function configures the FSMC and enable the bank 1 of FSMC.

#### 10.40.1.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.40.1.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.40.1.4 Parameter list (Input/Output)

Inputs : None

Outputs : None

#### 10.40.1.5 Return Value

None

#### 10.40.1.6 Other CSUs called by this CSU

FsmcNorSramInit

FsmcNorSramCmd

#### 10.40.1.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to XramInit

##### 10.40.1.7.1 daugwyxram-XramInit-LLR-001

Requirement ID: H398-LLD-GWY-FNC-4767

The function shall set fields of FSMC timing register as follows

1. fsmc\_address\_setuptime is set with M\_ZERO
2. fsmc\_address\_holdtime is set with M\_ZERO
3. fsmc\_data\_setuptime is set with M\_FSMC\_DATA\_SETUPTIME\_DEFAULT
4. fsmc\_bus\_turnaround\_duration is set with M\_FSMC\_BUS\_TURNAROUND\_DEFAULT
5. fsmc\_clk\_division is set with M\_ZERO
6. fsmc\_data\_latency is set with M\_ZERO
7. fsmc\_access\_mode is set with M\_FSMC\_ACCESSMODE\_A

##### 10.40.1.7.2 daugwyxram-XramInit-LLR-002

Requirement ID: H398-LLD-GWY-FNC-4768

The function shall set fields of sram init structure as follows

1. fsmc\_readwrite\_timing\_struct is set with reference of FSMC timing register
2. fsmc\_write\_timing\_struct is set with reference of FSMC timing register
3. fsmc\_bank is set with M\_FSMC\_BANK1\_NORSRAM1
4. fsmc\_data\_address\_mux is set with M\_FSMC\_DATAADDRESSMUX\_DISABLE
5. fsmc\_memory\_type is set with M\_FSMC\_MEMORYTYPE\_PSRAM
6. fsmc\_memory\_datawidth is set with M\_FSMC\_MEMORYDATAWIDTH\_16B
7. fsmc\_burst\_accessmode is set with M\_FSMC\_BURSTACCESSMODE\_DISABLE
8. fsmc\_asynchronous\_wait is set with M\_FSMC\_ASYNCHRONOUSWAIT\_DISABLE
9. fsmc\_waitsignal\_polarity is set with M\_FSMC\_WAITSIGNALPOLARITY\_LOW
10. fsmc\_wrap\_mode is set with M\_FSMC\_WRAPMODE\_DISABLE
11. fsmc\_waitsignal\_active is set with M\_FSMC\_WAIT\_SIG\_ACTIVE\_BEF\_WAIT
12. fsmc\_write\_operation is set with M\_FSMC\_WRITEOPERATION\_ENABLE
13. fsmc\_waitsignal is set with M\_FSMC\_WAITSIGNAL\_DISABLE
14. fsmc\_extended\_mode is set with M\_FSMC\_EXTENDEDMODE\_DISABLE
15. fsmc\_write\_burst is set with M\_FSMC\_WRITEBURST\_DISABLE

##### 10.40.1.7.3 daugwyxram-XramInit-LLR-003

Requirement ID: H398-LLD-GWY-FNC-4769

The function shall initialize NORSRAM bank in FSMC by calling FsmcNorSramInit with parameter reference to sram init structure.

##### 10.40.1.7.4 daugwyxram-XramInit-LLR-004

Requirement ID: H398-LLD-GWY-FNC-4770

The function shall enable the second block of bank one of FSMC by calling FsmcNorSramCmd with parameters M\_FSMC\_BANK1\_NORSRAM1 and ENABLE.

## 10.41 daugwylogicairbus

The daugwylogicairbus CSC defines Logic Air Bus Functionality and its labels.

### 10.41.1 EngineStateLogic

Low Level Design Details about CSU EngineStateLogic will follow in the sub sections.

#### 10.41.1.1 Brief Description

The function calculates the logic for different engine states updates the engine state logic, and calls additional functions related to engine state and display class limits

#### 10.41.1.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.41.1.3 List of global variables accessed and modified

Accessed: Dau\_eng\_data

Engine\_State

Xram\_data\_buffer

Modified: Engine\_State

#### 10.41.1.4 Parameter list (Input/Output)

None

#### 10.41.1.5 Return Value

None

#### 10.41.1.6 Other CSUs called by this CSU

None

#### 10.41.1.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to daugwylogicairbus

##### 10.41.1.7.1 daugwylogicairbus-EngineStateLogic-LLR-001

Requirement ID: H398-LLD-GWY-FNC-6444

The function shall loop from M\_ZERO to M\_MAX\_ENGINE

1. Set engine logic with index [M\_ENGINE\_1 plus loop index] to STANDBY and operational state with index of (M\_ENGINE\_1 plus loop index) to TRUE when (((NG\_valid with index M\_ENGINE\_1 plus loop index is equal to OK AND (Dau\_eng\_data with index [TRQ1 plus loop index] is less than M\_SIX\_F) AND (HwAbsFlt with parameter(Dau\_eng\_data with index TACH\_NG1 minus Dau\_eng\_data with index TACH\_NG2) greater than M\_SEVEN\_FIVE\_F AND (disci with index DI\_37\_REGM2\_PROVISIONAL minus loop index) of App\_A825 of Logical NOT Xram data buffer) AND (ENG Training with index [M\_ENGINE\_2 minus loop index] of Engine State is set to FALSE) OR ( Dau\_eng\_data with the index [TACH\_NG1 plus loop index] is less than M\_SIXTY\_F) returns TRUE

##### 10.41.1.7.2 daugwylogicairbus-EngineStateLogic-LLR-002

Requirement ID: H398-LLD-GWY-FNC-6445

The function shall loop from M\_ZERO to M\_MAX\_ENGINE

1. Set engine logic with index [M\_ENGINE\_1 plus loop index] to START

2. Set ENG\_Start\_State with index [M\_ENGINE\_1 plus loop index] of Engine\_State to TRUE

3. set operational state with index of (M\_ENGINE\_1 + loop index) to TRUE.

when (ENG\_START\_ADV with index [M\_ENGINE\_1 plus loop index] of Engine\_State to TRUE) AND with (Dau\_eng\_data with index [TACH\_NG1 plus loop index] is less than M\_SEVENTY\_FIVE\_F) OR (NG\_valid with index M\_ENGINE\_1 plus loop index is not equal to OK) returns TRUE

otherwise ENG\_Start\_State with index [M\_ENGINE\_1 plus loop index] of Engine\_State is set to FALSE

##### 10.41.1.7.3 daugwylogicairbus-EngineStateLogic-LLR-003

Requirement ID: H398-LLD-GWY-FNC-6446

The function shall loop from M\_ZERO to M\_MAX\_ENGINE

1.Set engine logic with index[ M\_ENGINE\_1 plus loop index] to OEI\_TF

2. operational state with index of (M\_ENGINE\_1 + loop index) to TRUE.

When ENG\_Training with index[ M\_ENGINE\_1 plus loop index] of Engine\_State is set to TRUE

##### 10.41.1.7.4 daugwylogicairbus-EngineStateLogic-LLR-004

Requirement ID: H398-LLD-GWY-FNC-6447

The function shall loop from M\_ZERO to M\_MAX\_ENGINE

1.Set engine logic with index [M\_ENGINE\_1 plus loop index] to OEI\_TI

2. operational state with index of (M\_ENGINE\_1 + loop index) to TRUE.

when

ENG\_Training with index [M\_ENGINE\_2 minus loop index] of Engine\_State is set to TRUE

##### 10.41.1.7.5 daugwylogicairbus-EngineStateLogic-LLR-005

Requirement ID: H398-LLD-GWY-FNC-6448

The function shall loop from M\_ZERO to M\_MAX\_ENGINE

1.Set engine logic with index [M\_ENGINE\_1 plus loop index] to MAJOR\_FAILURE

2. operational state with index of (M\_ENGINE\_1 + loop index) to TRUE.

when

(disci with index [DI\_36\_REGM1\_PROVISIONAL plus loop index] of App\_A825 of

Xram\_data\_buffer) AND ( Dau\_eng\_data with index [TACH\_NG1 plus loop index]

is greater than M\_TEN\_F) OR (NG\_valid with index M\_ENGINE\_1 plus loop index is not equal to OK) returns TRUE.

##### 10.41.1.7.6 daugwylogicairbus-EngineStateLogic-LLR-006

Requirement ID: H398-LLD-GWY-FNC-6449

The function shall set

1.Set Engine State Logic with index M\_ENGINE\_1 to engine logic with index M\_ENGINE\_1

2.Set Engine State Logic with index M\_ENGINE\_2 to engine logic with index M\_ENGINE\_2

3.Call the function CompositeEngineStateLogic

4.Call the function EngineOEIStateLogic

5.Call the function TRQ12DisplayClass

6.Call the function DeltaNG1DisplayClass

7.Call the function DeltaNG2DisplayClass

8.Call the function T41DisplayClass

9.Call the function T42DisplayClass

##### 10.41.1.7.7 daugwylogicairbus-EngineStateLogic-LLR-007

Requirement ID: H398-LLD-GWY-FNC-7467

The function shall set engine logic with index of M\_ENGINE\_1 to OPERATIONAL when operational state with index ZERO is equal to FALSE. Otherwise do nothing.

##### 10.41.1.7.8 daugwylogicairbus-EngineStateLogic-LLR-008

Requirement ID: H398-LLD-GWY-FNC-7468

The function shall set engine logic with index of M\_ENGINE\_2 to OPERATIONAL when operational state with index ONE is equal to FALSE. Otherwise do nothing.

##### 10.41.1.7.9 daugwylogicairbus-EngineStateLogic-LLR-009

Requirement ID: H398-LLD-GWY-FNC-7488

The function shall loop from M\_ZERO to M\_MAX\_ENGINE

1.Set NG\_valid with index M\_ENGINE\_1 plus loop index is equal to OK when Dau\_eng\_data with index TACH\_NG1 plus loop index is greater than or equal to M\_FIVE\_F AND Dau\_eng\_data with index TACH\_NG1 plus loop index is less than or equal to M\_ONE\_TWENTY\_FIVE\_F OR Dau\_eng\_data with index TACH\_NG1 plus loop index is less than M\_FIVE\_F AND Dau\_eng\_status with index EOP plus loop index is not equal to OK OR Dau\_eng\_data with index EOP1 plus loop index is less than M\_EOP\_LOW\_RED\_LIMIT\_DIGITAL otherwise,

2.Set NG\_valid with index M\_ENGINE\_1 plus loop index is equal to CHAN\_ERR

### 10.41.2 CompositeEngineStateLogic

Low Level Design Details about CSU CompositeEngineStateLogic will follow in the sub sections.

#### 10.41.2.1 Brief Description

This function determines the composite state of two engines based on their individual states, considering various combinations of operational, standby, major failure, and start conditions.

#### 10.41.2.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.41.2.3 List of global variables accessed and modified

Accessed : Engine\_State\_Logic

Engine\_State

Composite\_engine\_state

Modified : Engine\_State

Composite\_engine\_state

#### 10.41.2.4 Parameter list (Input/Output)

Inputs: None

Outputs: None

#### 10.41.2.5 Return Value

None

#### 10.41.2.6 Other CSUs called by this CSU

None

#### 10.41.2.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to CompositeEngineStateLogic

##### 10.41.2.7.1 daugwylogicairbus-CompositeEngineStateLogic-LLR-001

Requirement ID: H398-LLD-GWY-FNC-6459

The function shall set Composite\_engine\_state to ONE when (Engine\_State\_Logic with index M\_ENGINE\_1 is equal to

START) AND (Engine\_State\_Logic with index M\_ENGINE\_2

is equal to START) returns TRUE

##### 10.41.2.7.2 daugwylogicairbus-CompositeEngineStateLogic-LLR-002

Requirement ID: H398-LLD-GWY-FNC-6460

The function shall set Composite\_engine\_state to TWO when (Engine\_State\_Logic with index M\_ENGINE\_1 is equal

to START) AND (Engine\_State\_Logic with index M\_ENGINE\_2 is equal to OPERATIONAL) returns TRUE

##### 10.41.2.7.3 daugwylogicairbus-CompositeEngineStateLogic-LLR-003

Requirement ID: H398-LLD-GWY-FNC-6461

The function shall set Composite\_engine\_state to THREE when (Engine\_State\_Logic with index M\_ENGINE\_1 is equal to START) AND (Engine\_State\_Logic with index M\_ENGINE\_2 is equal to MAJOR\_FAILURE) returns TRUE

##### 10.41.2.7.4 daugwylogicairbus-CompositeEngineStateLogic-LLR-004

Requirement ID: H398-LLD-GWY-FNC-6462

The function shall set Composite\_engine\_state to FOUR when (Engine\_State\_Logic with index M\_ENGINE\_1 is equal to START) AND (Engine\_State\_Logic with index M\_ENGINE\_2 is equal to STANDBY) returns TRUE

##### 10.41.2.7.5 daugwylogicairbus-CompositeEngineStateLogic-LLR-005

Requirement ID: H398-LLD-GWY-FNC-6463

The function shall perform the following:

1. Set Composite\_engine\_state to FIVE when (Engine\_State\_Logic with index M\_ENGINE\_1 is equal to OPERATIONAL) AND (Engine\_State\_Logic with index M\_ENGINE\_2 is equal to MAJOR\_FAILURE) returns TRUE
2. Set ENG\_AEO\_State with index M\_ENGINE\_1 of Engine\_State to FALSE
3. Set ENG\_AEO\_State with index M\_ENGINE\_2 of Engine\_State to FALSE

##### 10.41.2.7.6 daugwylogicairbus-CompositeEngineStateLogic-LLR-006

Requirement ID: H398-LLD-GWY-FNC-6464

The function shall perform the following when (Engine\_State\_Logic with index M\_ENGINE\_1 is equal to OPERATIONAL) AND (Engine\_State\_Logic with index M\_ENGINE\_2 is equal to OPERATIONAL) returns TRUE:

1. Set Composite\_engine\_state to Six
2. Set ENG\_AEO\_State with index M\_ENGINE\_1 of Engine\_State to TRUE
3. Set ENG\_AEO\_State with index M\_ENGINE\_2 of Engine\_State to TRUE

##### 10.41.2.7.7 daugwylogicairbus-CompositeEngineStateLogic-LLR-007

Requirement ID: H398-LLD-GWY-FNC-6465

The function shall set Composite\_engine\_state to Seven when (Engine\_State\_Logic with index M\_ENGINE\_1 is equal to OPERATIONAL) AND (Engine\_State\_Logic with index M\_ENGINE\_2 is equal to STANDBY) returns TRUE

##### 10.41.2.7.8 daugwylogicairbus-CompositeEngineStateLogic-LLR-008

Requirement ID: H398-LLD-GWY-FNC-6466

The function shall set Composite\_engine\_state to Eight when (Engine\_State\_Logic with index M\_ENGINE\_1 is equal to OEI\_TF) AND (Engine\_State\_Logic with index M\_ENGINE\_2 is equal to OEI\_TI) returns TRUE

##### 10.41.2.7.9 daugwylogicairbus-CompositeEngineStateLogic-LLR-009

Requirement ID: H398-LLD-GWY-FNC-6467

The function shall set Composite\_engine\_state to Nine when (Engine\_State\_Logic with index M\_ENGINE\_1 is equal to MAJOR\_FAILURE) AND (Engine\_State\_Logic with index M\_ENGINE\_2 is equal to MAJOR\_FAILURE) returns TRUE

##### 10.41.2.7.10 daugwylogicairbus-CompositeEngineStateLogic-LLR-010

Requirement ID: H398-LLD-GWY-FNC-6468

The function shall set Composite\_engine\_state to TEN when (Engine\_State\_Logic with index M\_ENGINE\_1 is equal to MAJOR\_FAILURE) AND (Engine\_State\_Logic with index M\_ENGINE\_2 is equal to STANDBY) returns TRUE

##### 10.41.2.7.11 daugwylogicairbus-CompositeEngineStateLogic-LLR-011

Requirement ID: H398-LLD-GWY-FNC-6469

The function shall set Composite\_engine\_state to ELEVEN when (Engine\_State\_Logic with index M\_ENGINE\_1 is equal to STANDBY) AND (Engine\_State\_Logic with index M\_ENGINE\_2 is equal to STANDBY) returns TRUE

##### 10.41.2.7.12 daugwylogicairbus-CompositeEngineStateLogic-LLR-012

Requirement ID: H398-LLD-GWY-FNC-6470

The function shall set Composite\_engine\_state to TWELVE when (Engine\_State\_Logic with index M\_ENGINE\_1 is

equal to OERATIONAL) AND (Engine\_State\_Logic with index M\_ENGINE\_2 is equal to START) returns TRUE

##### 10.41.2.7.13 daugwylogicairbus-CompositeEngineStateLogic-LLR-013

Requirement ID: H398-LLD-GWY-FNC-6471

The function shall set Composite\_engine\_state to THIRTEEN when (Engine\_State\_Logic with index M\_ENGINE\_1 is equal to MAJOR\_FAILURE) AND (Engine\_State\_Logic with index M\_ENGINE\_2 is equal to START) returns TRUE

##### 10.41.2.7.14 daugwylogicairbus-CompositeEngineStateLogic-LLR-014

Requirement ID: H398-LLD-GWY-FNC-6472

The function shall set Composite\_engine\_state to FOURTEEN when (Engine\_State\_Logic with index M\_ENGINE\_1 is equal to STANDBY) AND (Engine\_State\_Logic with index M\_ENGINE\_2 is equal to START) returns TRUE

##### 10.41.2.7.15 daugwylogicairbus-CompositeEngineStateLogic-LLR-015

Requirement ID: H398-LLD-GWY-FNC-6473

The function shall set Composite\_engine\_state to FIFTEEN when (Engine\_State\_Logic with index M\_ENGINE\_1 is equal to MAJOR\_FAILURE) AND (Engine\_State\_Logic with index M\_ENGINE\_2 is equal to OPERATIONAL) returns TRUE

##### 10.41.2.7.16 daugwylogicairbus-CompositeEngineStateLogic-LLR-016

Requirement ID: H398-LLD-GWY-FNC-6474

The function shall set Composite\_engine\_state to SIXTEEN when (Engine\_State\_Logic with index M\_ENGINE\_1 is equal to STANDBY) AND (Engine\_State\_Logic with index M\_ENGINE\_2 is equal to OPERATIONAL) returns TRUE

##### 10.41.2.7.17 daugwylogicairbus-CompositeEngineStateLogic-LLR-017

Requirement ID: H398-LLD-GWY-FNC-6475

The function shall set Composite\_engine\_state to SEVENTEEN when (Engine\_State\_Logic with index M\_ENGINE\_1 is equal OEI\_TI)AND (Engine\_State\_Logic with index M\_ENGINE\_2 is equal to OEI\_TF) returns TRUE

##### 10.41.2.7.18 daugwylogicairbus-CompositeEngineStateLogic-LLR-018

Requirement ID: H398-LLD-GWY-FNC-6476

The function shall set Composite\_engine\_state to EIGHTEEN when (Engine\_State\_Logic with index M\_ENGINE\_1 is equal to STANDBY) AND (Engine\_State\_Logic with index M\_ENGINE\_2 is equal to MAJOR\_FAILURE) returns TRUE

### 10.41.3 EngineOEIStateLogic

Low Level Design Details about CSU EngineOEIStateLogic will follow in the sub sections.

#### 10.41.3.1 Brief Description

This function sets the logic for Engine OEI states based on composite engine state logic.

#### 10.41.3.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.41.3.3 List of global variables accessed and modified

Accessed: None

Modified: Engine\_oei\_state

Composite\_engine\_state

#### 10.41.3.4 Parameter list (Input/Output)

None

#### 10.41.3.5 Return Value

None

#### 10.41.3.6 Other CSUs called by this CSU

None

#### 10.41.3.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to EngineOEIStateLogic

##### 10.41.3.7.1 daugwylogicairbus-EngineOEIStateLogic-LLR-001

Requirement ID: H398-LLD-GWY-FNC-6485

The function shall set

1. Set ENG\_OEI\_State with index M\_ENGINE\_1 of Engine\_oei\_state to FALSE

2. Set ENG\_OEI\_State with index M\_ENGINE\_2 of Engine\_oei\_state to FALSE

3. Set ENG\_OEI\_Training\_State with index M\_ENGINE\_1 of Engine\_oei\_state to FALSE

4. Set ENG\_OEI\_Training\_State with index M\_ENGINE\_2 of Engine\_oei\_state to FALSE

5. Set ENG\_OEI\_State with index [M\_ENGINE\_1] of Engine\_oei\_state to TRUE when Composite\_engine\_state is FIVE.

6. Set ENG\_OEI\_State with index [M\_ENGINE\_1] of Engine\_oei\_state to TRUE when Composite\_engine\_state is SEVEN.

7. Set ENG\_OEI\_State with index [M\_ENGINE\_1] of Engine\_oei\_state to TRUE when Composite\_engine\_state is TWELVE.

8. Set ENG\_OEI\_Training\_State with index [M\_ENGINE\_1] of Engine\_oei\_state to TRUE and  ENG\_OEI\_State with index [M\_ENGINE\_1] of Engine\_oei\_state to TRUE when Composite\_engine\_state is EIGHT.

9. Set ENG\_OEI\_State with index [M\_ENGINE\_2] of Engine\_oei\_state to TRUE when Composite\_engine\_state is TWO.

10. Set ENG\_OEI\_State with index [M\_ENGINE\_2] of Engine\_oei\_state to TRUE when Composite\_engine\_state is FIFTEEN.

11. Set ENG\_OEI\_State with index [M\_ENGINE\_2] of Engine\_oei\_state to TRUE when Composite\_engine\_state is SIXTEEN.

12. Set ENG\_OEI\_Training\_State with index [M\_ENGINE\_2] of Engine\_oei\_state to TRUE and  ENG\_OEI\_State with index [M\_ENGINE\_2] of Engine\_oei\_state to TRUE when Composite\_engine\_state is SEVENTEEN.

13. Do nothing when Composite\_engine\_state is ONE, THREE, FOUR, SIX, NINE, TEN, ELEVEN , THIRTEEN, FOURTEEN and EIGHTEEN.

### 10.41.4 DiscreteSignals

Low Level Design Details about CSU DiscreteSignals will follow in the sub sections

#### 10.41.4.1 Brief Description

This function calls some discrete signals related to

- GroundState

- ExternalInterfaces

- BleedValue

- EngineState

- EngineChip

- MGBOil

- MGBChip

- IGBOilTemp

- TBGOilTemp

- HydraulicCautions

- DeltaNGThreshDiscreteOutput

#### 10.41.4.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.41.4.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 10.41.4.4 Parameter list (Input/Output)

None

#### 10.41.4.5 Return Value

None

#### 10.41.4.6 Other CSUs called by this CSU

None

#### 10.41.4.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to DiscreteSignals

##### 10.41.4.7.1 daugwylogicairbus-DiscreteSignals-LLR-001

Requirement ID: H398-LLD-GWY-FNC-6495

The function shall do the following

1. Call the function GroundState
2. Call the function ExternalInterfaces
3. Call the function BleedValue
4. Call the function EngineState
5. Call the function EngineChip
6. Call the function MGBOil
7. Call the function MGBChip
8. Call the function IGBOilTemp
9. Call the function TBGOilTemp
10. Call the function HydraulicCautions
11. Call DeltaNGThreshDiscreteOutput

### 10.41.5 GroundState

Low Level Design Details about CSU GroundState will follow in the sub sections

#### 10.41.5.1 Brief Description

This function sets the GroundState

#### 10.41.5.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.41.5.3 List of global variables accessed and modified

Accessed: Xram\_data\_buffer

Modified: Xram\_data\_buffer

#### 10.41.5.4 Parameter list (Input/Output)

None

#### 10.41.5.5 Return Value

None

#### 10.41.5.6 Other CSUs called by this CSU

None

#### 10.41.5.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to GroundState

##### 10.41.5.7.1 daugwylogicairbus-GroundState-LLR-001

Requirement ID: H398-LLD-GWY-FNC-6504

The function shall do the following

- Set on\_airborne of Xram\_data\_buffer to FALSE and Onground to TRUE when disci with index DI\_1\_ON\_GROUND of App\_A825 of Xram\_data\_buffer is not equal M\_ZERO ,

otherwise Set on\_airborne of Xram\_data\_buffer to TRUE and Onground to FALSE.

### 10.41.6 ExternalInterfaces

Low Level Design Details about CSU ExternalInterfaces will follow in the sub sections.

#### 10.41.6.1 Brief Description

This function sets the External interfaces of engine 1 and engine 2

#### 10.41.6.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.41.6.3 List of global variables accessed and modified

Accessed:

Xram\_data\_buffer.

Modified:

EXT\_interfaces

#### 10.41.6.4 Parameter list (Input/Output)

None

#### 10.41.6.5 Return Value

None

#### 10.41.6.6 Other CSUs called by this CSU

None

#### 10.41.6.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to ExternalInterfaces

##### 10.41.6.7.1 daugwylogicairbus-ExternalInterfaces-LLR-001

Requirement ID: H398-LLD-GWY-FNC-6513

The function shall set ENG\_HI\_MSG with index M\_ENGINE\_1 of EXT\_interfaces to TRUE

-when (disci of index DI\_2\_OEI30\_ENG1 of App\_A825 of Xram\_data\_buffer) OR with

(disci of index DI\_2\_OEI30\_ENG1 of App\_A825 of Xram\_data\_buffer)AND with(disci of index DI\_4\_OEI2\_ENG1 of App\_A825 of Xram\_data\_buffer)) returns TRUE

otherwise set ENG\_HI\_MSG with index M\_ENGINE\_1 of EXT\_interfaces to FALSE

##### 10.41.6.7.2 daugwylogicairbus-ExternalInterfaces-LLR-002

Requirement ID: H398-LLD-GWY-FNC-6514

The function shall set ENG\_HI\_MSG with index M\_ENGINE\_2 of EXT\_interfaces to TRUE

when

(disci of index DI\_3\_OEI30\_ENG2 of App\_A825 of Xram\_data\_buffer ) OR with

(disci of index DI\_3\_OEI30\_ENG2 of App\_A825 of Xram\_data\_buffer)AND with(disci of index DI\_5\_OEI2\_ENG2 of App\_A825 of Xram\_data\_buffer)) returns TRUE,

otherwise set ENG\_HI\_MSG with index M\_ENGINE\_2 of EXT\_interfaces to FALSE

##### 10.41.6.7.3 daugwylogicairbus-ExternalInterfaces-LLR-003

Requirement ID: H398-LLD-GWY-FNC-6515

The function shall set ENG\_LO\_MSG with index M\_ENGINE\_1 of EXT\_interfaces to TRUE

when

(disci of index DI\_4\_OEI2\_ENG1 of App\_A825 of Xram\_data\_buffer )AND with

NOT of(disci of index DI\_2\_OEI30\_ENG1 of App\_A825 of Xram\_data\_buffer )) returns TRUE

otherwise set ENG\_LO\_MSG with index M\_ENGINE\_1 of EXT\_interfaces to FALSE

##### 10.41.6.7.4 daugwylogicairbus-ExternalInterfaces-LLR-004

Requirement ID: H398-LLD-GWY-FNC-6516

The function shall set ENG\_LO\_MSG with index M\_ENGINE\_2 of EXT\_interfaces to TRUE

when

(disci of index DI\_5\_OEI2\_ENG2 of App\_A825 of Xram\_data\_buffer ) AND with

NOT of( (disci of index DI\_3\_OEI30\_ENG2 of App\_A825 of Xram\_data\_buffer ) returns TRUE

otherwise set ENG\_LO\_MSG with index M\_ENGINE\_2 of EXT\_interfaces to FALSE

##### 10.41.6.7.5 daugwylogicairbus-ExternalInterfaces-LLR-005

Requirement ID: H398-LLD-GWY-FNC-6517

The function shall increment gov1\_activetimer and set gov1\_inactivetimer to M\_ZERO

when disci with index DI\_11\_ENG\_GOV1 of App\_A825 of Xram\_data\_buffer is TRUE,

otherwise increment gov1\_inactivetimer and set gov1\_activetimer to M\_ZERO.

##### 10.41.6.7.6 daugwylogicairbus-ExternalInterfaces-LLR-009

Requirement ID: H398-LLD-GWY-FNC-7478

The function shall performs the following:

* Set ENG\_GOV\_Caution with index M\_ENGINE\_1 of Ext\_interfaces to TRUE

Set gov1\_activetimer to M\_300\_MILLISECOND\_GOV

Set gov1\_setblink to M\_ZERO

when gov1\_activetimer is greater than M\_300\_MILLISECOND\_GOV

* Set ENG\_GOV\_Caution with index M\_ENGINE\_1 of Ext\_interfaces to FALSE

Set gov1\_inactivetimer to M\_300\_MILLISECOND\_GOV

Set gov1\_setblink to M\_ZERO

when gov1\_activetimer is less than M\_300\_MILLISECOND\_GOV and gov1\_inactivetimer is greater than M\_300\_MILLISECOND\_GOV

* Otherwise,

the function shall increment gov1\_setblink.

##### 10.41.6.7.7 daugwylogicairbus-ExternalInterfaces-LLR-010

Requirement ID: H398-LLD-GWY-FNC-7479

The function shall performs the following when gov1\_setblink is greater than M\_300\_MILLISECOND\_GOV

* Set gov1\_setblink to M\_300\_MILLISECOND\_GOV
* Set ENG GOV Caution with index M\_ENGINE\_1 of Ext interfaces to FALSE, increment gov1\_offblinktimer and set gov1\_onblinktimer to M\_ZERO when gov1\_offblinktimer is less than or equal to M\_400\_MILLISECOND\_GOV.
* Set ENG\_GOV\_Caution with index M\_ENGINE\_1 of Ext\_interfaces to TRUE and increment gov1\_onblinktimer when gov1\_onblinktimer is less than or equal to M\_800\_MILLISECOND\_GOV and gov1\_offblinktimer greater than M\_400\_MILLISECOND\_GOV.
* Set gov1\_offblinktimer to M\_ZERO when gov1\_onblinktimer is greater than M\_800\_MILLISECOND\_GOV and gov1\_offblinktimer greater than M\_400\_MILLISECOND\_GOV.

##### 10.41.6.7.8 daugwylogicairbus-ExternalInterfaces-LLR-006

Requirement ID: H398-LLD-GWY-FNC-6518

The function shall increment gov2\_activetimer and set gov2\_inactivetimer to M\_ZERO when disci with index DI\_12\_ENG\_GOV2 of App\_A825 of Xram\_data\_buffer is equal toTRUE,

otherwise increment gov2\_inactivetimer and set gov2\_activetimer to M\_ZERO.

##### 10.41.6.7.9 daugwylogicairbus-ExternalInterfaces-LLR-011

Requirement ID: H398-LLD-GWY-FNC-7482

The function shall performs the following:

* Set ENG\_GOV\_Caution with index M\_ENGINE\_2 of Ext\_interfaces to TRUE

Set gov2\_activetimer to M\_300\_MILLISECOND\_GOV

Set gov2\_setblink to M\_ZERO

when gov2\_activetimer is greater than M\_300\_MILLISECOND\_GOV

* Set ENG\_GOV\_Caution with index M\_ENGINE\_2 of Ext\_interfaces to FALSE

Set gov2\_inactivetimer to M\_300\_MILLISECOND\_GOV

Set gov2\_setblink to M\_ZERO

when gov2\_activetimer is less than M\_300\_MILLISECOND\_GOV and gov2\_inactivetimer is greater than M\_300\_MILLISECOND\_GOV

* Otherwise,

the function shall increment gov2\_setblink.

##### 10.41.6.7.10 daugwylogicairbus-ExternalInterfaces-LLR-012

Requirement ID: H398-LLD-GWY-FNC-7483

The function shall performs the following when gov2\_setblink is greater than M\_300\_MILLISECOND\_GOV

* Set gov2\_setblink to M\_300\_MILLISECOND\_GOV
* Set ENG\_GOV\_Caution with index M\_ENGINE\_2 of Ext\_interfaces to FALSE, increment gov2\_offblinktimer and set gov2\_onblinktimer to M\_ZERO when gov2\_offblinktimer is less than or equal to M\_400\_MILLISECOND\_GOV.
* Set ENG\_GOV\_Caution with index M\_ENGINE\_2 of Ext\_interfaces to TRUE and increment gov2\_onblinktimer when gov2\_onblinktimer is less than or equal to M\_800\_MILLISECOND\_GOV and gov2\_offblinktimer greater than M\_400\_MILLISECOND\_GOV.
* Set gov2\_offblinktimer to M\_ZERO when gov2\_onblinktimer is greater than M\_800\_MILLISECOND\_GOV and gov2\_offblinktimer greater than M\_400\_MILLISECOND\_GOV.

##### 10.41.6.7.11 daugwylogicairbus-ExternalInterfaces-LLR-007

Requirement ID: H398-LLD-GWY-FNC-6519

The function shall set ENG\_PWR\_C\_Caution of EXT\_interfaces to TRUE

when disci with index DI\_32\_PWR\_C of App\_A825 of Xram\_data\_buffer is not equal to M\_ZERO,

otherwise set ENG\_PWR\_C\_Caution of EXT\_interfaces to FALSE

##### 10.41.6.7.12 daugwylogicairbus-ExternalInterfaces-LLR-008

Requirement ID: H398-LLD-GWY-FNC-6520

The function shall set ENG\_THRT\_Warning of EXT\_interfaces to TRUE

when disci with index DI\_33\_THRT of App\_A825 of Xram\_data\_buffer is not equal to M\_ZERO,

otherwise set ENG\_THRT\_Warning of EXT\_interfaces to FALSE

##### 10.41.6.7.13 daugwylogicairbus-ExternalInterfaces-LLR-013

Requirement ID: H398-LLD-GWY-FNC-7487

The function shall set ENG\_GOV\_Caution with index M\_ENGINE\_1 of Ext\_interfaces to TRUE and increment gov1\_onblinktimer otherwise set gov1\_offblinktimer to M\_ZERO

### 10.41.7 BleedValue

Low Level Design Details about CSU BleedValue will follow in the sub sections.

#### 10.41.7.1 Brief Description

This function sets the Bleed value for engine 1 and engine 2

#### 10.41.7.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.41.7.3 List of global variables accessed and modified

Accessed:

Xram\_data\_buffer

Modified:

Bleed\_Value

#### 10.41.7.4 Parameter list (Input/Output)

None

#### 10.41.7.5 Return Value

None

#### 10.41.7.6 Other CSUs called by this CSU

None

#### 10.41.7.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to BleedValue.

##### 10.41.7.7.1 daugwylogicairbus-BleedValue-LLR-001

Requirement ID: H398-LLD-GWY-FNC-6529

The function shall set ENG\_1\_Bleed\_Val\_Open of Bleed\_Value to TRUE

when disci with index DI\_6\_BVALUE1\_IN of App\_A825 of Xram\_data\_buffer is not equal to M\_ZERO,

otherwise set ENG\_1\_Bleed\_Val\_Open of Bleed\_Value to FALSE

##### 10.41.7.7.2 daugwylogicairbus-BleedValue-LLR-002

Requirement ID: H398-LLD-GWY-FNC-6530

The function shall set ENG\_2\_Bleed\_Val\_Open of Bleed\_Value to TRUE

when disci with index DI\_7\_BVALUE2\_IN of App\_A825 of Xram\_data\_buffer not equal to M\_ZERO,

otherwise set ENG\_2\_Bleed\_Val\_Open of Bleed\_Value to FALSE

##### 10.41.7.7.3 daugwylogicairbus-BleedValue-LLR-003

Requirement ID: H398-LLD-GWY-FNC-6531

The function shall set ENG\_1\_Bleed\_Val\_Offset of Bleed\_Value to TRUE

when disci with index DI\_17\_BLEED\_OFFSET1 of App\_A825 of Xram\_data\_buffer not equal to M\_ZERO,

otherwise set ENG\_1\_Bleed\_Val\_Offset of Bleed\_Value to FALSE

##### 10.41.7.7.4 daugwylogicairbus-BleedValue-LLR-004

Requirement ID: H398-LLD-GWY-FNC-6532

The function shall set ENG\_2\_Bleed\_Val\_Offset of Bleed\_Value to TRUE

when disci with index DI\_18\_BLEED\_OFFSET2 of App\_A825 of Xram\_data\_buffer not equal to M\_ZERO,

otherwise set ENG\_2\_Bleed\_Val\_Offset of Bleed\_Value to FALSE

### 10.41.8 EngineState

Low Level Design Details about CSU EngineState will follow in the sub sections.

#### 10.41.8.1 Brief Description

This function sets the Engine state for engine 1 and engine 2

#### 10.41.8.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.41.8.3 List of global variables accessed and modified

Accessed: Engine\_State\_Logic

Engine\_oei\_state

Modified: Engine\_State

#### 10.41.8.4 Parameter list (Input/Output)

None

#### 10.41.8.5 Return Value

None

#### 10.41.8.6 Other CSUs called by this CSU

None

#### 10.41.8.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to EngineState.

##### 10.41.8.7.1 daugwylogicairbus-EngineState-LLR-001

Requirement ID: H398-LLD-GWY-FNC-6541

The function shall set ENG\_START\_ADV with index M\_ENGINE\_1 of Engine\_State to TRUE

when disci with index DI\_9\_ENG\_START1 of App\_A825 of Xram\_data\_buffer is TRUE AND Dau\_eng\_data with index TACH\_NG1 is less than M\_SEVENTY\_FIVE.

otherwise set ENG\_START\_ADV with index M\_ENGINE\_1 of Engine\_State to  to FALSE.

##### 10.41.8.7.2 daugwylogicairbus-EngineState-LLR-002

Requirement ID: H398-LLD-GWY-FNC-6542

The function shall set ENG\_START\_ADV with index M\_ENGINE\_2 of Engine\_State to TRUE

when disci with index DI\_10\_ENG\_START2 of App\_A825 of Xram\_data\_buffer is TRUE AND Dau\_eng\_data with index TACH\_NG2 is less than M\_SEVENTY\_FIVE.

otherwise set ENG\_START\_ADV with index M\_ENGINE\_2 of Engine\_State to FALSE.

##### 10.41.8.7.3 daugwylogicairbus-EngineState-LLR-003

Requirement ID: H398-LLD-GWY-FNC-6543

The function shall set ENG\_Training with index M\_ENGINE\_1 of Engine\_State to TRUE

when disci with index DI\_15\_T1 of App\_A825 of Xram\_data\_buffer is not equal to M\_ZERO ,

otherwise set ENG\_Training with index M\_ENGINE\_1 of Engine\_State to FALSE.

##### 10.41.8.7.4 daugwylogicairbus-EngineState-LLR-004

Requirement ID: H398-LLD-GWY-FNC-6544

The function shall set ENG\_Training with index M\_ENGINE\_2 of Engine\_State to TRUEwhen

disci with index DI\_16\_T2 of App\_A825 of Xram\_data\_buffer is not equal to M\_ZERO,

otherwise set ENG\_Training with index M\_ENGINE\_2 of Engine\_State to FALSE.

##### 10.41.8.7.5 daugwylogicairbus-EngineState-LLR-005

Requirement ID: H398-LLD-GWY-FNC-6545

The function shall do the following

* Set ENG\_Failure\_Frame with index M\_ENGINE\_1 of Engine\_State to TRUE when

   (((  ENG\_OEI\_State with index M\_ENGINE\_2  of Engine\_oei\_state is set to TRUE)  AND (ENG\_Training  with

index M\_ENGINE\_1 of  Engine\_State is FALSE)) OR

(ENG\_Training with index M\_ENGINE\_2 of Engine\_State is TRUE)  AND (ENG\_START\_ADV with  index

M\_ENGINE\_1 of Engine\_State is FALSE) AND

(arinc\_Valid to TRUE)) returns TRUE

Otherwise set ENG\_Failure\_Frame with index M\_ENGINE\_1 of Engine\_State to FALSE

##### 10.41.8.7.6 daugwylogicairbus-EngineState-LLR-006

Requirement ID: H398-LLD-GWY-FNC-6546

The function shall do the following

* Set ENG\_Failure\_Frame with index M\_ENGINE\_2 of Engine\_State to TRUE when

((((((  ENG\_OEI\_State with index M\_ENGINE\_1  of Engine\_oei\_state is set to TRUE)  AND  (ENG\_Training  with

index M\_ENGINE\_2 of  Engine\_State to FALSE)) OR

(ENG\_Training with index M\_ENGINE\_1 of Engine\_State is TRUE)  AND (ENG\_START\_ADV with  index

M\_ENGINE\_2 of Engine\_State is FALSE) AND

(arinc\_valid is TRUE)) returns TRUE

Otherwise set ENG\_Failure\_Frame with index M\_ENGINE\_2 of Engine\_State to FALSE

### 10.41.9 EngineChip

Low Level Design Details about CSU EngineChip will follow in the sub sections.

#### 10.41.9.1 Brief Description

This function sets the Engine 1 and Engine 2 chip detectors

#### 10.41.9.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.41.9.3 List of global variables accessed and modified

Accessed:

Xram\_data\_buffer

Modified:

Engine\_Chip

#### 10.41.9.4 Parameter list (Input/Output)

None

#### 10.41.9.5 Return Value

None

#### 10.41.9.6 Other CSUs called by this CSU

None

#### 10.41.9.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to EngineChip

##### 10.41.9.7.1 daugwylogicairbus-EngineChip-LLR-001

Requirement ID: H398-LLD-GWY-FNC-6555

The function shall set ENG\_1\_Chip\_Detector of Engine\_Chip to TRUE

when disci with index DI\_13\_ENG\_CHIP1 of App\_A825 of Xram\_data\_buffer is not equal to M\_ZERO,

otherwise set ENG\_1\_Chip\_Detector of Engine\_Chip to FALSE.

##### 10.41.9.7.2 daugwylogicairbus- EngineChip-LLR-002

Requirement ID: H398-LLD-GWY-FNC-6556

The function shall set ENG\_2\_Chip\_Detector of Engine\_Chip to TRUE

when disci with index DI\_14\_ENG\_CHIP2 of App\_A825 of Xram\_data\_buffer is not equal to M\_ZERO,

otherwise set ENG\_2\_Chip\_Detector of Engine\_Chip to FALSE.

### 10.41.10 MGBOil

Low Level Design Details about CSU MGBOil will follow in the sub sections.

#### 10.41.10.1 Brief Description

This function sets the alarm for MGBOil parameters

#### 10.41.10.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.41.10.3 List of global variables accessed and modified

Accessed: Xram\_data\_buffer

Mgb\_oil

Modified: Mgb\_oil

#### 10.41.10.4 Parameter list (Input/Output)

None

#### 10.41.10.5 Return Value

None

#### 10.41.10.6 Other CSUs called by this CSU

None

#### 10.41.10.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to MGBOil.

##### 10.41.10.7.1 daugwylogicairbus-MGBOil-LLR-001

Requirement ID: H398-LLD-GWY-FNC-6565

The function shall set MGB\_Main\_pump\_low\_pressure\_alarm of MGB\_Oil to TRUE

when disci with index DI\_19\_XMSN\_M\_PRESS of App\_A825 of Xram\_data\_buffer is not equal to M\_ZERO,

otherwise set MGB\_Main\_pump\_low\_pressure\_alarm of MGB\_Oil to FALSE.

##### 10.41.10.7.2 daugwylogicairbus-MGBOil-LLR-002

Requirement ID: H398-LLD-GWY-FNC-6566

The function shall set MGB\_Stby\_pump\_low\_pressure\_alarm of MGB\_Oil to TRUE

when disci with index DI\_20\_XMSN\_SB\_PRESS of App\_A825 of Xram\_data\_buffer is not equal to M\_ZERO,

otherwise set MGB\_Stby\_pump\_low\_pressure\_alarm of MGB\_Oil to FALSE.

##### 10.41.10.7.3 daugwylogicairbus-MGBOil-LLR-003

Requirement ID: H398-LLD-GWY-FNC-6567

The function shall set MGB\_oil\_temp\_alarm of MGB\_Oil to TRUE

when disci with index DI\_21\_XMSN\_MGBT of App\_A825 of Xram\_data\_buffer is not equal to M\_ZERO,

otherwise set MGB\_oil\_temp\_alarm of MGB\_Oil to FALSE.

### 10.41.11 MGBChip

Low Level Design Details about CSU MGB Chip will follow in the sub sections.

#### 10.41.11.1 Brief Description

This function sets the MGB Chip alaram

#### 10.41.11.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.41.11.3 List of global variables accessed and modified

Accessed: Xram\_data\_buffer

Modified: Mgb\_chip

#### 10.41.11.4 Parameter list (Input/Output)

None

#### 10.41.11.5 Return Value

None

#### 10.41.11.6 Other CSUs called by this CSU

None

#### 10.41.11.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to MGBChip.

##### 10.41.11.7.1 daugwylogicairbus-MGB-Chip -LLR-001

Requirement ID: H398-LLD-GWY-FNC-6576

The function shall set MGB Chip to TRUE when disci with index DI\_22\_XMSN\_CHIP of App\_A825 of Xram\_data\_buffer is not equal to M\_ZERO,

otherwise set MGB Chip to FALSE.

### 10.41.12 IGBOilTemp

Low Level Design Details about CSU IGBOilTemp will follow in the sub sections.

#### 10.41.12.1 Brief Description

This function sets the IGB oil temperature alarm

#### 10.41.12.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.41.12.3 List of global variables accessed and modified

Accessed: Xram\_data\_buffer

Modified: Igb\_oil\_temp

#### 10.41.12.4 Parameter list (Input/Output)

None

#### 10.41.12.5 Return Value

None

#### 10.41.12.6 Other CSUs called by this CSU

None

#### 10.41.12.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to IGBOilTemp.

##### 10.41.12.7.1 daugwylogicairbus-IGBOilTemp-LLR-001

Requirement ID: H398-LLD-GWY-FNC-6585

The function shall set IGB\_Oil\_Temp to TRUE when

disci with index DI\_23\_XMSN\_IGBT of App\_A825 of Xram\_data\_buffer is not equal to M\_ZERO, otherwise set IGB\_Oil\_Temp to FALSE.

### 10.41.13 TBGOilTemp

Low Level Design Details about CSU TBGOilTemp will follow in the sub sections.

#### 10.41.13.1 Brief Description

This function sets the TBG Oil Temperature alarm

#### 10.41.13.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.41.13.3 List of global variables accessed and modified

Accessed: Xram\_data\_buffer

Modified: Tgb\_oil\_temp

#### 10.41.13.4 Parameter list (Input/Output)

None

#### 10.41.13.5 Return Value

None

#### 10.41.13.6 Other CSUs called by this CSU

None

#### 10.41.13.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TBGOilTemp.

##### 10.41.13.7.1 daugwylogicairbus- TBGOilTemp -LLR-001

Requirement ID: H398-LLD-GWY-FNC-6594

The function shall set TGB\_Oil\_Temp to TRUE

when disci with index DI\_24\_XMSN\_TGBT of App\_A825 of Xram\_data\_buffer is not equal to M\_ZERO,

otherwise set TGB\_Oil\_Temp to FALSE.

### 10.41.14 HydraulicCautions

Low Level Design Details about CSU HydraulicCautions will follow in the sub sections.

#### 10.41.14.1 Brief Description

This function sets the alerts for Hydraulic Cautions

#### 10.41.14.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.41.14.3 List of global variables accessed and modified

Accessed:

Xram\_data\_buffer

Modified:

Hyd\_Cautions

#### 10.41.14.4 Parameter list (Input/Output)

None

#### 10.41.14.5 Return Value

None

#### 10.41.14.6 Other CSUs called by this CSU

None

#### 10.41.14.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to HydraulicCautions.

##### 10.41.14.7.1 daugwylogicairbus-HydraulicCautions-LLR-001

Requirement ID: H398-LLD-GWY-FNC-6603

The function shall set HYD\_aux\_pump\_low\_pressure\_alert of Hyd\_Cautions to TRUE when

disci with index DI\_25\_HYD\_AUX\_PUMP\_PRESS of App\_A825 of Xram\_data\_buffer is not equal to M\_ZERO,

otherwise set HYD\_aux\_pump\_low\_pressure\_alert of Hyd\_cautions to FALSE.

##### 10.41.14.7.2 daugwylogicairbus-HydraulicCautions-LLR-002

Requirement ID: H398-LLD-GWY-FNC-6604

The function shall set HYD\_auto\_pilot\_circuit\_pressure\_alert of Hyd\_Cautions to TRUE

when disci with index DI\_26\_HYD\_AP\_PRESS of App\_A825 of Xram\_data\_buffer is not equal to M\_ZERO,

otherwise set HYD\_auto\_pilot\_circuit\_pressure\_alert of Hyd\_Cautions to FALSE.

##### 10.41.14.7.3 daugwylogicairbus-HydraulicCautions-LLR-003

Requirement ID: H398-LLD-GWY-FNC-6605

The function shall set HYD\_1\_main\_pump\_low\_pressure\_alert of Hyd\_Cautions to TRUE

when disci with index DI\_27\_HYD\_MAIN\_PRESS of App\_A825 of Xram\_data\_buffer is not equal to M\_ZERO,

otherwise set HYD\_1\_main\_pump\_low\_pressure\_alert of Hyd\_Cautions to FALSE.

##### 10.41.14.7.4 daugwylogicairbus-HydraulicCautions-LLR-004

Requirement ID: H398-LLD-GWY-FNC-6606

The function shall set HYD\_1\_low\_level\_alert of Hyd\_Cautions to TRUE

when disci with index DI\_28\_HYD\_MAIN\_LEVEL of App\_A825 of Xram\_data\_buffer is not equal to M\_ZERO,

otherwise set HYD\_1\_low\_level\_alert of Hyd\_Cautions to FALSE.

##### 10.41.14.7.5 daugwylogicairbus-HydraulicCautions-LLR-005

Requirement ID: H398-LLD-GWY-FNC-6607

The function shall set HYD\_2\_pump\_low\_pressure\_alert of Hyd\_Cautions to TRUE when

disci with index DI\_29\_HYD\_RH\_PRESS of App\_A825 of Xram\_data\_buffer is not equal to M\_ZERO,

otherwise set HYD\_2\_pump\_low\_pressure\_alert of Hyd\_Cautions to FALSE.

##### 10.41.14.7.6 daugwylogicairbus-HydraulicCautions-LLR-006

Requirement ID: H398-LLD-GWY-FNC-6608

The function shall set HYD\_2\_low\_level\_alert of Hyd\_Cautions to TRUE

when disci with index DI\_30\_HYD\_RH\_LOW\_LEVEL of App\_A825 of Xram\_data\_buffer is not equal to M\_ZERO,

otherwise set HYD\_2\_low\_level\_alert of Hyd\_Cautions to FALSE.

##### 10.41.14.7.7 daugwylogicairbus-HydraulicCautions-LLR-007

Requirement ID: H398-LLD-GWY-FNC-6609

The function shall set HYD\_SERVO\_alert of Hyd\_Cautions to TRUE

when disci with index DI\_34\_SERVO of App\_A825 of Xram\_data\_buffer is not equal to M\_ZERO,

otherwise set HYD\_SERVO\_alert of Hyd\_Cautions to FALSE.

##### 10.41.14.7.8 daugwylogicairbus-HydraulicCautions-LLR-008

Requirement ID: H398-LLD-GWY-FNC-6610

The function shall set HYD\_1\_aux\_pump\_fail\_alert of Hyd\_Cautions to TRUE

when disci with index DI\_35\_AUX\_PUMP\_FAIL of App\_A825 of Xram\_data\_buffer is not equal to M\_ZERO,

otherwise set HYD\_1\_aux\_pump\_fail\_alert of Hyd\_Cautions to FALSE.

### 10.41.15 TxProcessAirbusLabel

Low Level Design Details about CSU TxProcessAirbusLabel will follow in the sub sections.

#### 10.41.15.1 Brief Description

This function processes all the Airbus labels and prepares the data packets for transmission.

#### 10.41.15.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.41.15.3 List of global variables accessed and modified

None

#### 10.41.15.4 Parameter list (Input/Output)

None

#### 10.41.15.5 Return Value

None

#### 10.41.15.6 Other CSUs called by this CSU

None

#### 10.41.15.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TxProcessAirbusLabel

##### 10.41.15.7.1 daugwylogicairbus-TxProcessAirbusLabel-LLR-001

Requirement ID: H398-LLD-GWY-FNC-6619

The function shall do the following

1. Call ArincLineStatus
2. Call DeltaNG1OEITopping
3. Call DeltaNG2OEITopping
4. Call TxProcessLabel050TRQ
5. Call TxProcessLabel051TRQAngle
6. Call TxProcessLabel052TRQAEOYellow
7. Call TxProcessLabel053TRQAEOYelRec
8. Call TxProcessLabel054TRQAEORed
9. Call TxProcessLabel055TRQAEORdDi
10. Call TxProcessLabel057TRQOEIYelDash
11. Call TxProcessLabel060TRQOEIRedDash
12. Call TxProcessLabel061TRQOEIRedLine
13. Call TxProcessLabel174HYD
14. Call TxProcessLabel175HYDAng
15. Call TxProcessLabel176MGBOilPress
16. Call TxProcessLabel177MGBOilPrsAng
17. Call TxProcessLabel200MGBOilTemp
18. Call TxProcessLabel201MGBOilTempAng
19. Call TxProcessLabel262FuelPress
20. Call TxProcessLabel265T4
21. Call TxProcessLabel266T4Angle
22. Call TxProcessLabel270T4AEOYellow
23. Call TxProcessLabel271T4AEOYelRec
24. Call TxProcessLabel272T4AEORed

##### 10.41.15.7.2 daugwylogicairbus-TxProcessAirbusLabel-LLR-002

Requirement ID: H398-LLD-GWY-FNC-6620

The function shall do the following

1. Call TxProcessLabel273T4AEORedDiode
2. Call TxProcessLabel274T4OEIYelDash
3. Call TxProcessLabel275T4OEIRedDash
4. Call TxProcessLabel276T4OEIRedLine
5. Call TxProcessLabel300T4STARTWhiTri
6. Call TxProcessLabel301T4STRTRdTriLo
7. Call TxProcessLabel302T4STRTRdTriHi
8. Call TxProcessLabel316EOT
9. Call TxProcessLabel317EOTAng
10. Call TxProcessLabel320EOP
11. Call TxProcessLabel321EOPAng
12. Call TxProcessLabel344DeltaNGAngle
13. Call TxProcessLabel343NG
14. Call TxProcessLabel345DltNGAEOYl
15. Call TxProcessLabel346DltNGAEOYlRc
16. Call TxProcessLabel347DeltaNGAEORed
17. Call TxProcessLabel350DltNGAEORdDi
18. Call TxProcessLabel351DltNGOEIYlDsh
19. Call TxProcessLabel352DltNGOEIRdDsh
20. Call TxProcessLabel353DltNGOEIRdLn
21. Call TxProcessLabel012WarningStatus
22. Call TxProcessLabel013WarningStatus
23. Call TxProcessLabel030ModeMgmt
24. Call TxProcessLabel034SysMsgStat
25. Call TxProcessLabel035DiscrepRpt
26. Call TxProcessLabel077LightingState
27. Call TxProcessLabel356EngineStatus1
28. Call TxProcessLabel357EngineStatus2
29. Call TxProcessLabel062DAUBIT
30. Call TxProcessLabel064CKSUMMSB
31. Call TxProcessLabel065CKSUMLSB
32. Call TxProcessLabel031Units

### 10.41.16 ConvertRange

Low Level Design Details about CSU convertRange will follow in the sub sections.

#### 10.41.16.1 Brief Description

This function converts an angle value from a 360-degree format to a 180-degree format if the

angle is greater than 180 degrees.

#### 10.41.16.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.41.16.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 10.41.16.4 Parameter list (Input/Output)

Inputs: positiveNum

Outputs: None

#### 10.41.16.5 Return Value

T\_FLOAT32- Returns Negative number

#### 10.41.16.6 Other CSUs called by this CSU

None

#### 10.41.16.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to convertRange

##### 10.41.16.7.1 daugwylogicairbus-ConvertRange-LLR-001

Requirement ID: H398-LLD-GWY-DRQ-6629

The function shall perform the following:

1. Set Negative\_number to positiveNum
2. Set Negative\_number to difference of (Negative\_number and M\_FP\_THREE\_HUNDRED\_SIXTY) when Negative\_number is greater than M\_FP\_HUNDRED\_EIGHTY.
3. Returns Negative\_number

### 10.41.17 TRQ12DisplayClass

Low Level Design Details about CSU TRQ12DisplayClass will follow in the sub sections

#### 10.41.17.1 Brief Description

This function sets the TRQ1+2 display class limit based on Engine States Number and sets the visibility of corresponding TRQ AEO and OEI limits.

#### 10.41.17.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.41.17.3 List of global variables accessed and modified

Accessed: None

Modified: Displayclass\_trq

#### 10.41.17.4 Parameter list (Input/Output)

None

#### 10.41.17.5 Return Value

None

#### 10.41.17.6 Other CSUs called by this CSU

None

#### 10.41.17.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TRQ12DisplayClass.

##### 10.41.17.7.1 daugwylogicairbus-TRQ12DisplayClass-LLR-001

Requirement ID: H398-LLD-GWY-FNC-6639

The function shall do the following

1. Set Displayclass\_trq to NO\_LIMITATION when Composite\_engine\_state is equal to ONE
2. Set Displayclass\_trq to OEI when Composite\_engine\_state Set to TWO.
3. Set Displayclass\_trq to OEI when Composite\_engine\_state Set to FIVE.
4. Set Displayclass\_trq to OEI when Composite\_engine\_state Set to SEVEN.
5. Set Displayclass\_trq  to OEI when Composite\_engine\_state Set to EIGHT.
6. Set Displayclass\_trq to OEI when Composite\_engine\_state Set to TWELVE.
7. Set Displayclass\_trq to OEI when Composite\_engine\_state Set to FIFTEEN.
8. Set Displayclass\_trq to OEI when Composite\_engine\_state Set to SIXTEEN.
9. Set Displayclass\_trq to OEI when Composite\_engine\_state Set to SEVENTEEN.
10. Set Displayclass\_trq to OEI\_AND\_AEO when Composite\_engine\_state set to THREE.
11. Set Displayclass\_trq to OEI\_AND\_AEO when Composite\_engine\_state set to FOUR.
12. Set Displayclass\_trq to OEI\_AND\_AEO when Composite\_engine\_state set to NINE.
13. Set Displayclass\_trq to OEI\_AND\_AEO when Composite\_engine\_state set to TEN.
14. Set Displayclass\_trq to OEI\_AND\_AEO when Composite\_engine\_state set to ELEVEN.
15. Set Displayclass\_trq to OEI\_AND\_AEO when Composite\_engine\_state set to THIRTEEN
16. Set Displayclass\_trq to OEI\_AND\_AEO when Composite\_engine\_state set to FOURTEEN.
17. Set Displayclass\_trq to OEI\_AND\_AEO when Composite\_engine\_state set to EIGHTEEN.
18. Set Displayclass\_trq to AEO when Composite\_engine\_state set to SIX.
19. Do nothing and break the statement when Composite\_engine\_state  is other than ONE, TWO, FIVE, SEVEN, EIGHT, NINE,

TWELVE, FIFTEEN, SIXTEEN, SEVENTEEN, THREE, FOUR, NINE, TEN, ELEVEN, THIRTEEN, FOURTEEN, EIGHTEEN and

SIX.

1. Set opr limits with index L052\_TRQ\_AEO\_YEL\_LO\_SDI01 to FALSE.
2. Set opr limits with index L052\_TRQ\_AEO\_YEL\_UP\_SDI10 to FALSE.
3. Set opr limits with index L053\_TRQ\_AEO\_YEL\_REC to FALSE.
4. Set opr limits with index L054\_TRQ\_AEO\_RED to FALSE.
5. Set opr limits with index L055\_TRQ\_AEO\_RED\_DIO to FALSE.
6. Set opr limits with index L057\_TRQ\_OEI\_YEL\_DASH to FALSE.
7. Set opr limits with index L060\_TRQ\_OEI\_RED\_DASH to FALSE.
8. Set opr limits with index L061\_TRQ\_OEI\_RED\_LINE to FALSE.

##### 10.41.17.7.2 daugwylogicairbus-TRQ12DisplayClass-LLR-002

Requirement ID: H398-LLD-GWY-FNC-6640

The function shall set the following when Displayclass\_trq is equal to AEO

1. Set Operational limits with index L052\_TRQ\_AEO\_YEL\_LO\_SDI01 to TRUE
2. Set Operational limits with index L052\_TRQ\_AEO\_YEL\_UP\_SDI10 to TRUE
3. Set Operational limits with index L054\_TRQ\_AEO\_RED to TRUE
4. Set Operational limits with index L055\_TRQ\_AEO\_RED\_DIO to TRUE

##### 10.41.17.7.3 daugwylogicairbus-TRQ12DisplayClass-LLR-003

Requirement ID: H398-LLD-GWY-FNC-7446

The function shall set the following when Displayclass\_trq is equal to OEI

1. Set Operational limits with index L057\_TRQ\_OEI\_YEL\_DASH to TRUE
2. Set Operational limits with index L060\_TRQ\_OEI\_RED\_DASH to TRUE
3. Set Operational limits with index L061\_TRQ\_OEI\_RED\_LINE to TRUE

##### 10.41.17.7.4 daugwylogicairbus-TRQ12DisplayClass-LLR-004

Requirement ID: H398-LLD-GWY-FNC-6641

The function shall set the following when Displayclass\_trq is equal to OEI\_AND\_AEO

1. Set Operational limits with index L052\_TRQ\_AEO\_YEL\_LO\_SDI01 to TRUE
2. Set Operational limits with index L052\_TRQ\_AEO\_YEL\_UP\_SDI10 to TRUE
3. Set Operational limits with index L054\_TRQ\_AEO\_RED to TRUE
4. Set Operational limits with index L057\_TRQ\_OEI\_YEL\_DASH to TRUE
5. Set Operational limits with index L060\_TRQ\_OEI\_RED\_DASH to TRUE
6. Set Operational limits with index L061\_TRQ\_OEI\_RED\_LINE to TRUE

### 10.41.18 CalculateEmbeddedResolution

Low Level Design Details about CSU CalculateEmbeddedResolution will follow in the sub sections

#### 10.41.18.1 Brief Description

This function calculates the data part with resolution only for data values and returns the calculated value.

#### 10.41.18.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.41.18.3 List of global variables accessed and modified

Accessed: Cmdownload\_lp\_cmd\_data

Modified: None

#### 10.41.18.4 Parameter list (Input/Output)

Inputs: T\_UINT32 index

Outputs: T\_FLOAT32 rawData

#### 10.41.18.5 Return Value

T\_FLOAT32 rawData

#### 10.41.18.6 Other CSUs called by this CSU

None

#### 10.41.18.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to CalculateEmbeddedResolution.

##### 10.41.18.7.1 daugwylogicairbus-CalculateEmbeddedResolution-LLR-001

Requirement ID: H398-LLD-GWY-DRQ-6651

The function shall perform the following:

1. Set rawData to rawData divided by rawdataPacketResolution of A429out\_bnr with index of Cmdownload\_lp\_cmd\_data

when rawdataPacketResolution of A429out\_bnr with index of Index of Cmdownload\_lp\_cmd\_data is not equal to M\_ZERO

2.Function returns rawData

### 10.41.20 TxProcessLabel034SysMsgStat

Low Level Design Details about CSU TxProcessLabel034SysMsgStat will follow in the sub sections.

#### 10.41.20.1 Brief Description

This function sets the Reconfigured Mode and Master discrepancies,channel interconnect failure, EDAU Internal Failure and sets the System Message Status data packet.

#### 10.41.20.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.41.20.3 List of global variables accessed and modified

Accessed: Tx\_a429\_bnr

reconfig\_discrepancy

AnalogMiscompareFail

System\_message

reconfiguration\_state\_sent

source\_selection\_sent

Mis\_cmp\_fail

Xram\_data\_buffer

mode\_sent

Modified: Mis\_cmp\_fail

Tx\_a429\_bnr

Xram\_data\_buffer

mode\_sent

#### 10.41.20.4 Parameter list (Input/Output)

None

#### 10.41.20.5 Return Value

None

#### 10.41.20.6 Other CSUs called by this CSU

None

#### 10.41.20.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TxProcessLabel034SysMsgStat

##### 10.41.20.7.1 daugwylogicairbus-TxProcessLabel034SysMsgStat-LLR-007

Requirement ID: H398-LLD-GWY-FNC-6676

The function shall do the following.

- Increment tenseconds counter and

- Set channel\_interconnect\_failure of Mis\_cmp\_fail to FALSE when tenseconds counter is less than( product of M\_TIMER\_TEMP and M\_TEN)

otherwise set channel\_interconnect\_failure of Mis\_cmp\_fail to NOT of Rs232\_transmit\_data

##### 10.41.20.7.2 daugwylogicairbus-TxProcessLabel034SysMsgStat-LLR-008

Requirement ID: H398-LLD-GWY-FNC-6677

The function shall loops from M\_ZERO to M\_A429\_MAX\_RX\_CHAN of u8 channelindex and perform the following

The function loops from M\_ZERO to numOf429in of Cmdownload\_lp\_cmd\_data minus one and sets EDAU channel of index u8 channelindex

to TRUE when MISSING is equal to status of Rx\_a429 of index [u8 channelindex] and [u8 labelindex] ,

otherwise set EDAU channel of index u8 channelindex to FALSE and beaks the loop

##### 10.41.20.7.3 daugwylogicairbus-TxProcessLabel034SysMsgStat-LLR-009

Requirement ID: H398-LLD-GWY-FNC-6678

The function shall loops from M\_ZERO to M\_A429\_MAX\_RX\_CHAN of u8 channelindex and perform the following

- Set channel DU of index u8 channelindex to TRUE when EDU channel of index u8 channelindex is set to TRUE,otherwise set channel DU of index u8 channelindex to FALSE.

##### 10.41.20.7.4 daugwylogicairbus-TxProcessLabel034SysMsgStat-LLR-010

Requirement ID: H398-LLD-GWY-FNC-6679

The function shall do the following.

- Set System\_message to DAU1\_FAIL\_STATUS when

(((Reconfig\_discrepancy is equal to TRUE) OR (Master\_discrepancy is equal to TRUE) OR (channel\_interconnect\_failure of Mis\_cmp\_fail is equal to TRUE) OR (Reconfig\_capability\_test\_fail is equal to TRUE)) AND (Source\_selection\_sent is equal to EIU\_2)) returns TRUE

##### 10.41.20.7.5 daugwylogicairbus-TxProcessLabel034SysMsgStat-LLR-011

Requirement ID: H398-LLD-GWY-FNC-6680

The function shall do the following.

- Set System\_messageto DAU2\_FAIL\_STATUS when

(((Reconfig\_discrepancy is equal to TRUE) OR (Master\_discrepancy is equal to TRUE) ) OR (channel\_interconnect\_failure of Mis\_cmp\_fail is equal to TRUE) OR (Reconfig\_capability\_test\_fail is equal to TRUE)) AND (Source\_selection\_sent is equal to EIU\_1)) returns TRUE.

##### 10.41.20.7.6 daugwylogicairbus-TxProcessLabel034SysMsgStat-LLR-012

Requirement ID: H398-LLD-GWY-FNC-6681

The function shall do the following.

- Set System\_message to CHECK\_DAU\_STATUS

when (Analog\_miscompare\_fail is TRUE) AND ( Reconfiguration\_state\_sent is equal to DAU\_N\_MODE )) OR

(channel\_interconnect\_failure of Mis\_cmp\_fail is equal to TRUE)) returns TRUE

##### 10.41.20.7.7 daugwylogicairbus-TxProcessLabel034SysMsgStat-LLR-013

Requirement ID: H398-LLD-GWY-FNC-6682

The function shall do the following.

- Set System\_message to PRIM\_VMD\_FAIL\_STATUS

when ((channel DU of index DU\_1 is equal to TRUE) OR (du cross monitor fail is equal to TRUE)).

##### 10.41.20.7.8 daugwylogicairbus-TxProcessLabel034SysMsgStat-LLR-014

Requirement ID: H398-LLD-GWY-FNC-6683

The function shall do the following.

- Set System\_message to B\_UP\_VMD\_FAIL\_STATUS

when ((channel DU of index DU\_2 is equal to TRUE) OR (remise du cross monitor fail is equal to TRUE)) otherwise Set System\_message to NO\_MSG

##### 10.41.20.7.9 daugwylogicairbus-TxProcessLabel034SysMsgStat-LLR-015

Requirement ID: H398-LLD-GWY-FNC-6684

The function shall do the following.

- Set id of Tx\_a429\_bnr of index LABEL\_034\_SYSTEM\_MESSAGE\_STATUS to M\_ZERO

- Set data package to Source\_selection\_sent.

- Set data package to BITWISE OR of data package and Source\_selection\_sent LEFT SHIFT by M\_TWO

- Set data package to BITWISE OR of data package and Reconfiguration\_state\_sent LEFT SHIFT by M\_FOUR.

- Set data package to BITWISE OR of data package and System\_message LEFT SHIFT by M\_SIX.

- Set data package to BITWISE OR of data package and Essential\_miscompare\_fail LEFT SHIFT by M\_TEN when ((Reconfiguration\_state\_sent is not equal to DAU\_1\_MODE) AND (Reconfiguration\_state\_sent is not equal to DAU\_2\_MODE)).

- Set data package to BITWISE OR of data package and Synthesis\_discrepancy LEFT SHIFT by M\_ELEVEN.

- Set data package to BITWISE OR of data package and Critical\_miscompare\_fail LEFT SHIFT by M\_TWELVE.

When ((Reconfiguration\_state\_sent is not equal to DAU\_1\_MODE) AND (Reconfiguration\_state\_sent is not equal to DAU\_2\_MODE)).

- Set data package to BITWISE OR of data package and Reconfig\_discrepancy LEFT SHIFT by M\_FOURTEEN.

- Set data package to BITWISE OR of data package and channel\_interconnect\_failure of Mis\_cmp\_fail LEFT SHIFT BY M\_FIFTEEN\_A.

- Set data of Tx\_a429\_bnr of index of LABEL\_034\_SYSTEM\_MESSAGE\_STATUS to data package

- Set status of Tx\_a429\_bnr of index of LABEL\_034\_SYSTEM\_MESSAGE\_STATUS to OK

##### 10.41.20.7.10 daugwylogicairbus-TxProcessLabel034SysMsgStat-LLR-016

Requirement ID: H398-LLD-GWY-FNC-7406

The function shall do the following.

- Set synthesis\_discrepancy to TRUE when (Essential\_miscompare\_fail is equal to TRUE) OR

(Critical\_miscompare\_fail is equal toTRUE) OR (reconfig\_discrepancy is equal to TRUE)) returns TRUE

Otherwise Set synthesis\_discrepancy to FALSE.

##### 10.41.20.7.11 daugwylogicairbus-TxProcessLabel034SysMsgStat-LLR-017

Requirement ID: H398-LLD-GWY-FNC-7469

The function shall set du cross monitor fail to TRUE when following conditions are satisfied

1. BITWISE AND of (data of Rx\_a429 with index of ((Priority with index of IP\_LABEL\_074\_CROSSMONITOR\_1) and IP\_LABEL\_074\_CROSSMONITOR\_1) and THREE) is equal to M\_ZERO .
2. status of Rx\_a429 with index of ((Priority with index of IP\_LABEL\_074\_CROSSMONITOR\_1) and IP\_LABEL\_074\_CROSSMONITOR\_1) is equal to OK.

Otherwise set du cross monitor fail to TRUE.

##### 10.41.20.7.12 daugwylogicairbus-TxProcessLabel034SysMsgStat-LLR-018

Requirement ID: H398-LLD-GWY-FNC-7470

The function shall set remise du cross monitor fail to TRUE when the following conditions are satisfied

1. BITWISE AND of (data of Rx\_a429 with index of ((Priority with index of IP\_LABEL\_074\_CROSSMONITOR\_2) and IP\_LABEL\_074\_CROSSMONITOR\_2) and THREE) is equal to M\_ZERO .
2. status of Rx\_a429 with index of ((Priority with index of IP\_LABEL\_074\_CROSSMONITOR\_2) and IP\_LABEL\_074\_CROSSMONITOR\_2) is equal to OK.

Otherwise set remise du cross monitor fail to TRUE.

### 10.41.21 TxProcessLabel050TRQ

Low Level Design Details about CSU TxProcessLabel050TRQ will follow in the sub sections

#### 10.41.21.1 Brief Description

This function processes and transmits digital value data packets related to Engine 1, Engine 2, and TRQ1+2.

It considers various engine states and operation limits to set the appropriate status and values for transmission.

#### 10.41.21.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.41.21.3 List of global variables accessed and modified

Accessed: Tx\_a429\_bnr

Engine\_oei\_state

Composite\_engine\_state

Dau\_eng\_data

Modified: Tx\_a429\_bnr

#### 10.41.21.4 Parameter list (Input/Output)

None

#### 10.41.21.5 Return Value

None

#### 10.41.21.6 Other CSUs called by this CSU

None

#### 10.41.21.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TxProcessLabel050TRQ

##### 10.41.21.7.1 daugwylogicairbus-TxProcessLabel050TRQ-LLR-001

Requirement ID: H398-LLD-GWY-FNC-6693

The function shall loops from M\_ZERO to M\_MAX\_ENGINE minus one and performs the following.

1. Set id of Tx\_a429\_bnr of index LABEL\_050\_TRQ\_1 with loop index to loop index with M\_ONE.
2. Set data package to ANALOG\_SRC.
3. Set Number Status to DIGIT\_NOMINAL.

##### 10.41.21.7.2 daugwylogicairbus-TxProcessLabel050TRQ-LLR-002

Requirement ID: H398-LLD-GWY-FNC-6694

The function shall loops from M\_ZERO to M\_MAX\_ENGINE minus one and performs the following:

when( ENG\_OEI\_State of index [M\_ENGINE\_1 plus loop index] of Engine\_oei\_state is TRUE) OR (ENG\_OEI\_Training\_State of index

[M\_ENGINE\_1 plus loop index]of Engine\_oei\_state is set to TRUE) AND (Dau\_eng\_data of index ( TRQ1 plus loop index ) is greater than

M\_TRQ\_OEI\_CT\_LIMIT and lesser than or equal to M\_TRQ\_OEI\_LO\_LIMIT) returns TRUE

1. Number Status is set to DIGIT\_AMBER\_COLOR.

##### 10.41.21.7.3 daugwylogicairbus-TxProcessLabel050TRQ-LLR-003

Requirement ID: H398-LLD-GWY-FNC-6695

The function shall loops from M\_ZERO to M\_MAX\_ENGINE minus one and performs  the following:

1. Set  Number Status to DIGIT\_RED\_COLOR when( ENG\_OEI\_State of index [M\_ENGINE\_1 plus loop index] of Engine\_oei\_state is TRUE) OR (ENG\_OEI\_Training\_State of index [M\_ENGINE\_1 plus loop index] of Engine\_oei\_state is set to TRUE) AND (Dau\_eng\_data of index ( TRQ1 plus loop index ) is greater than M\_TRQ\_OEI\_LO\_LIMIT) returns TRUE.
2. Set data package to data package of bitwise OR of Number Status left shift M\_ONE.
3. Set Number status Blink to NOT\_BLINKING.
4. Set data package to data package of bitwise OR of Number status Blink LEFT shift M\_THREE.
5. Set torque rounded value with index loop\_index to return value of TRQRoundup with parameters Dau\_eng\_data index of TRQ1 added with loop\_index.
6. Set data package to Bitwise OR of data package with return value of Calculate Embedded Resolution with parameters Torque rounded value with index of loop index and LABEL\_050\_TRQ\_1 is left shifted by M\_FOUR.
7. Set  data of Tx\_a429\_bnr of index LABEL\_050\_TRQ\_1 with loop index to data package.
8. Set status of Tx\_a429\_bnr of index LABEL\_050\_TRQ\_1 with loop index to OK when Dau\_eng\_status with index (LABEL\_050\_TRQ\_1 plus loop index) is equal to OK, otherwise set status of Tx\_a429\_bnr of index LABEL\_050\_TRQ\_1 with loop index to CHAN\_ERR.

##### 10.41.21.7.4 daugwylogicairbus-TxProcessLabel050TRQ-LLR-004

Requirement ID: H398-LLD-GWY-FNC-6696

The function shall do the following.

1. Set id of Tx\_a429\_bnr of index LABEL\_050\_TRQ\_12 to M\_ENGINE\_1\_2.
2. Set data package to ANALOG\_SRC.
3. Set torque data with index of M\_ENGINE\_1 to Dau engine data with index of TRQ1.
4. Set torque data with index of M\_ENGINE\_2 to Dau engine data with index of TRQ2.

##### 10.41.21.7.5 daugwylogicairbus-TxProcessLabel050TRQ-LLR-005

Requirement ID: H398-LLD-GWY-FNC-6697

The function shall do the following when Composite\_engine\_state is equal to SIX OR ENG\_OEI\_State of index M\_ENGINE\_1 of Engine\_oei\_state is equal to TRUE OR ENG\_OEI\_State of index M\_ENGINE\_2 of Engine\_oei\_state is equal to TRUE.

1. Set torque data with index ZERO to M\_ZERO when analog\_input\_miscompare with index of (M\_TEN plus TRQ\_1) of Mis\_cmp\_fail is equal to TRUE OR Dau\_eng\_status with index TRQ1 is not equal to OK.
2. Set torque data with index ONE to M\_ZERO when analog\_input\_miscompare with index of (M\_TEN plus TRQ\_2) of Mis\_cmp\_fail is equal to TRUE OR Dau\_eng\_status with index TRQ2 is not equal to OK.

##### 10.41.21.7.6 daugwylogicairbus-TxProcessLabel050TRQ-LLR-006

Requirement ID: H398-LLD-GWY-FNC-7451

The function shall perform the following

1. Set Torque to torque data with index ZERO when Engine\_state\_logic with index M\_ENGINE\_1 is equal to OEI\_TF AND Engine\_state\_logic with index M\_ENGINE\_2 is equal to OEI\_TI.
2. Set Torque to torque data with index ONE When Engine\_state\_logic with index M\_ENGINE\_1 is

equal to OEI\_TI AND Engine\_state\_logic with index M\_ENGINE\_2 is equal to OEI\_TF,otherwise set

Torque to sum of torque data with index ZERO and torque data with index ONE.

1. Set Computed\_torque12 to torque.
2. Set Number Status to DIGIT\_NOMINAL.

##### 10.41.21.7.7 daugwylogicairbus-TxProcessLabel050TRQ-LLR-007

Requirement ID: H398-LLD-GWY-FNC-6698

The function shall set numstatus to DIGIT\_AMBER\_COLOR

1. When ENG\_OEI\_Training\_State of index M\_ENGINE\_1 of Engine\_oei\_state OR ENG\_OEI\_Training\_State of   
   index M\_ENGINE\_2 of Engine\_oei\_state is TRUE and torque is greater than M\_TRQ\_OEI\_CT\_LIMIT AND torque is  
    lesser than or equal to M\_TRQ\_OEI\_LO\_LIMIT OR Composite\_engine\_state is equal to SIX, torque   
   is greater than M\_TRQ\_MCP\_LIMIT and torque lesser than or equal to M\_TRQ\_TOP\_LIMIT OR OEI\_AND\_AEO is equal to Displayclass\_trq AND torque is greater than M\_TRQ\_MCP\_LIMIT and torque lesser than or equal to M\_TRQ\_TOP\_LIMIT.   
   Otherwise, the function shall set to DIGIT\_RED\_COLOR   
   when either ENG\_OEI\_Training\_State of index M\_ENGINE\_1 of Engine\_oei\_state OR ENG\_OEI\_Training\_State   
   of index M\_ENGINE\_2 of Engine\_oei\_statesi equal to TRUE and  
   torque is greater than M\_TRQ\_OEI\_LO\_LIMIT OR Composite\_engine\_state is equal to SIX AND torque greater than M\_TRQ\_TOP\_LIMIT and OR OEI\_AND\_AEO is equal to Displayclass\_trq AND torque is greater than M\_TRQ\_TOP\_LIMIT.
2. Set data package to data package of bitwise OR of Number Status left shift M\_ONE.
3. Set Number status blink to NOT\_BLINKING.
4. Set data package to data package of bitwise OR of Number status blink left shift M\_THREE.
5. Set torque to return value by calling the function TRQRoundup with parameter (torque)
6. Set torque to return value by calling the function Calculate Embedded Resolution with parameters (torque and LABEL\_050\_TRQ\_12).
7. Set data package to data package of bitwise OR of torque left shift M\_FOUR.
8. Set data of Tx\_a429\_bnr of index LABEL\_050\_TRQ\_12 to data package.

##### 10.41.21.7.8 daugwylogicairbus-TxProcessLabel050TRQ-LLR-008

Requirement ID: H398-LLD-GWY-FNC-6699

The function shall perform the following:

1. set torque Status to OK when either Dau\_eng\_status of index TRQ1 and Dau\_eng\_status of index TRQ2 is equal to OK. Otherwise, torque Status is set to CHAN\_ERR.
2. Set status of Tx\_a429\_bnr of index LABEL\_050\_TRQ\_12 of Tx\_a429\_bnr to torque status.

### 10.41.22 TxProcessLabel051TRQAngle

Low Level Design Details about CSU TxProcessLabel051TRQAngle will follow in the sub sections.

#### 10.41.22.1 Brief Description

This function processes the TRQ angle data and prepares the data packet:Label 051-11: Engine 1 & 2 TRQ Pointer Angle Value Data Packet.

#### 10.41.22.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.41.22.3 List of global variables accessed and modified

Accessed: Tx\_a429\_bnr

Modified: Tx\_a429\_bnr

#### 10.41.22.4 Parameter list (Input/Output)

None

#### 10.41.22.5 Return Value

None

#### 10.41.22.6 Other CSUs called by this CSU

LookupTableLookup

#### 10.41.22.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TxProcessLabel051TRQAngle

##### 10.41.22.7.1 daugwylogicairbus-TxProcessLabel051TRQAngle-LLR-001

Requirement ID: H398-LLD-GWY-FNC-6708

The function shall do the following.

1. Set pi32\_x of ltable to table x
2. Set pi32\_y of ltable to table y
3. Set id of Tx\_a429\_bnr of index LABEL\_051\_TRQ\_ANGLE\_12 to M\_ENGINE\_1\_2
4. Set torque Angle to return by calling the function Lookup Table Lookup with parameters(address of ltable, Computed\_torque12 and address of u8 status).
5. Set torque Angle to computed Torque Angle.
6. Set torque Angle in range to return of convert Range with parameter torque Angle.
7. Set compute torque Angle in range to torque Angle in range.
8. Set data package to torque Angle in range left shift M\_SEVEN.
9. Set data of Tx\_a429\_bnr of LABEL\_051\_TRQ\_ANGLE\_12 to return value by   
   calling the function Calculate Embedded Resolution with parameters (data package and LABEL\_051\_TRQ\_ANGLE\_12.
10. Set status of Tx\_a429\_bnr of index LABEL\_051\_TRQ\_ANGLE\_12 to torque Status.

### 10.41.23 TxProcessLabel052TRQAEOYellow

Low Level Design Details about CSU TxProcessLabel052TRQAEOYellow will follow in the sub sections.

#### 10.41.23.1 Brief Description

#### This function calculates and sets the TRQ AEO YELLOW value data packet for labels 052-01 and 052-10.

10.41.23.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.41.23.3 List of global variables accessed and modified

Accessed: Tx\_a429\_bnr

Modified: Tx\_a429\_bnr

#### 10.41.23.4 Parameter list (Input/Output)

None

#### 10.41.23.5 Return Value

None

#### 10.41.23.6 Other CSUs called by this CSU

None

#### 10.41.23.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TxProcessLabel052TRQAEOYellow

##### 10.41.23.7.1 daugwylogicairbus-TxProcessLabel052TRQAEOYellow-LLR-001

Requirement ID: H398-LLD-GWY-FNC-7447

The function shall do the following.

1. Set torque aeoyellow angle value with index M\_ZERO to the return value by calling the function ConvertRange with parameter(M\_TRQ\_AEO\_YEL\_LO).
2. Set torque aeoyellow angle value with index M\_ONE to the return value by calling the function ConvertRange with parameter(M\_TRQ\_AEO\_YEL\_UP).

##### 10.41.23.7.2 daugwylogicairbus-TxProcessLabel052TRQAEOYellow-LLR-002

Requirement ID: H398-LLD-GWY-FNC-6717

The function shall loop from M\_ZERO to M\_TWO minus one :

1. Set id of Tx\_a429\_bnr of index LABEL\_052\_TRQ\_AEO\_YEL\_LO with loop index to M\_ONE with loop index.
2. Set data package of index of loop index to operation limits of L052\_TRQ\_AEO\_YEL\_LO\_SDI01 with loop index.
3. Set data package of index of loop index of bitwise OR of return value by calling the function   
    of Calculate Embedded Resolution with parameters(torque aeoyellow angle value with index   
   loop index,and LABEL\_052\_TRQ\_AEO\_YEL\_LO plus loopindex)) LEFT SHIFT by M\_SEVEN.
4. Set data of Tx\_a429\_bnr of index LABEL\_052\_TRQ\_AEO\_YEL\_LO with loop index to data package of index loop index.
5. Set status of Tx\_a429\_bnr of index of LABEL\_052\_TRQ\_AEO\_YEL\_LO with loop index to torque Status.

### 10.41.24 TxProcessLabel053TRQAEOYelRec

Low Level Design Details about CSU TxProcessLabel053TRQAEOYelRec will follow in the sub sections.

#### 10.41.24.1 Brief Description

This function calculates and sets the TRQ AEO YELLOW rectangle value data packet for label 053-00.

#### 10.41.24.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.41.24.3 List of global variables accessed and modified

Accessed: Tx\_a429\_bnr

Modified: Tx\_a429\_bnr

#### 10.41.24.4 Parameter list (Input/Output)

None

#### 10.41.24.5 Return Value

None

#### 10.41.24.6 Other CSUs called by this CSU

None

#### 10.41.24.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TxProcessLabel053TRQAEOYelRec

##### 10.41.24.7.1 daugwylogicairbus-TxProcessLabel053TRQAEOYelRec-LLR-001

Requirement ID: H398-LLD-GWY-FNC-6726

The function shall do the following.

1. Set id of Tx\_a429\_bnr of index LABEL\_053\_TRQ\_AEO\_YEL\_REC to M\_ZERO.
2. Set data package to operation limits of L053\_TRQ\_AEO\_YEL\_REC.
3. Set data package to data package of bitwise OR of (Torque AEO Yellow Record Value LEFT SHIFT by M\_SEVEN) .
4. Set data of Tx\_a429\_bnr of LABEL\_053\_TRQ\_AEO\_YEL\_REC to the return value by  
    calling the function CalculateEmbeddedResolution with parameters (data package and LABEL\_053\_TRQ\_AEO\_YEL\_REC) .
5. Set status of Tx\_a429\_bnr of index LABEL\_053\_TRQ\_AEO\_YEL\_REC to torque Status.

### 10.41.25 TxProcessLabel054TRQAEORed

Low Level Design Details about CSU TxProcessLabel054TRQAEORed will follow in the sub sections.

#### 10.41.25.1 Brief Description

This function processes the TRQ AEO Red data and prepares the data packet: Label 054-00 - TRQ AEO Red value Data Packet

#### 10.41.25.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.41.25.3 List of global variables accessed and modified

Accessed: Tx\_a429\_bnr

Modified: Tx\_a429\_bnr

#### 10.41.25.4 Parameter list (Input/Output)

None

#### 10.41.25.5 Return Value

None

#### 10.41.25.6 Other CSUs called by this CSU

None

#### 10.41.25.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TxProcessLabel054TRQAEORed

##### 10.41.25.7.1 daugwylogicairbus-TxProcessLabel054TRQAEORed-LLR-001

Requirement ID: H398-LLD-GWY-FNC-6735

The function shall do the following.

1. Set Torque AEO Red Angle Value to return value by calling the function ConvertRange with parameter(M\_TRQ\_AEO\_RED).
2. Set id of Tx\_a429\_bnr of index LABEL\_054\_TRQ\_AEO\_RED to M\_ZERO.
3. Set data package to operation limits of index L054\_TRQ\_AEO\_RED.
4. Set data package to data package of bitwise OR of (return value by calling the function CalculateEmbeddedResolution   
   with parameters (Torque AEO Red Angle Value and LABEL\_054\_TRQ\_AEO\_RED) LEFT SHIFT by M\_SEVEN).
5. Set data of Tx\_a429\_bnr of index LABEL\_054\_TRQ\_AEO\_RED to data package.
6. Set status of Tx\_a429\_bnr of index LABEL\_054\_TRQ\_AEO\_RED to torque Status.

### 10.41.26 TxProcessLabel055TRQAEORdDi

Low Level Design Details about CSU TxProcessLabel055TRQAEORdDi will follow in the sub sections.

#### 10.41.26.1 Brief Description

This function processes the TRQ AEO Red Diode data and prepares the data packet:

Label 055-00 - TRQ AEO Red Diode value Data Packet

#### 10.41.26.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.41.26.3 List of global variables accessed and modified

Accessed: Tx\_a429\_bnr

Modified: Tx\_a429\_bnr

#### 10.41.26.4 Parameter list (Input/Output)

None

#### 10.41.26.5 Return Value

None

#### 10.41.26.6 Other CSUs called by this CSU

None

#### 10.41.26.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TxProcessLabel055TRQAEORdDi

##### 10.41.26.7.1 daugwylogicairbus-TxProcessLabel055TRQAEORdDi-LLR-001

Requirement ID: H398-LLD-GWY-FNC-6744

The function shall do the following.

1. Set torque AEO Red Diode Value to return value by calling the function ConvertRange with parameter M\_TRQ\_AEO\_RED\_DIO.
2. Set id Tx\_a429\_bnr of index LABEL\_055\_TRQ\_AEO\_RED\_DIO to M\_ZERO.
3. Set data package to operation limits of L055\_TRQ\_AEO\_RED\_DIO.
4. Set torque AEO Red Dio Sel to TRIANGLE.
5. Set data package to data package of bitwise OR of torque AEO Red Dio Sel left shift M\_ONE.
6. Set data package to data package of bitwise OR (to return value by calling the function   
   CalculateEmbeddedResolution with parameters(torque AEO Red Diode Value and  
   LABEL\_055\_TRQ\_AEO\_RED\_DIO) LEFT SHIFT by M\_SEVEN).
7. Set data of Tx\_a429\_bnr of index LABEL\_055\_TRQ\_AEO\_RED\_DIO to data package.
8. Set status of Tx\_a429\_bnr of index LABEL\_055\_TRQ\_AEO\_RED\_DIO to torque Status.

### 10.41.27 TxProcessLabel057TRQOEIYelDash

Low Level Design Details about CSU TxProcessLabel057TRQOEIYelDash will follow in the sub sections.

#### 10.41.27.1 Brief Description

This function processes the TRQ OEI Yellow Dash data and prepares the data packet:

Label 057-00 - TRQ OEI Yellow Dash value Data Packet.

#### 10.41.27.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.41.27.3 List of global variables accessed and modified

Accessed: Tx\_a429\_bnr

Modified: Tx\_a429\_bnr

#### 10.41.27.4 Parameter list (Input/Output)

None

#### 10.41.27.5 Return Value

None

#### 10.41.27.6 Other CSUs called by this CSU

None

#### 10.41.27.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TxProcessLabel057TRQOEIYelDash

##### 10.41.27.7.1 daugwylogicairbus-TxProcessLabel057TRQOEIYelDash-LLR-001

Requirement ID: H398-LLD-GWY-FNC-6753

The function shall do the following.

1. Set torque oei yellow dashvalue to the return value by calling the function ConvertRange with parameter (M\_TRQ\_OEI\_YEL\_DASH) .
2. Set id of Tx\_a429\_bnr of index LABEL\_057\_TRQ\_OEI\_YEL\_DASH to M\_ZERO
3. Set data package to operation limits of index of L057\_TRQ\_OEI\_YEL\_DASH.
4. Set data package to data package of bitwise OR of torque OEI Low Diode left shift M\_ONE.
5. Set data package to data package of bitwise OR to the( return value by calling the function   
   CalculateEmbeddedResolution with parameters(torque oei yellow dashvalue and  
   LABEL\_057\_TRQ\_OEI\_YEL\_DASH) LEFT SHIFT by M\_SEVEN).
6. Set data of Tx\_a429\_bnr of index LABEL\_057\_TRQ\_OEI\_YEL\_DASH to data package.
7. Set status of Tx\_a429\_bnr of index LABEL\_057\_TRQ\_OEI\_YEL\_DASH to torque Status.

### 10.41.28 TxProcessLabel060TRQOEIRedDash

Low Level Design Details about CSU TxProcessLabel060TRQOEIRedDash will follow in the sub sections.

#### 10.41.28.1 Brief Description

This function processes the TRQ OEI Red Dash data and prepares the data packet:

Label 060-00 - TRQ OEI Red Dash value Data Packet

#### 10.41.28.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.41.28.3 List of global variables accessed and modified

Accessed : Tx\_a429\_bnr

Modified : Tx\_a429\_bnr

#### 10.41.28.4 Parameter list (Input/Output)

None

#### 10.41.28.5 Return Value

None

#### 10.41.28.6 Other CSUs called by this CSU

None

#### 10.41.28.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TxProcessLabel060TRQOEIRedDash.

##### 10.41.28.7.1 daugwylogicairbus-TxProcessLabel060TRQOEIRedDash-LLR-001

Requirement ID: H398-LLD-GWY-FNC-6762

The function shall perform the following:

1. Call convertRange with parameter M\_TRQ\_OEI\_RED\_DASH and set the return value to trqoei\_reddash\_value.
2. Set id of Tx\_a429\_bnr with index LABEL\_060\_TRQ\_OEI\_RED\_DASH to M\_ZERO.
3. Set datapackage to operation\_limit with index L060\_TRQ\_OEI\_RED\_DASH.
4. Update datapackage with BITWISE OR of datapackage and (return value from function CalculateEmbeddedResolution with parameters (trqoei\_reddash\_value and LABEL\_060\_TRQ\_OEI\_RED\_DASH) LEFTSHIFT by M\_SEVEN).
5. Set data of Tx\_a429\_bnr with index LABEL\_060\_TRQ\_OEI\_RED\_DASH to datapackage.
6. Set status of Tx\_a429\_bnr with index LABEL\_060\_TRQ\_OEI\_RED\_DASH to torque\_12\_status.

### 10.41.29 TxProcessLabel061TRQOEIRedLine

Low Level Design Details about CSU TxProcessLabel061TRQOEIRedLine will follow in the sub sections.

#### 10.41.29.1 Brief Description

This function processes the TRQ OEI Red Line data and prepares the data packet:

Label 061-00 - TRQ OEI Red Line value Data Packet.

#### 10.41.29.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.41.29.3 List of global variables accessed and modified

Accessed : Tx\_a429\_bnr

Modified : Tx\_a429\_bnr

#### 10.41.29.4 Parameter list (Input/Output)

None

#### 10.41.29.5 Return Value

None

#### 10.41.29.6 Other CSUs called by this CSU

None

#### 10.41.29.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TxProcessLabel061TRQOEIRedLine

##### 10.41.29.7.1 daugwylogicairbus-TxProcessLabel061TRQOEIRedLine-LLR-001

Requirement ID: H398-LLD-GWY-FNC-6771

The function shall perform the following:

1. Call convertRange with parameter M\_TRQ\_OEI\_RED\_LINE and set the return value to trqoei\_redline\_value.
2. Set id of Tx\_a429\_bnr with index LABEL\_061\_TRQ\_OEI\_RED\_LINE to M\_ZERO.
3. Set datapackage to operation\_limit with index L061\_TRQ\_OEI\_RED\_LINE.
4. Update datapackage with BITWISE OR of datapackage and (return value from function   
   CalculateEmbeddedResolution with parameters (trqoei\_redline\_value and LABEL\_061\_TRQ\_OEI\_RED\_LINE) LEFT SHIFT by M\_SEVEN).
5. Set data of Tx\_a429\_bnr with index LABEL\_061\_TRQ\_OEI\_RED\_LINE to datapackage.
6. Set status of Tx\_a429\_bnr with index LABEL\_061\_TRQ\_OEI\_RED\_LINE to torque\_12\_status.

### 10.41.30 TxProcessLabel174HYD

Low Level Design Details about CSU TxProcessLabel174HYD will follow in the sub sections.

#### 10.41.30.1 Brief Description

This function processes the hydraulic pressure data for engines 1 and 2 and prepares two data packets:

Label 174-01 - Engine 1 Hydraulic Pressure Digital Value Data Packet and Label 174-10 - Engine 2 Hydraulic Pressure Digital Value Data Packet.

#### 10.41.30.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD(H398-003-012-GWY).

#### 10.41.30.3 List of global variables accessed and modified

Accessed : Dau\_eng\_data

Dau\_eng\_status

Modified : Tx\_a429\_bnr

#### 10.41.30.4 Parameter list (Input/Output)

None

#### 10.41.30.5 Return Value

None

#### 10.41.30.6 Other CSUs called by this CSU

None

#### 10.41.30.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TxProcessLabel174HYD

##### 10.41.30.7.1 daugwylogicairbus-TxProcessLabel174HYD-LLR-001

Requirement ID: H398-LLD-GWY-FNC-6781

The function shall perform the following:

1. Set id of Tx\_a429\_bnr with index LABEL\_174\_HYD\_1 to M\_ONE.
2. Set hydraulic\_status to DIGIT\_RED\_COLOR when

(Dau\_eng\_data with index LHP\_SIG is less than M\_HYD\_PRESS\_LOW\_RED\_LIMIT) OR (Dau\_eng\_data with index

LHP\_SIG is greater than M\_HYD\_PRESS\_HIGH\_RED\_LIMIT) results TRUE , otherwise set to DIGIT\_NOMINAL.

c) Set datapackage with index M\_ZERO to BITWISE OR of datapackage with index M\_ZERO and (hydraulic\_status LEFTSHIFT by M\_ONE).

d) Update datapackage with index as M\_ZERO with BITWISE OR of datapackage with index as M\_ZERO and (numstatusblink LEFTSHIFT by M\_THREE).

e) Update datapackage with index as M\_ZERO with BITWISE OR of datapackage with index as

M\_ZERO and (return value from function CalculateEmbeddedResolution with parameters (Dau\_eng\_status with index LHP\_SIG and LABEL\_174\_HYD\_1) LEFTSHIFT by M\_SIX).

f) Set data of Tx\_a429\_bnr with index LABEL\_174\_HYD\_1 to datapackage with index as M\_ZERO.

g) Set status of Tx\_a429\_bnr with index LABEL\_174\_HYD\_1 to OK when Dau\_eng\_status with index LHP\_SIG is equal to OK, otherwise set status of Tx\_a429\_bnr with index LABEL\_174\_HYD\_1 to CHAN\_ERR.

##### 10.41.30.7.2 daugwylogicairbus-TxProcessLabel174HYD-LLR-002

Requirement ID: H398-LLD-GWY-FNC-6782

The function shall perform the following:

1. Set id of Tx\_a429\_bnr with index LABEL\_174\_HYD\_2 to M\_TWO.
2. Set hydraulic\_status to DIGIT\_RED\_COLOR when

(Dau\_eng\_data with index RHP\_SIG is less than M\_HYD\_PRESS\_LOW\_RED\_LIMIT) OR (Dau\_eng\_data with index RHP\_SIG is greater than M\_HYD\_PRESS\_HIGH\_RED\_LIMIT) results TRUE , otherwise set to DIGIT\_NOMINAL.

c) Set datapackage with index M\_ONE to BITWISE OR of datapackage with index M\_ONE and (hydraulic\_status LEFTSHIFT by M\_ONE).

d) Update datapackage with index as M\_ONE with BITWISE OR of datapackage with index as M\_ONE and (numstatusblink LEFTSHIFT by M\_THREE).

e) Update datapackage with index as M\_ONE with BITWISE OR of datapackage with index as M\_ONE and (return value from function CalculateEmbeddedResolution with parameters (Dau\_eng\_data with index RHP\_SIG and LABEL\_174\_HYD\_2) LEFTSHIFT by M\_SIX).

f) Set data of Tx\_a429\_bnr with index LABEL\_174\_HYD\_2 to datapackage with index as M\_ONE.

g) Set status of Tx\_a429\_bnr with index LABEL\_174\_HYD\_2 to OK when Dau\_eng\_status with index RHP\_SIG is equal to OK, otherwise set status of Tx\_a429\_bnr with index LABEL\_174\_HYD\_2 to CHAN\_ERR.

### 10.41.31 TxProcessLabel175HYDAng

Low Level Design Details about CSU TxProcessLabel175HYDAng will follow in the sub sections.

#### 10.41.31.1 Brief Description

This function processes the hydraulic pressure analog data for engines 1 and 2 and prepares two data packets:

Label 175-01 - Engine 1 Hydraulic Pressure Analog Value Data Packet and Label 175-10 - Engine 2 Hydraulic Pressure Analog Value Data Packet.

#### 10.41.31.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.41.31.3 List of global variables accessed and modified

Accessed : Dau\_eng\_data

Modified : Tx\_a429\_bnr

#### 10.41.31.4 Parameter list (Input/Output)

Inputs: None

Outputs: None

#### 10.41.31.5 Return Value

None

#### 10.41.31.6 Other CSUs called by this CSU

LookupTableLookup

#### 10.41.31.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TxProcessLabel175HYDAng

##### 10.41.31.7.1 daugwylogicairbus-TxProcessLabel175HYDAng-LLR-001

Requirement ID: H398-LLD-GWY-FNC-6791

The function shall perform the following:

1. Set id of Tx\_a429\_bnr with index LABEL\_175\_HYD\_ANG\_1 to M\_ONE

b) Set hydraulic\_analog of index M\_ZERO to M\_HYD\_PRESS\_BTM\_THIN\_RED\_LINE When the following conditions are satisfied:

1. Dau\_eng\_data with index LHP\_SIG greater than or equal to M\_ZERO
2. Dau\_eng\_data with index LHP\_SIG less than M\_HYD\_PRESS\_LOW\_RED\_LIMIT

##### 10.41.31.7.2 daugwylogicairbus-TxProcessLabel175HYDAng-LLR-002

Requirement ID: H398-LLD-GWY-FNC-6792

The function shall set the following:

1. Set pi32X of Ltable to Table1 X
2. Set pi32Y of Ltable to Table1 Y
3. Set Hydraulic analog with index M\_ZERO to return value of LookupTableLookup with parameters (reference to Ltable, Dau\_eng\_data with index LHP\_SIG and reference to u8status)

When the following conditions are satisfied:

1. Dau\_eng\_data with index LHP\_SIG greater than or equal to M\_HYD\_PRESS\_LOW\_RED\_LIMIT
2. Dau\_eng\_data with index LHP\_SIG less than or equal to M\_HYD\_PRESS\_MIDDLE

##### 10.41.31.7.3 daugwylogicairbus-TxProcessLabel175HYDAng-LLR-003

Requirement ID: H398-LLD-GWY-FNC-6793

The function shall set the following:

1. Set pi32X of Ltable to Table2 X
2. Set pi32Y of Ltable to Table2 Y
3. Set Hydraulic analog with index M\_ZERO to return value of LookupTableLookup with parameters (reference to Ltable, Dau\_eng\_data with index LHP\_SIG and reference to u8status)

When the following conditions are satisfied:

1. Dau\_eng\_data with index LHP\_SIG greater than M\_HYD\_PRESS\_MIDDLE
2. Dau\_eng\_data with index LHP\_SIG less than or equal to M\_HYD\_PRESS\_HIGH\_RED\_LIMIT.

##### 10.41.31.7.4 daugwylogicairbus-TxProcessLabel175HYDAng-LLR-004

Requirement ID: H398-LLD-GWY-FNC-6794

The function shall set the following:

-Set hydraulic\_analog of index M\_ZERO to M\_HYD\_PRESS\_TOP\_THIN\_RED\_LINE

When

the following conditions are satisfied:

1. Dau\_eng\_data with index LHP\_SIG greater than M\_HYD\_PRESS\_HIGH\_RED\_LIMIT
2. Dau\_eng\_data with index LHP\_SIG less than or equal to M\_TWO\_HUNDREAD\_FIFTY.

##### 10.41.31.7.5 daugwylogicairbus-TxProcessLabel175HYDAng-LLR-005

Requirement ID: H398-LLD-GWY-FNC-6795

The function shall perorm the following:

1. Set data package with index M\_ZERO to Btmthinredline
2. Update data package with index M\_ZEROto bitwise OR of data package with index M\_ZEROand (Topthinredline left shift by M\_THREE)
3. Update data package with index M\_ZERO to bitwise OR of data package with index M\_ZERO and call   
   CalculateEmbeddedResolution with parameter (hydraulic analog with index M\_ZERO, LABEL\_175\_HYD\_ANG\_1) left shift by M\_SIX
4. Set data of Tx\_a429\_bnr with index LABEL\_175\_HYD\_ANG\_1 to data package with index M\_ZERO
5. Set status of Tx\_a429\_bnr with index LABEL\_175\_HYD\_ANG\_1 to OK when Dau\_eng\_status with index LHP\_SIG is equal to OK, otherwise set status of Tx\_a429\_bnr with index LABEL\_175\_HYD\_ANG\_1 to CHAN\_ERR.

##### 10.41.31.7.6 daugwylogicairbus-TxProcessLabel175HYDAng-LLR-006

Requirement ID: H398-LLD-GWY-FNC-6796

The function shall set the following:

1. Set id of Tx\_a429\_bnr with index LABEL\_175\_HYD\_ANG\_2 to M\_TWO
2. Set hydraulic analog with index ONE to M\_HYD\_PRESS\_BTM\_THIN\_RED\_LINE and btmthinredline to TRUE  
   when Dau\_eng\_data with index RHP\_SIG is equal to M\_HYD\_PRESS\_LOW\_RED\_LIMIT AND with Dau\_eng\_data of index RHP\_SIG is less than M\_HYD\_PRESS\_LOW\_RED\_LIMIT

##### 10.41.31.7.7 daugwylogicairbus-TxProcessLabel175HYDAng-LLR-007

Requirement ID: H398-LLD-GWY-FNC-6797

The function shall set the following:

1. Set pi32X of Ltable to Table1 X
2. Set pi32Y of Ltable to Table1 Y
3. Set Hydraulic analog with index M\_ONE to return value of LookupTableLookup with parameters (reference to Ltable, Dau\_eng\_data with index RHP\_SIG and reference to u8status)

When the following conditions are satisfied:

1. Dau\_eng\_data with index RHP\_SIG greater than or equal to M\_HYD\_PRESS\_LOW\_RED\_LIMIT
2. Dau\_eng\_data with index RHP\_SIG less than or equal to M\_HYD\_PRESS\_MIDDLE

##### 10.41.31.7.8 daugwylogicairbus-TxProcessLabel175HYDAng-LLR-008

Requirement ID: H398-LLD-GWY-FNC-6798

The function shall set the following:

1. Set pi32X of Ltable to Table2 X
2. Set pi32Y of Ltable to Table2 Y
3. Set Hydraulic analog with index M\_ONEto return value of LookupTableLookup with parameters (reference to Ltable, Dau\_eng\_data with index RHP\_SIG and reference to u8status)

When the following conditions are satisfied:

1. Dau\_eng\_data with index RHP\_SIG greater than M\_HYD\_PRESS\_MIDDLE
2. Dau\_eng\_data with index RHP\_SIG less than or equal to M\_HYD\_PRESS\_HIGH\_RED\_LIMIT

##### 10.41.31.7.9 daugwylogicairbus-TxProcessLabel175HYDAng-LLR-009

Requirement ID: H398-LLD-GWY-FNC-6799

The function shall set hydraulic analog with index ONE to M\_HYD\_PRESS\_TOP\_THIN\_RED\_LINE

When the following conditions are satisfied:

1. Dau\_eng\_data with index RHP\_SIG greater than M\_HYD\_PRESS\_HIGH\_RED\_LIMIT AND with

(Dau\_eng\_data with index RHP\_SIG less than or equal to M\_TWO\_HUNDREAD\_FIFTY

##### 10.41.31.7.10 daugwylogicairbus-TxProcessLabel175HYDAng-LLR-010

Requirement ID: H398-LLD-GWY-FNC-6800

The function shall perorm the following:

1. Set data package with index M\_ONE to Btmthinredline
2. Update data\_package with index M\_ONE to bitwise OR of data\_package with index M\_ONE and (Topthinredline left shift by M\_THREE)
3. Update data\_package with index ONE to bitwise OR of data\_package with index M\_ONE and call to CalculateEmbeddedResolution with parameter (hydraulic analog with index ONE, LABEL\_175\_HYD\_ANG\_2) left shift by M\_SIX
4. Set data of Tx\_a429\_bnr with index LABEL\_175\_HYD\_ANG\_2 to data\_package with index M\_ONE
5. Set status of Tx\_a429\_bnr with index LABEL\_175\_HYD\_ANG\_2 to OK when Dau\_eng\_status with index RHP\_SIG is equal to OK, otherwise set status of Tx\_a429\_bnr with index LABEL\_175\_HYD\_ANG\_2 to CHAN\_ERR.

### 10.41.32 TxProcessLabel176MGBOilPress

Low Level Design Details about CSU TxProcessLabel176MGBOilPress will follow in the sub sections.

#### 10.41.32.1 Brief Description

This function processes the MGB oil pressure data and prepares the data packet:

Label 176-01 - Engine 1 MGB Oil Pressure Digital Value Data Packet.

#### 10.41.32.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD(H398-003-012-GWY).

#### 10.41.32.3 List of global variables accessed and modified

Accessed : Dau\_eng\_data

Dau\_eng\_status

Modified : Tx\_a429\_bnr

#### 10.41.32.4 Parameter list (Input/Output)

None

#### 10.41.32.5 Return Value

None

#### 10.41.32.6 Other CSUs called by this CSU

None

#### 10.41.32.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TxProcessLabel176MGBOilPress

##### 10.41.32.7.1 daugwylogicairbus-TxProcessLabel176MGBOilPress-LLR-001

Requirement ID: H398-LLD-GWY-FNC-6809

The function shall perform the following

1. Set id of Tx\_a429\_bnr with index LABEL\_176\_MGB\_OIL\_PRESS\_1 is set to M\_ONE.
2. Set mgb\_oil \_pressure\_sst to DIGIT\_RED\_COLOR when Dau\_eng\_data with index MGBOP\_SIG is less than M\_MGB\_OIL\_PRESS\_LOW\_RED\_LIM\_DIG, otherwise set to DIGIT\_NOMINAL.
3. Set datapackage to BITWISE OR of datapackage and (mgb\_oil \_pressure\_sst LEFTSHIFT by M\_ONE).
4. Update datapackage with BITWISE OR of datapackage and (numstatusblink LEFTSHIFT by M\_THREE).
5. Update datapackage with BITWISE OR of datapackage and (return value from function CalculateEmbeddedResolution with parameters (Dau\_eng\_data with index MGBOP\_SIG and LABEL\_176\_MGB\_OIL\_PRESS\_1) LEFTSHIFT by M\_SIX).
6. Set data of Tx\_a429\_bnr with index LABEL\_176\_MGB\_OIL\_PRESS\_1 is set to datapackage.
7. Set status of Tx\_a429\_bnr with index LABEL\_176\_MGB\_OIL\_PRESS\_1 to OK when Dau\_eng\_status with index MGBOP\_SIG is equal to OK,otherwise set status of Tx\_a429\_bnr with index LABEL\_176\_MGB\_OIL\_PRESS\_1 to CHAN\_ERR.

### 10.41.33 TxProcessLabel177MGBOilPrsAng

Low Level Design Details about CSU TxProcessLabel177MGBOilPrsAng will follow in the sub sections.

#### 10.41.33.1 Brief Description

This function processes the MGB oil pressure analog data for engine 1 and prepares the data packet:

Label 177-01 - MGB Oil Pressure Analog Value Data Packet

#### 10.41.33.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD(H398-003-012-GWY).

#### 10.41.33.3 List of global variables accessed and modified

Accessed : Dau\_eng\_data

Dau\_eng\_status

Modified : Tx\_a429\_bnr

#### 10.41.33.4 Parameter list (Input/Output)

None

#### 10.41.33.5 Return Value

None

#### 10.41.33.6 Other CSUs called by this CSU

LookupTableLookup

#### 10.41.33.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TxProcessLabel177MGBOilPrsAng

##### 10.41.33.7.1 daugwylogicairbus-TxProcessLabel177MGBOilPrsAng-LLR-001

Requirement ID: H398-LLD-GWY-FNC-6818

The function shall perform the following

1. Set pi32\_x of l table to tablex.
2. Set pi32\_y of l table to tabley..
3. Set id of Tx\_a429\_bnr with index LABEL\_177\_MGB\_OIL\_PRESS\_ANG\_1 is set to M\_ONE.
4. Set btmthinredline to TRUE.
5. Set MGBOP\_signal to the product of Dau\_eng\_status with index MGBOP\_SIG and M\_HUNDRED

##### 10.41.33.7.2 daugwylogicairbus-TxProcessLabel177MGBOilPrsAng-LLR-002

Requirement ID: H398-LLD-GWY-FNC-6819

The function shall perform the following:

1. Call LookupTableLookup with parameters (reference to ltable, mgbop signal and reference to u8status) and set the return value to mgboilpressanalog.
2. Set datapackage to btmthinredline.
3. Update datapackage with BITWISE OR of datapackage and (return value from function CalculateEmbeddedResolution with  
    parameters (mgb\_oilpress\_analog and LABEL\_177\_MGB\_OIL\_PRESS\_ANG\_1 LEFTSHIFT by M\_SIX).
4. Set data of Tx\_a429\_bnr with index LABEL\_177\_MGB\_OIL\_PRESS\_ANG\_1 is set to datapackage.
5. Set status of Tx\_a429\_bnr with index LABEL\_177\_MGB\_OIL\_PRESS\_ANG\_1 to OK when Dau\_eng\_status with index MGBOP\_SIG is equal to OK, otherwise set status of Tx\_a429\_bnr with index LABEL\_177\_MGB\_OIL\_PRESS\_ANG\_1 to CHAN\_ERR .

### 10.41.34 TxProcessLabel200MGBOilTemp

Low Level Design Details about CSU TxProcessLabel200MGBOilTemp will follow in the sub sections.

#### 10.41.34.1 Brief Description

This function processes the MGB oil temperature data for engine 1 and prepares the data packet:

Label 200-01 - MGB Oil Temperature Digital Value Data Packet

#### 10.41.34.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.41.34.3 List of global variables accessed and modified

Accessed : Dau\_eng\_data

Dau\_eng\_status

Modified : Tx\_a429\_bnr

#### 10.41.34.4 Parameter list (Input/Output)

None

#### 10.41.34.5 Return Value

None

#### 10.41.34.6 Other CSUs called by this CSU

None

#### 10.41.34.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TxProcessLabel200MGBOilTemp

##### 10.41.34.7.1 daugwylogicairbus-TxProcessLabel200MGBOilTemp-LLR-001

Requirement ID: H398-LLD-GWY-FNC-6828

The function shall perform the following

1. Set id of Tx\_a429\_bnr with index LABEL\_200\_MGB\_OIL\_TEMP\_1 is set to M\_ONE.
2. Set mgboiltemperaturesst to DIGIT\_RED\_COLOR when either of the following conditions is satisfied
3. Dau\_eng\_data with index MGBT is less than M\_MGB\_OIL\_TEMP\_LOW\_RED\_LIMIT
4. Dau\_eng\_data with index MGBT is greater than M\_MGB\_OIL\_TEMP\_HIGH\_RED\_LIMIT.

Otherwise set mgboiltemperaturesst to DIGIT\_NOMINAL

##### 10.41.34.7.2 daugwylogicairbus-TxProcessLabel200MGBOilTemp-LLR-002

Requirement ID: H398-LLD-GWY-FNC-6829

The function shall perform the following

1. Set datapackage to BITWISE OR of datapackage and (mgboiltemperaturesst LEFTSHIFT by M\_ONE).
2. Update datapackage with BITWISE OR of datapackage and (numstatusblink LEFTSHIFT by M\_THREE).
3. Update datapackage with BITWISE OR of datapackage and (return value from function CalculateEmbeddedResolution with parameters (Dau\_eng\_data with index MGBT and LABEL\_200\_MGB\_OIL\_TEMP\_1) LEFTSHIFT by M\_SIX).
4. Set data of Tx\_a429\_bnr with index LABEL\_200\_MGB\_OIL\_TEMP\_1is set to datapackage.
5. Set status of Tx\_a429\_bnr with index LABEL\_200\_MGB\_OIL\_TEMP\_1 to OK when Dau\_eng\_status with index MGBT is equal to OK, otherwise set status of Tx\_a429\_bnr with index LABEL\_200\_MGB\_OIL\_TEMP\_1 to CHAN\_ERR.

### 10.41.35 TxProcessLabel201MGBOilTempAng

Low Level Design Details about CSU TxProcessLabel201MGBOilTempAng will follow in the sub sections.

#### 10.41.35.1 Brief Description

This function sets the TxProcessLabel201MGBOilTempAngThis function processes the MGB oil temperature analog data for engine 1 and prepares the data packet: Label 201-01 - MGB Oil Temperature Analog Value Data Packet.

#### 10.41.35.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.41.35.3 List of global variables accessed and modified

Accessed : Dau\_eng\_data

Dau\_eng\_status

Modified : Tx\_a429\_bnr

#### 10.41.35.4 Parameter list (Input/Output)

None

#### 10.41.35.5 Return Value

None

#### 10.41.35.6 Other CSUs called by this CSU

LookupTableLookup

#### 10.41.35.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TxProcessLabel201MGBOilTempAng

##### 10.41.35.7.1 daugwylogicairbus-TxProcessLabel201MGBOilTempAng-LLR-001

Requirement ID: H398-LLD-GWY-FNC-6838

The function shall perform the following

1. Set pi32\_x of table to table x.
2. Set pi32\_x of table to table y.
3. Set id of Tx\_a429\_bnr with index LABEL\_201\_MGB\_OIL\_TEMP\_ANG\_1 is set to M\_ONE.
4. Set btmthinredline to TRUE.

##### 10.41.35.7.2 daugwylogicairbus-TxProcessLabel201MGBOilTempAng-LLR-002

Requirement ID: H398-LLD-GWY-FNC-6839

The function shall do the following:

1. set topthinredLine to TRUE.

##### 10.41.35.7.3 daugwylogicairbus-TxProcessLabel201MGBOilTempAng-LLR-003

Requirement ID: H398-LLD-GWY-FNC-6840

The function shall perform the following:

1. Call LookupTableLookup with parameters (reference to ltable, Dau\_eng\_data with index MGBT and reference to u8status) and set the return value to mgboiltempanalog.
2. Set datapackage to btmthinredline.
3. Update datapackage with BITWISE OR of datapackage and (topthinredLine LEFTSHIFT by M\_THREE).

d) Update datapackage with BITWISE OR of datapackage and (return value from function CalculateEmbeddedResolution with

parameters (mgboiltempanalog and LABEL\_201\_MGB\_OIL\_TEMP\_ANG\_1 LEFTSHIFT by M\_SIX).

e) Set data of Tx\_a429\_bnr with index LABEL\_201\_MGB\_OIL\_TEMP\_ANG\_1 is set to datapackage.

f) Set status of Tx\_a429\_bnr with index LABEL\_201\_MGB\_OIL\_TEMP\_ANG\_1 to OK when Dau\_eng\_status with index of MGBT is equal to OK, otherwise set status of Tx\_a429\_bnr with index LABEL\_201\_MGB\_OIL\_TEMP\_ANG\_1 to CHAN\_ERR.

### 10.41.36 TxProcessLabel262FuelPress

Low Level Design Details about CSU TxProcessLabel262FuelPress will follow in the sub sections.

#### 10.41.36.1 Brief Description

This function processes the fuel pressure data for engines 1 and 2 and prepares two data packets:

Label 262-01 - Engine 1 Fuel Pressure Digital Value Data Packet and Label 262-10 - Engine 2 Fuel Pressure Digital Value Data Packet.

#### 10.41.36.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD(H398-003-012-GWY).

#### 10.41.36.3 List of global variables accessed and modified

Accessed : Dau\_eng\_data

Dau\_eng\_status

Modified : Tx\_a429\_bnr

#### 10.41.36.4 Parameter list (Input/Output)

None

#### 10.41.36.5 Return Value

None

#### 10.41.36.6 Other CSUs called by this CSU

None

#### 10.41.36.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TxProcessLabel262FuelPress.

##### 10.41.36.7.1 daugwylogicairbus-TxProcessLabel262FuelPress-LLR-001

Requirement ID: H398-LLD-GWY-FNC-6849

The function shall perform the following

1. Set id of Tx\_a429\_bnr with index LABEL\_262\_FUEL\_PRESS\_1 is set to M\_ONE.
2. Set fuelpressuresst to DIGIT\_RED\_COLOR when either of the following conditions is satisfied
3. Dau\_eng\_data with index FP1\_SIG is less than M\_FUEL\_PRESS\_LOW\_RED\_LIMIT.
4. Dau\_eng\_data with index FP1\_SIG is greater than M\_FUEL\_PRESS\_HIGH\_RED\_LIMIT.
5. Otherwise set fuelpressuresst to DIGIT\_NOMINAL.

c) set datapackage with index M\_ZERO to BITWISE OR of datapackage with index M\_ZERO and (fuelpressuresst LEFTSHIFT by M\_ONE).

d) Update datapackage with index M\_ZERO, with BITWISE OR of datapackage with index M\_ZERO and (numstatusblink LEFTSHFT by M\_THREE).

e) Update datapackage with index M\_ZERO, with BITWISE OR of datapackage with index M\_ZERO and (return value from function

CalculateEmbeddedResolution with parameters (Dau\_eng\_data with index FP1\_SIG and LABEL\_262\_FUEL\_PRESS\_1 LEFTSHIFT by M\_FOUR).

f) Set data of Tx\_a429\_bnr with index of LABEL\_262\_FUEL\_PRESS\_1 to datapackage with index M\_ZERO.

g) Set status of Tx\_a429\_bnr with LABEL\_262\_FUEL\_PRESS\_1 to OK when Dau\_eng\_status with index FP1\_SIG is equal to OK, otherwise set status of Tx\_a429\_bnr with LABEL\_262\_FUEL\_PRESS\_1 to CHAN\_ERR.

##### 10.41.36.7.2 daugwylogicairbus-TxProcessLabel262FuelPress-LLR-002

Requirement ID: H398-LLD-GWY-FNC-6850

The function shall perform the following

1. Set id of Tx\_a429\_bnr with index LABEL\_262\_FUEL\_PRESS\_2 is set to M\_TWO.
2. Set fuelpressuresst to DIGIT\_RED\_COLOR when either of the following conditions is satisfied
3. Dau\_eng\_data with index FP2\_SIG is less than M\_FUEL\_PRESS\_LOW\_RED\_LIMIT.
4. Dau\_eng\_data with index FP2\_SIG is greater than M\_FUEL\_PRESS\_HIGH\_RED\_LIMIT
5. Otherwise set fuelpressuresst to DIGIT\_NOMINAL.

c) Set datapackage with index M\_ONE to BITWISE OR of datapackage with index M\_ONE and (fuelpressuresst LEFTSHIFT by M\_ONE).

d) Update datapackage with index M\_ONE, with BITWISE OR of datapackage with index M\_ONE and (numstatusblink LEFTSHFT by M\_THREE).

e) Update datapackage with index M\_ONE, with BITWISE OR of datapackage with index M\_ONE and (return value from function CalculateEmbeddedResolution with parameters (Dau\_eng\_data with index FP2\_SIG and LABEL\_262\_FUEL\_PRESS\_2 LEFTSHIFT by M\_FOUR).

f) Set data of Tx\_a429\_bnr with index of LABEL\_262\_FUEL\_PRESS\_2 to datapackage with index M\_ONE.

g) Set status of Tx\_a429\_bnr with LABEL\_262\_FUEL\_PRESS\_2 to Dau\_eng\_status with index

FP2\_SIG is equal to OK, otherwise set status of Tx\_a429\_bnr with

LABEL\_262\_FUEL\_PRESS\_2 to CHAN\_ERR.

### 10.41.37 T41DisplayClass

Low Level Design Details about CSU T41DisplayClass will follow in the sub sections.

#### 10.41.37.1 Brief Description

This function sets the Engine 1 T4 display class limit based on Engine States Number and sets the Engine 1 T4 Limits status based on based on Engine 1 T4 Validity.

#### 10.41.37.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.41.37.3 List of global variables accessed and modified

Accessed                 : Composite\_engine\_state

Modified : displayClass\_T41

#### 10.41.37.4 Parameter list (Input/Output)

Inputs: None

Outputs: None

#### 10.41.37.5 Return Value

None

#### 10.41.37.6 Other CSUs called by this CSU

None

#### 10.41.37.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to T41DisplayClass

##### 10.41.37.7.1 daugwylogicairbus-T41DisplayClass-LLR-001

Requirement ID: H398-LLD-GWY-FNC-6860

The function shall determine displayClass\_T41 as follows:

When Composite\_engine\_state is set in the following format:

1. Set displayClass\_T41 to START\_CLASS when Composite\_engine\_state is set to ONE.
2. Set displayClass\_T41 to START\_CLASS when Composite\_engine\_state is set to TWO
3. Set displayClass\_T41 to START\_CLASS when Composite\_engine\_state is set to THREE
4. Set displayClass\_T41 to START\_CLASS when Composite\_engine\_state is set to FOUR
5. Set displayClass\_T41 to OEI when Composite\_engine\_state is set to FIVE.
6. Set displayClass\_T41 to OEI when Composite\_engine\_state is set to SEVEN.
7. Set displayClass\_T41 to OEI when Composite\_engine\_state is set to EIGHT.
8. Set displayClass\_T41 to OEI when Composite\_engine\_state is set to TWELVE.
9. Set displayClass\_T41 to OEI\_AND\_AEO When Composite\_engine\_state is set to NINE.
10. Set displayClass\_T41 to OEI\_AND\_AEO When Composite\_engine\_state is set to TEN.
11. Set displayClass\_T41 to OEI\_AND\_AEO When Composite\_engine\_state is set to ELEVEN.
12. Set displayClass\_T41 to OEI\_AND\_AEO When Composite\_engine\_state is set to THIRTEEN.
13. Set displayClass\_T41 to OEI\_AND\_AEO When Composite\_engine\_state is set to FOURTEEN.
14. Set displayClass\_T41 to OEI\_AND\_AEO When Composite\_engine\_state is set to FIFTEEN.
15. Set displayClass\_T41 to OEI\_AND\_AEO When Composite\_engine\_state is set to SIXTEEN.
16. Set displayClass\_T41 to OEI\_AND\_AEO When Composite\_engine\_state is set to SEVENTEEN.
17. Set displayClass\_T41 to OEI\_AND\_AEO When Composite\_engine\_state is set to EIGHTEEN.
18. Set displayClass\_T41 to AEO When Composite\_engine\_state is set to SIX.
19. When Composite\_engine\_state is set to other than ONE, TWO, THREE, FOUR, FIVE, SEVEN, EIGHT, TWELVE, NINE, TEN, ELEVEN, THIRTEEN, FOURTEEN, SIXTEEN and SIX then

displayClass\_T41 remains unchanged.

##### 10.41.37.7.2 daugwylogicairbus-T41DisplayClass-LLR-002

Requirement ID: H398-LLD-GWY-FNC-6862

The function shall perform the following:

* Set T41limitstatus to TRUE when T4validity of index ZERO is equal to OK and otherwise Set T41limitstatus to FALSE.

##### 10.41.37.7.3 daugwylogicairbus-T41DisplayClass-LLR-003

Requirement ID: H398-LLD-GWY-FNC-6863

The function shall perform the following:

1. Set operational Limits with index L270\_T4\_AEO\_YEL\_LO\_SDI01\_1 to FALSE
2. Set operational Limits with index L270\_T4\_AEO\_YEL\_UP\_SDI10\_1 to FALSE
3. Set operational Limits with index L271\_T4\_AEO\_YEL\_REC\_1 to FALSE
4. Set operational Limits with index L272\_T4\_AEO\_RED\_1 to FALSE
5. Set operational Limits with index L273\_T4\_AEO\_RED\_DIO\_1 to FALSE
6. Set operational Limits with index L274\_T4\_OEI\_YEL\_DASH\_1 to FALSE
7. Set operational Limits with index L275\_T4\_OEI\_RED\_DASH\_1 to FALSE
8. Set operational Limits with index L276\_T4\_OEI\_RED\_LINE\_1 to FALSE
9. Set operational Limits with index L300\_T4\_START\_WHI\_TRI\_1 to FALSE
10. Set operational Limits with index L301\_T4\_START\_RED\_TRI\_LOW\_1 to FALSE
11. Set operational Limits with index L302\_T4\_START\_RED\_TRI\_HIGH\_1 to FALSE

##### 10.41.37.7.4 daugwylogicairbus-T41DisplayClass-LLR-004

Requirement ID: H398-LLD-GWY-FNC-6864

The function shall perform the following when displayClass\_T41 is equal to AEO or displayClass\_T41 is equal to OEI\_AND\_AEO

1. Set operational Limits with index L270\_T4\_AEO\_YEL\_LO\_SDI01\_1 to BITWISE AND of TRUE and T41LimitStatus

b) Set operational Limits with index L270\_T4\_AEO\_YEL\_UP\_SDI10\_1 to BITWISE AND of TRUE and T41LimitStatus

c) Set operational Limits with index L272\_T4\_AEO\_RED\_1 to BITWISE AND of TRUE and T41LimitStatus

##### 10.41.37.7.5 daugwylogicairbus-T41DisplayClass-LLR-005

Requirement ID: H398-LLD-GWY-FNC-6865

The function shall perform the following when displayClass\_T41 is equal to OEI or displayClass\_T41 is equal to OEI\_AND\_AEO :

1. Set operational Limits with index L274\_T4\_OEI\_YEL\_DASH\_1 to BITWISE AND of TRUE and T41LimitStatus
2. Set operational Limits with index L275\_T4\_OEI\_RED\_DASH\_1 BITWISE AND of TRUE and T41LimitStatus

##### 10.41.37.7.6 daugwylogicairbus-T41DisplayClass-LLR-006

Requirement ID: H398-LLD-GWY-FNC-6866

The function shall perform the following when displayClass\_T41 is equal to START\_CLASS:

1. Set operational Limits with index L300\_T4\_START\_WHI\_TRI\_1 to BITWISE AND of TRUE and T41LimitStatus
2. Set operational Limits with index L302\_T4\_START\_RED\_TRI\_HIGH\_1 to BITWISE AND of TRUE and T41LimitStatus

### 10.41.38 T42DisplayClass

Low Level Design Details about CSU T42DisplayClass will follow in the sub sections.

#### 10.41.38.1 Brief Description

This function sets the Engine 2 T4 display class limit based on Engine States Number and sets the Engine 2 T4 Limits status based

on based on Engine 2 T4 Validity.

#### 10.41.38.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.41.38.3 List of global variables accessed and modified

Accessed : Composite\_engine\_state

Modified : displayClass\_T42

#### 10.41.38.4 Parameter list (Input/Output)

Inputs: None

Outputs: None

#### 10.41.38.5 Return Value

None

#### 10.41.38.6 Other CSUs called by this CSU

None

#### 10.41.38.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to T42DisplayClass

##### 10.41.38.7.1 daugwylogicairbus-T42DisplayClass-LLR-001

Requirement ID: H398-LLD-GWY-FNC-6876

The function shall determine displayClass\_T42 as follows:

When Composite\_engine\_state is set in the following format:

1. Set DisplayClass\_T42 to START\_CLASS when Composite\_engine\_state is set to ONE.
2. Set DisplayClass\_T42 to START\_CLASS when Composite\_engine\_state is set to TWELVE.
3. Set DisplayClass\_T42 to START\_CLASS when Composite\_engine\_state is set to THIRTEEN.
4. Set DisplayClass\_T42 to START\_CLASS when Composite\_engine\_state is set to FOURTEEN.
5. Set DisplayClass\_T42 to OEI when Composite\_engine\_state is set to TWO.
6. Set DisplayClass\_T42 to OEI when Composite\_engine\_state is set to FIFTEEN.
7. Set DisplayClass\_T42 to OEI when Composite\_engine\_state is set to SIXTEEN.
8. Set DisplayClass\_T42 to OEI when Composite\_engine\_state is set to SEVENTEEN .
9. Set DisplayClass\_T42 to OEI\_AND\_AEO when Composite\_engine\_state is set to THREE.
10. Set DisplayClass\_T42 to OEI\_AND\_AEO when Composite\_engine\_state is set to FOUR.
11. Set DisplayClass\_T42 to OEI\_AND\_AEO when Composite\_engine\_state is set to FIVE.
12. Set DisplayClass\_T42 to OEI\_AND\_AEO when Composite\_engine\_state is set to SEVEN.
13. Set DisplayClass\_T42 to OEI\_AND\_AEO when Composite\_engine\_state is set to EIGHT.
14. Set DisplayClass\_T42 to OEI\_AND\_AEO when Composite\_engine\_state is set to NINE.
15. Set DisplayClass\_T42 to OEI\_AND\_AEO when Composite\_engine\_state is set to TEN.
16. Set DisplayClass\_T42 to OEI\_AND\_AEO when Composite\_engine\_state is set to ELEVEN .
17. Set DisplayClass\_T42 to OEI\_AND\_AEO when Composite\_engine\_state is set to EIGHTEEN.
18. Set DisplayClass\_T42 to AEO when Composite\_engine\_state is set to SIX.
19. When Composite\_engine\_state is set to other than ONE, TWELVE, THIRTEEN, FOURTEEN, TWO, FIFTEEN, SIXTEEN,

SEVENTEEN, THREE, FOUR, FIVE, SEVEN, EIGHT, NINE, TEN, ELEVEN, EIGHTEEN and SIX then DisplayClass\_T42 remains

unchanged.

##### 10.41.38.7.2 daugwylogicairbus-T42DisplayClass-LLR-002

Requirement ID: H398-LLD-GWY-FNC-6877

The function shall perform the following:

* Set T42limitstatus to TRUE when T4validity of index ONE is equal to OK and otherwise Set T42limitstatus to FALSE.

##### 10.41.38.7.3 daugwylogicairbus-T42DisplayClass-LLR-003

Requirement ID: H398-LLD-GWY-FNC-6878

The function shall perform the following:

1. Set operational Limits with index L270\_T4\_AEO\_YEL\_LO\_SDI01\_2 to FALSE
2. Set operational Limits with index L270\_T4\_AEO\_YEL\_UP\_SDI10\_2 to FALSE
3. Set operational Limits with index L271\_T4\_AEO\_YEL\_REC\_2 to FALSE
4. Set operational Limits with index L272\_T4\_AEO\_RED\_2 to FALSE
5. Set operational Limits with index L273\_T4\_AEO\_RED\_DIO\_2 to FALSE
6. Set operational Limits with index L274\_T4\_OEI\_YEL\_DASH\_2 to FALSE
7. Set operational Limits with index L275\_T4\_OEI\_RED\_DASH\_2 to FALSE
8. Set operational Limits with index L276\_T4\_OEI\_RED\_LINE\_2 to FALSE
9. Set operational Limits with index L300\_T4\_START\_WHI\_TRI\_2 to FALSE
10. Set operational Limits with index L301\_T4\_START\_RED\_TRI\_LOW\_2 to FALSE
11. Set operational Limits with index L302\_T4\_START\_RED\_TRI\_HIGH\_2 to FALSE

##### 10.41.38.7.4 daugwylogicairbus-T42DisplayClass-LLR-004

Requirement ID: H398-LLD-GWY-FNC-6879

The function shall perform the following when displayClass\_T42 is equal to AEO or displayClass\_T42 is equal to OEI\_AND\_AEO

1. Set operational Limits with index L270\_T4\_AEO\_YEL\_LO\_SDI01\_2 to BITWISE AND of TRUE and T42LimitStatus
2. Set operational Limits with index L270\_T4\_AEO\_YEL\_UP\_SDI10\_2 to to BITWISE AND of TRUE and T42LimitStatus
3. Set operational Limits with index L272\_T4\_AEO\_RED\_2 to to BITWISE AND of TRUE and T42LimitStatus

##### 10.41.38.7.5 daugwylogicairbus-T42DisplayClass-LLR-005

Requirement ID: H398-LLD-GWY-FNC-6880

The function shall perform the following when displayClass\_T42 is equal to OEI or displayClass\_T42 is equal to OEI\_AND\_AEO :

1. Set operational Limits with index L274\_T4\_OEI\_YEL\_DASH\_2 to to BITWISE AND of TRUE and T42LimitStatus
2. Set operational Limits with index L275\_T4\_OEI\_RED\_DASH\_2 to BITWISE AND of TRUE and T42LimitStatus

##### 10.41.38.7.6 daugwylogicairbus-T42DisplayClass-LLR-006

Requirement ID: H398-LLD-GWY-FNC-6881

The function shall perform the following when displayClass\_T42 is equal to START\_CLASS:

1. Set operational Limits with index L300\_T4\_START\_WHI\_TRI\_2 to BITWISE AND of TRUE and T42LimitStatus
2. Set operational Limits with index L302\_T4\_START\_RED\_TRI\_HIGH\_2 to BITWISE AND of TRUE and T42LimitStatus

### 10.41.39 TxProcessLabel265T4

Low Level Design Details about CSU TxProcessLabel265T4 will follow in the sub sections.

#### 10.41.39.1 Brief Description

This function prepares two data packets:

Label 265-01 - Engine 1 T4 Digital Value Data Packet and

Label 265-10 - Engine 2 T4 Digital Value Data Packet.

#### 10.41.39.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.41.39.3 List of global variables accessed and modified

Accessed                 :  Engine\_oei\_state

          Composite\_engine\_state

          Dau\_eng\_data

          Dau\_eng\_status

Modified                   :  Tx\_a429\_bnr

  T4 rounded value

Dau\_eng\_data

#### 10.41.39.4 Parameter list (Input/Output)

None

#### 10.41.39.5 Return Value

None

#### 10.41.39.6 Other CSUs called by this CSU

T4Roundup

CalculateEmbeddedResolution

#### 10.41.39.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TxProcessLabel265T4

##### 10.41.39.7.1 daugwylogicairbus-TxProcessLabel265T4-LLR-001

Requirement ID: H398-LLD-GWY-FNC-6890

The function shall loop from count M\_ZERO until M\_MAX\_ENGINE minus one and perform the following:

1. Set display class t4 temp to Displayclass\_t41 when loop index s equal to M\_ZERO. Otherwise set display class t4 temp to Displayclass\_t42.
2. Set id of Tx\_a429\_bnr with index (LABEL\_265\_T4\_1 plus loop index) to (loop index plus M\_ONE).
3. Set data package with index of  loop index to ANALOG\_SRC.
4. Set T4 Validity with index of loop index to OK when (Dau\_eng\_status with index of (TOT1 plus loop index) is equal to OK AND Dau\_eng\_status with index of (TOT\_R1 plus loop index) is equal to OK) otherwise set T4 Validity with index of loop index to CHAN\_ERR.
5. Set totr\_corrective\_value to addition of (product of tot\_ corrective\_a and (product of Dau\_eng\_data with index of (TOT\_R1 plus loop index) and Dau\_eng\_data with index of (TOT\_R1 plus loop index)) and product of (tot\_ corrective\_b and Dau\_eng\_data with index of (TOT\_R1 plus loop index) ) and tot\_ corrective\_c) and set Dau\_eng\_data with index of (TOT1 plus loop index) to product of (Dau\_eng\_data with index of (TOT1 plus loop index) and totr\_corrective\_value) when T4 Validity with index of loop index is equal to OK otherwise set Dau\_eng\_data with index of (TOT1 plus loop index) to M\_ZERO.
6. Set T4 rounded value with index of loop index to return value of T4Roundup with parameters (Dau\_eng\_data with index of (TOT\_R1 plus loop index)).
7. Set numstatus to DIGIT\_NOMINAL.

##### 10.41.39.7.2 daugwylogicairbus-TxProcessLabel265T4-LLR-002

Requirement ID: H398-LLD-GWY-FNC-6891

The function shall loop from count M\_ZERO until M\_MAX\_ENGINE minus one and set numstatus to DIGIT\_AMBER\_COLOR when

1. ((ENG\_OEI\_State with index (M\_ENGINE\_1 plus loop index) of Engine\_oei\_state is equal to TRUE OR ENG\_OEI\_Training\_State with index (M\_ENGINE\_1 plus loop index) of Engine\_oei\_state is equal to TRUE) AND (T4 rounded value with index of loop index is greater than M\_T4\_OEI\_CT\_LIMIT AND less than or equal to M\_T4\_OEI\_LO\_LIMIT))     OR

((Composite\_engine\_state is equal to SIX OR OEI\_AND\_AEO is equal to display class t4 temp) AND (T4 rounded value with index of loop index is greater than M\_T4\_MCP\_LIMIT AND less than or equal to M\_T4\_TOP\_LIMIT)).

##### 10.41.39.7.3 daugwylogicairbus-TxProcessLabel265T4-LLR-003

Requirement ID: H398-LLD-GWY-FNC-6892

The function shall loop from count M\_ZERO until M\_MAX\_ENGINE minus one and set numstatus to DIGIT\_RED\_COLOR when

1. ((ENG\_OEI\_State with index (M\_ENGINE\_1 plus loop index) of Engine\_oei\_state is equal to TRUE OR ENG\_OEI\_Training\_State with index (M\_ENGINE\_1 plus loop index) of Engine\_oei\_state is equal to TRUE) AND (T4 rounded value with index of loop index is greater than M\_T4\_OEI\_LO\_LIMIT))  OR

((Composite\_engine\_state is equal to SIX OR OEI\_AND\_AEO is equal to display class t4 temp) AND T4 rounded value with index of loop index is greater than M\_T4\_TOP\_LIMIT)

OR

(Engine\_state\_logic with index of loop index is equal to START AND T4 rounded value with index of loop index is greater than M\_T4\_START\_RED\_TRI\_HIGH\_LIMIT)

Otherwise set numstatus to DIGIT\_NOMINAL.

##### 10.41.39.7.4 daugwylogicairbus-TxProcessLabel265T4-LLR-004

Requirement ID: H398-LLD-GWY-FNC-6893

The function shall loop from count M\_ZERO until M\_MAX\_ENGINE minus one and perform the following:

1. Update data package with loop index, with BITWISE OR of data package with loop index and (numstatus LEFTSHIFT by M\_ONE).
2. Update data package with loop index, with BITWISE OR of data package with loop index and (numstatusblink LEFTSHIFT by M\_THREE).
3. Update data package with loop index, with BITWISE OR of data package with loop index with (return value of CalculateEmbeddedResolution with parameters (T4 rounded value with index of loop index and LABEL\_265\_T4\_1 plus loop index) left shifted by M\_FOUR).
4. Set data of Tx\_a429\_bnr with index (LABEL\_265\_T4\_1 plus loop index) to datapackage with loop index
5. Set status of Tx\_a429\_bnr with index (LABEL\_265\_T4\_1 plus loop index) to T4 Validity with loop index.

### 10.41.40 TxProcessLabel266T4Angle

Low Level Design Details about CSU TxProcessLabel266T4Angle will follow in the sub sections.

#### 10.41.40.1 Brief Description

This function prepares two data packets:

Label 266-01 - Engine 1 T4 Pointer Angle Value Data Packet and

Label 266-10 - Engine 2 T4 Pointer Angle Value Data Packet.

#### 10.41.40.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.41.40.3 List of global variables accessed and modified

Accessed : Dau\_eng\_data

Modified : Tx\_a429\_bnr

#### 10.41.40.4 Parameter list (Input/Output)

None

#### 10.41.40.5 Return Value

None

#### 10.41.40.6 Other CSUs called by this CSU

LookupTableLookup

ConvertRange

CalculateEmbeddedResolution

#### 10.41.40.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TxProcessLabel266T4Angle

##### 10.41.40.7.1 daugwylogicairbus-TxProcessLabel266T4Angle-LLR-001

Requirement ID: H398-LLD-GWY-FNC-6904

The function shall loop from counter M\_ZERO until M\_MAX\_ENGINE minus one, and perform the following:

1. Set id of Tx\_a429\_bnr with index as (LABEL\_266\_T4\_ANGLE\_1 plus loop index) to M\_ONE plus counter.
2. Call LookupTableLookup with parameters (reference of ltable, Dau\_eng\_data with index as (TOT1 plus counter), and reference of u8status) and set the return value to t4angle.
3. Call convertRange wit parameter t4angle, and set the return value to t4angle\_In\_range.
4. Set datapackage with index as counter to (t4angle\_In\_range leftshift by M\_SEVEN).
5. Call CalculateEmbeddedResolution with parameters (datapackage with index as counter and (LABEL\_266\_T4\_ANGLE\_1 plus counter) and set the return value to data of Tx\_a429\_bnr with index as (LABEL\_266\_T4\_ANGLE\_1 plus counter)
6. Set status of Tx\_a429\_bnr with index as (LABEL\_266\_T4\_ANGLE\_1 plus counter) to T4 Validity with index of counter.

### 10.41.41 TxProcessLabel270T4AEOYellow

Low Level Design Details about CSU TxProcessLabel270T4AEOYellow will follow in the sub sections.

#### 10.41.41.1 Brief Description

This function prepares two data packets:

Label 270-01 - T4 AEO Yellow Value Data Packet and

Label 270-10 - T4 AEO Yellow Value Data Packet.

#### 10.41.41.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.41.41.3 List of global variables accessed and modified

Accessed                : None

Modified                  : Tx\_a429\_bnr

T4\_status

#### 10.41.41.4 Parameter list (Input/Output)

None

#### 10.41.41.5 Return Value

None

#### 10.41.41.6 Other CSUs called by this CSU

ConvertRange

CalculateEmbeddedResolution

#### 10.41.41.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TxProcessLabel270T4AEOYellow

##### 10.41.41.7.1 daugwylogicairbus-TxProcessLabel270T4AEOYellow-LLR-001

Requirement ID: H398-LLD-GWY-FNC-6913

The function shall Call convertRange with parameter M\_T4\_AEO\_YEL\_LO and set the return value to

t4aeoyelanglevalue with index as M\_ZERO.

##### 10.41.41.7.2 daugwylogicairbus-TxProcessLabel270T4AEOYellow-LLR-002

Requirement ID: H398-LLD-GWY-FNC-6914

The function shall Call convertRange with parameter M\_T4\_AEO\_YEL\_UP and set the return value to t4aeoyelanglevalue with index as M\_ONE.

##### 10.41.41.7.3 daugwylogicairbus-TxProcessLabel270T4AEOYellow-LLR-003

Requirement ID: H398-LLD-GWY-FNC-6915

The function shall set T4status to OK when

(T4validity with index as ZERO is equal to OK) OR (T4validity with index as ONE is equal to OK) , otherwise set T4status to CHAN\_ERR.

##### 10.41.41.7.4 daugwylogicairbus-TxProcessLabel270T4AEOYellow-LLR-004

Requirement ID: H398-LLD-GWY-FNC-6916

The Function shall loop from count M\_ZERO until M\_TWO minus one and perform the following:

1. Set id of Tx\_a429\_bnr with index as (LABEL\_270\_T4\_AEO\_YEL\_LO plus count) to count plus M\_ONE
2. Set datapackage with index as count to operation\_limit with index as (L270\_T4\_AEO\_YEL\_LO\_SDI01\_1 plus count).
3. Set datapackage with index as count to datapackage with index as count BITWISE OR (operation \_limit with index as (L270\_T4\_AEO\_YEL\_LO\_SDI01\_2 plus count) LEFTSHIFT by M\_ONE).
4. Update the datapackage with index as count with datapackage with index as count BITWISE OR with (return value from calling the function CalculateEmbeddedResolution with parameters (t4aeoyelanglevalue with index as count and LABEL\_270\_T4\_AEO\_YEL\_LO) LEFTSHIFT by M\_SEVEN)
5. Set data of Tx\_a429\_bnr with index as (LABEL\_270\_T4\_AEO\_YEL\_LO plus count) to datapackage with index as count.

##### 10.41.41.7.5 daugwylogicairbus-TxProcessLabel270T4AEOYellow-LLR-005

Requirement ID: H398-LLD-GWY-FNC-6917

The function shall set status of Tx\_a429\_bnr with index as (LABEL\_270\_T4\_AEO\_YEL\_LO plus count) to t4Status.

### 10.41.42 TxProcessLabel271T4AEOYelRec

Low Level Design Details about CSU TxProcessLabel271T4AEOYelRec will follow in the sub sections.

#### 10.41.42.1 Brief Description

This function prepares a data packet:

Label 271-00 - T4 AEO YELLOW Rectangle value Data Packet.

#### 10.41.42.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD(H398-003-012-GWY).

#### 10.41.42.3 List of global variables accessed and modified

Accessed : None

Modified : Tx\_a429\_bnr

#### 10.41.42.4 Parameter list (Input/Output)

None

#### 10.41.42.5 Return Value

None

#### 10.41.42.6 Other CSUs called by this CSU

None

#### 10.41.42.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TxProcessLabel271T4AEOYelRec

##### 10.41.42.7.1 daugwylogicairbus-TxProcessLabel271T4AEOYelRec-LLR-001

Requirement ID: H398-LLD-GWY-FNC-6926

The function shall perform the following:

1. Set id of Tx\_a429\_bnr with index as LABEL\_271\_T4\_AEO\_YEL\_REC to M\_ZERO.
2. Set datapackage to operation\_limit with index as L271\_T4\_AEO\_YEL\_REC\_1.
3. Update datapackage with datapackage BITWISE OR of (operation\_limit with index as (L271\_T4\_AEO\_YEL\_REC\_2) LEFT SHIFT by M\_ONE)
4. Update datapackage with datapackage BITWISE OR of (t4aeoyelrec\_value LEFT SHIFT by M\_SEVEN).
5. Set data of Tx\_a429\_bnr with index as LABEL\_271\_T4\_AEO\_YEL\_REC to return value by calling  
    CalculateEmbeddedResolution with parameters (datapackage and LABEL\_271\_T4\_AEO\_YEL\_REC)
6. Set status of Tx\_a429\_bnr with index as LABEL\_271\_T4\_AEO\_YEL\_REC to t4status.

### 10.41.43 TxProcessLabel272T4AEORed

Low Level Design Details about CSU TxProcessLabel272T4AEORed will follow in the sub sections.

#### 10.41.43.1 Brief Description

This function prepares a data packet:

Label 272-00 - T4 AEO Red value Data Packet.

#### 10.41.43.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.41.43.3 List of global variables accessed and modified

Accessed : None

Modified : Tx\_a429\_bnr

#### 10.41.43.4 Parameter list (Input/Output)

None

#### 10.41.43.5 Return Value

None

#### 10.41.43.6 Other CSUs called by this CSU

None

#### 10.41.43.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TxProcessLabel272T4AEORed

##### 10.41.43.7.1 daugwylogicairbus-TxProcessLabel272T4AEORed-LLR-001

Requirement ID: H398-LLD-GWY-FNC-6935

The function shall perform the following:

1. Call convertRange with parameter M\_T4\_AEO\_RED and set the return value to t4aeoredanglevalue.
2. Set id of Tx\_a429\_bnr with index as LABEL\_272\_T4\_AEO\_RED to M\_ZERO.
3. Set datapackage to operation\_limit with index as L272\_T4\_AEO\_RED\_1.
4. Update datapackage with datapackage BITWISE OR of (operation\_limit with index as (L272\_T4\_AEO\_RED\_2) LEFTSHIFT by M\_ONE)
5. Update datapackage with datapackage BITWISE OR (return value from calling   
   CalculateEmbeddedResolution with parameters (t4aeoredanglevalue and LABEL\_272\_T4\_AEO\_RED) LEFTSHIFT by M\_SEVEN).
6. Set data of Tx\_a429\_bnr with index as LABEL\_272\_T4\_AEO\_RED to datapackage.
7. Set status of Tx\_a429\_bnr with index as LABEL\_272\_T4\_AEO\_RED to t4status.

### 10.41.44 TxProcessLabel273T4AEORedDiode

Low Level Design Details about CSU TxProcessLabel273T4AEORedDiode will follow in the sub sections.

#### 10.41.44.1 Brief Description

This function prepares a data packet:

Label 273-00 - T4 AEO Red Diode value Data Packet.

#### 10.41.44.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.41.44.3 List of global variables accessed and modified

Accessed : None

Modified : Tx\_a429\_bnr

#### 10.41.44.4 Parameter list (Input/Output)

None

#### 10.41.44.5 Return Value

None

#### 10.41.44.6 Other CSUs called by this CSU

None

#### 10.41.44.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TxProcessLabel273T4AEORedDiode

##### 10.41.44.7.1 daugwylogicairbus-TxProcessLabel273T4AEORedDiode-LLR-001

Requirement ID: H398-LLD-GWY-FNC-6944

The function shall perform the following:

1. Set id of Tx\_a429\_bnr with index as LABEL\_273\_T4\_AEO\_RED\_DIO to M\_ZERO.
2. Set datapackage to operation\_limit with index as L273\_T4\_AEO\_RED\_DIO\_1.
3. Update datapackage with datapackage BITWISE OR of (operation\_limit with (index as L273\_T4\_AEO\_RED\_DIO\_2) LEFTSHIFT by M\_ONE).
4. Set t41aeoreddiosel to TRIANGLE.
5. Update datapackage with BITWISE OR of datapackage and (t41aeoreddiosel LEFTSHIFT by M\_TWO).
6. Set t42aeoreddiosel to TRIANGLE
7. Update datapackage with BITWISE OR of datapackage and (t42aeoreddioselLEFTSHIFT by M\_THREE).
8. Update datapackage with BITWISE OR of datapackage and (t4aeoreddiodevalue LEFTSHIFT by M\_SEVEN).
9. set data of Tx\_a429\_bnr with index as LABEL\_273\_T4\_AEO\_RED\_DIO to datapackage.
10. Set status of Tx\_a429\_bnr with index as LABEL\_273\_T4\_AEO\_RED\_DIO to t4status.

### 10.41.45 TxProcessLabel274T4OEIYelDash

Low Level Design Details about CSU TxProcessLabel274T4OEIYelDash will follow in the sub sections.

#### 10.41.45.1 Brief Description

This function prepares a data packet:

Label 274-00 - T4 OEI Yellow Dash value Data Packet.

#### 10.41.45.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD(H398-003-012-GWY).

#### 10.41.45.3 List of global variables accessed and modified

Accessed : None

Modified : Tx\_a429\_bnr

#### 10.41.45.4 Parameter list (Input/Output)

None

#### 10.41.45.5 Return Value

None

#### 10.41.45.6 Other CSUs called by this CSU

None

#### 10.41.45.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TxProcessLabel274T4OEIYelDash

##### 10.41.45.7.1 daugwylogicairbus-TxProcessLabel274T4OEIYelDash-LLR-001

Requirement ID: H398-LLD-GWY-FNC-6953

The function shall perform the following:

1. Call convertRange with parameter M\_T4\_OEI\_YEL\_DASH and set the return value to t4\_oeiyeldashvalue.
2. Set id of Tx\_a429\_bnr with index as LABEL\_274\_T4\_OEI\_YEL\_DASH to M\_ZERO.
3. Set datapackage to operation\_limit with index as L274\_T4\_OEI\_YEL\_DASH\_1.
4. Update datapackage with datapackage BITWISE OR of (operation\_limit with (index as L274\_T4\_OEI\_YEL\_DASH\_2) LEFT SHIFT by M\_ONE)
5. Update datapackage with datapackage BITWISE OR (return value from calling   
   CalculateEmbeddedResolution with parameters (t4\_oeiyeldashvalue and LABEL\_274\_T4\_OEI\_YEL\_DASH) LEFTSHIFT by M\_SEVEN).
6. Set data of Tx\_a429\_bnr with index as LABEL\_274\_T4\_OEI\_YEL\_DASH to datapackage.
7. Set status of Tx\_a429\_bnr with index as LABEL\_274\_T4\_OEI\_YEL\_DASH to t4status

### 10.41.46 TxProcessLabel275T4OEIRedDash

Low Level Design Details about CSU TxProcessLabel275T4OEIRedDash will follow in the sub sections.

#### 10.41.46.1 Brief Description

This function prepares a data packet:

Label 275-00 - T4 OEI Red Dash value Data Packet.

#### 10.41.46.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.41.46.3 List of global variables accessed and modified

Accessed : None

Modified : Tx\_a429\_bnr

#### 10.41.46.4 Parameter list (Input/Output)

None

#### 10.41.46.5 Return Value

None

#### 10.41.46.6 Other CSUs called by this CSU

None

#### 10.41.46.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TxProcessLabel275T4OEIRedDash

##### 10.41.46.7.1 daugwylogicairbus-TxProcessLabel275T4OEIRedDash-LLR-001

Requirement ID: H398-LLD-GWY-FNC-6962

The function shall perform the following:

1. Call convertRange with parameter M\_T4\_OEI\_RED\_DASH and set the return value to t4\_oeireddashvalue.
2. Set id of Tx\_a429\_bnr with index as LABEL\_275\_T4\_OEI\_RED\_DASH to M\_ZERO.
3. Set datapackage to operation\_limit with index as L275\_T4\_OEI\_RED\_DASH\_1.
4. Update datapackage with datapackage BITWISE OR of (operation\_limit with (index as L275\_T4\_OEI\_RED\_DASH\_2) LEFTSHIFT by M\_ONE)
5. Update datapackage with datapackage BITWISE OR (return value from calling   
   CalculateEmbeddedResolution with parameters (t4\_oeireddashvalue and LABEL\_275\_T4\_OEI\_RED\_DASH) LEFTSHIFT by M\_SEVEN).
6. Set data of Tx\_a429\_bnr with index as LABEL\_275\_T4\_OEI\_RED\_DASH to datapackage.
7. Set status of Tx\_a429\_bnr with index as LABEL\_275\_T4\_OEI\_RED\_DASH to t4status

### 10.41.47 TxProcessLabel276T4OEIRedLine

Low Level Design Details about CSU TxProcessLabel276\_T4\_OEI\_Red\_Line will follow in the sub sections.

#### 10.41.47.1 Brief Description

This function prepares a data packet:

Label 276-00 - T4 OEI Red Line value Data Packet.

#### 10.41.47.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.41.47.3 List of global variables accessed and modified

Accessed : None

Modified : Tx\_a429\_bnr

#### 10.41.47.4 Parameter list (Input/Output)

None

#### 10.41.47.5 Return Value

None

#### 10.41.47.6 Other CSUs called by this CSU

None

#### 10.41.47.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TxProcessLabel276T4OEIRedLine

##### 10.41.47.7.1 daugwylogicairbus-TxProcessLabel276T4OEIRedLine-LLR-001

Requirement ID: H398-LLD-GWY-FNC-6971

The function shall perform the following:

1. Set id of Tx\_a429\_bnr with index as LABEL\_276\_T4\_OEI\_RED\_LINE to M\_ZERO.
2. Set datapackage to operation\_limit with index as L276\_T4\_OEI\_RED\_LINE\_1.
3. Update datapackage with datapackage BITWISE OR of (operation\_limit with (index as L276\_T4\_OEI\_RED\_LINE\_2) LEFT SHIFT by M\_ONE)
4. Update datapackage with BITWISE OR of datapackage and (t4\_oeiredlinevalue LEFTSHFT by M\_SEVEN).

e) Update datapackage with datapackage BITWISE OR (return value from calling

CalculateEmbeddedResolution with parameters ((t4\_oei\_red\_line\_value and LABEL\_276\_T4\_OEI\_RED\_LINE) LEFT SHIFT by M\_SEVEN).

f) Set data of Tx\_a429\_bnr with index as LABEL\_276\_T4\_OEI\_RED\_LINE to datapackage.

g) Set status of Tx\_a429\_bnr with index as LABEL\_276\_T4\_OEI\_RED\_LINE to t4status

### 10.41.48 TxProcessLabel300T4STARTWhiTri

Low Level Design Details about CSU TxProcessLabel300T4STARTWhiTri will follow in the sub sections.

#### 10.41.48.1 Brief Description

This function prepares a data packet:

Label 300-00 - T4 START White Triangle value Data Packet.

#### 10.41.48.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.41.48.3 List of global variables accessed and modified

Accessed : None

Modified : Tx\_a429\_bnr

#### 10.41.48.4 Parameter list (Input/Output)

None

#### 10.41.48.5 Return Value

None

#### 10.41.48.6 Other CSUs called by this CSU

None

#### 10.41.48.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TxProcessLabel300T4STARTWhiTri

##### 10.41.48.7.1 daugwylogicairbus-TxProcessLabel300T4STARTWhiTri-LLR-001

Requirement ID: H398-LLD-GWY-FNC-6981

The function shall perform the following:

1. Call convertRange with parameter M\_T4\_START\_WHI\_TRI and set the return value to t4\_startwhitrivalue.
2. Set id of Tx\_a429\_bnr with index as LABEL\_300\_T4\_START\_WHI\_TRI to M\_ZERO.
3. Set datapackage to operation\_limit with index as L300\_T4\_START\_WHI\_TRI\_1.
4. Update datapackage with datapackage BITWISE OR of (operation\_limit with (index as L300\_T4\_START\_WHI\_TRI\_2) LEFTSHIFT by M\_ONE)
5. Update datapackage with datapackage BITWISE OR (return value from calling   
   CalculateEmbeddedResolution with parameters (t4\_startwhitrivalue and LABEL\_300\_T4\_START\_WHI\_TRI) LEFTSHIFT by M\_SEVEN).
6. Set data of Tx\_a429\_bnr with index as LABEL\_300\_T4\_START\_WHI\_TRI to datapackage.
7. Set status of Tx\_a429\_bnr with index as LABEL\_300\_T4\_START\_WHI\_TRI to t4status

### 10.41.49 TxProcessLabel301T4STRTRdTriLo

Low Level Design Details about CSU TxProcessLabel301T4STRTRdTriLo will follow in the sub sections.

#### 10.41.49.1 Brief Description

This function prepares a data packet:

Label 301-00 - T4 START Red Triangle Low value Data Packet.

#### 10.41.49.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.41.49.3 List of global variables accessed and modified

Accessed : None

Modified : Tx\_a429\_bnr

#### 10.41.49.4 Parameter list (Input/Output)

None

10.41.49.5 Return Value

None

#### 10.41.49.6 Other CSUs called by this CSU

None

#### 10.41.49.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TxProcessLabel301T4STRTRdTriLo

##### 10.41.49.7.1 daugwylogicairbus-TxProcessLabel301T4STRTRdTriLo-LLR-001

Requirement ID: H398-LLD-GWY-FNC-6990

The function shall perform the following:

1. Set id of Tx\_a429\_bnr with index as LABEL\_301\_T4\_START\_RED\_TRI\_LOW to M\_ZERO.
2. Set datapackage to operation\_limit with index as L301\_T4\_START\_RED\_TRI\_LOW\_1.
3. Update datapackage with datapackage BITWISE OR of (operation\_limit with (index as L301\_T4\_START\_RED\_TRI\_LOW\_2) LEFTSHIFT by M\_ONE)
4. Update datapackage with BITWISE OR of datapackage and (t4\_startredtrilowvalue LEFTSHIFT by M\_SEVEN).
5. Call CalculateEmbeddedResolution with parameters (datapackage and LABEL\_301\_T4\_START\_RED\_TRI\_LOW) and   
   set the return value to data of Tx\_a429\_bnr with index as LABEL\_301\_T4\_START\_RED\_TRI\_LOW.
6. Set status of Tx\_a429\_bnr with index as LABEL\_301\_T4\_START\_RED\_TRI\_LOW to t4status

### 10.41.50 TxProcessLabel302T4STARTRedTriHig

Low Level Design Details about CSU TxProcessLabel302T4STRTRdTriHi will follow in the sub sections.

#### 10.41.50.1 Brief Description

This function prepares a data packet:

Label 302-00 - T4 START Red Triangle High value Data Packet.

#### 10.41.50.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.41.50.3 List of global variables accessed and modified

Accessed : None

Modified : Tx\_a429\_bnr

#### 10.41.50.4 Parameter list (Input/Output)

None

#### 10.41.50.5 Return Value

None

#### 10.41.50.6 Other CSUs called by this CSU

None

#### 10.41.50.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TxProcessLabel302T4STRTRdTriHi

##### 10.41.50.7.1 daugwylogicairbus-TxProcessLabel302T4STRTRdTriHi-LLR-001

Requirement ID: H398-LLD-GWY-FNC-6999

The function shall perform the following:

1. Call convertRange with parameter M\_T4\_START\_RED\_TRI\_HIGH and set the return value to t4\_startredtrihighvalue.
2. Set id of Tx\_a429\_bnr with index as LABEL\_302\_T4\_START\_RED\_TRI\_HIGH to M\_ZERO.
3. Set datapackage to operation\_limit with index as L302\_T4\_START\_RED\_TRI\_HIGH\_1.
4. Update datapackage with datapackage BITWISE OR of (operation\_limit with (index as L302\_T4\_START\_RED\_TRI\_HIGH\_2) LEFTSHIFT by M\_ONE)
5. Update datapackage with datapackage BITWISE OR (return value from   
   calling CalculateEmbeddedResolution with parameters (t4\_startredtrihighvalue and LABEL\_275\_T4\_OEI\_RED\_DASH) LEFTSHIFT by M\_SEVEN).
6. Set data of Tx\_a429\_bnr with index as LABEL\_302\_T4\_START\_RED\_TRI\_HIGH to datapackage.
7. Set status of Tx\_a429\_bnr with index as LABEL\_302\_T4\_START\_RED\_TRI\_HIGH to t4status

### 10.41.51 TxProcessLabel316EOT

Low Level Design Details about CSU TxProcessLabel316EOT will follow in the sub sections.

#### 10.41.51.1 Brief Description

This function prepares two data packets:

Label 316-01- Engine 1 Engine Oil Temperature Digital Value Data Packet and

Label 316-10 - Engine 2 Engine Oil Temperature Digital Value Data Packet.

#### 10.41.51.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.41.51.3 List of global variables accessed and modified

Accessed : Dau\_eng\_data

Dau\_eng\_status

Modified : Tx\_a429\_bnr

#### 10.41.51.4 Parameter list (Input/Output)

None

#### 10.41.51.5 Return Value

None

#### 10.41.51.6 Other CSUs called by this CSU

None

#### 10.41.51.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TxProcessLabel316EOT

##### 10.41.51.7.1 daugwylogicairbus-TxProcessLabel316EOT-LLR-001

Requirement ID: H398-LLD-GWY-FNC-7008

The function shall set id of Tx\_a429\_bnr with index LABEL\_316\_EOT\_1 to M\_ONE

##### 10.41.51.7.2 daugwylogicairbus-TxProcessLabel316EOT-LLR-002

Requirement ID: H398-LLD-GWY-FNC-7009

The function shall set oiltemp to DIGIT\_RED\_COLOR when Dau\_eng\_data with index EOT1 is greater or equal to M\_EOT\_HIGH\_RED\_LIMIT, otherwise set to DIGIT\_NOMINAL.

##### 10.41.51.7.3 daugwylogicairbus-TxProcessLabel316EOT-LLR-003

Requirement ID: H398-LLD-GWY-FNC-7010

The function shall perform the following:

1. Set datapackage to BITWISE OR of datapackage and (oiltemp LEFT SHIFT by M\_ONE).
2. Set datapackage to BITWISE OR of datapackage and (numstatusblink LEFT SHIFT by M\_THREE).
3. Update datapackage with BITWISE OR of datapackage and (return value   
   from function CalculateEmbeddedResolution with parameters (Dau\_eng\_data with index EOT1 and LABEL\_316\_EOT\_1) LEFT SHIFT by M\_SEVEN).
4. Set data of Tx\_a429\_bnr with index LABEL\_316\_EOT\_1 to datapackage.
5. Set status of Tx\_a429\_bnr with index LABEL\_316\_EOT\_1 to OK when Dau\_eng\_status with index EOT1 is equal to OK, otherwise set status of Tx\_a429\_bnr with index LABEL\_316\_EOT\_1 to CHAN\_ERR.

##### 10.41.51.7.4 daugwylogicairbus-TxProcessLabel316EOT-LLR-004

Requirement ID: H398-LLD-GWY-FNC-7012

The function shall set id of Tx\_a429\_bnr with index LABEL\_316\_EOT\_2 to M\_TWO.

##### 10.41.51.7.5 daugwylogicairbus-TxProcessLabel316EOT-LLR-005

Requirement ID: H398-LLD-GWY-FNC-7013

The function shall set oiltemp to DIGIT\_RED\_COLOR when Dau\_eng\_data with index EOT2 is greater or equal to M\_EOT\_HIGH\_RED\_LIMIT, otherwise set to DIGIT\_NOMINAL.

##### 10.41.51.7.6 daugwylogicairbus-TxProcessLabel316EOT-LLR-006

Requirement ID: H398-LLD-GWY-FNC-7014

The function shall perform the following:

1. Set datapackage to BITWISE OR of datapackage and (oilTemp LEFT SHIFT by M\_ONE).
2. Set datapackage to BITWISE OR of datapackage and (numstatusblink LEFT SHIFT by M\_THREE).
3. Update datapackage with BITWISE OR of datapackage and (return value from function   
   CalculateEmbeddedResolution with parameters (Dau\_eng\_data with index EOT2 and LABEL\_316\_EOT\_2) LEFTSHIFT by M\_SEVEN).
4. Set data of Tx\_a429\_bnr with index LABEL\_316\_EOT\_2 to datapackage.
5. Set status of Tx\_a429\_bnr with index LABEL\_316\_EOT\_2 to OK when Dau\_eng\_status with index EOT2 is equal to OK, otherwise set status of Tx\_a429\_bnr with index LABEL\_316\_EOT\_2 to CHAN\_ERR.

### 10.41.52 TxProcessLabel317EOTAng

Low Level Design Details about CSU TxProcessLabel317EOTAng will follow in the sub sections.

#### 10.41.52.1 Brief Description

This function prepares two data packets:

Label 317-01 - Engine 1 Engine Oil Temperature Analog Value Data Packet and

Label 317-10 - Engine 2 Engine Oil Temperature Analog Value Data Packet.

#### 10.41.52.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.41.52.3 List of global variables accessed and modified

Accessed : Dau\_eng\_data

Dau\_eng\_status

Modified : Tx\_a429\_bnr

#### 10.41.52.4 Parameter list (Input/Output)

None

#### 10.41.52.5 Return Value

None

#### 10.41.52.6 Other CSUs called by this CSU

LookupTableLookup

#### 10.41.52.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TxProcessLabel317EOTAng

##### 10.41.52.7.1 daugwylogicairbus-TxProcessLabel317EOTAng-LLR-001

Requirement ID: H398-LLD-GWY-FNC-7023

The function shall perform the following:

1. Set id of Tx\_a429\_bnr with index LABEL\_317\_EOT\_ANG\_1 to M\_ONE.
2. Call LookupTableLookup with parameters (reference of ltable, Dau\_eng\_data with index EOT1 and reference of u8status)   
   and set the return value to oiltempanalog with index M\_ZERO.
3. Set datapackage with index M\_ZERO to BITWISE OR of datapackage with index M\_ZERO and (topthinredlineLEFT SHIFT by M\_THREE).
4. Update datapackage with index M\_ZERO with BITWISE OR of datapackage with index M\_ZERO and   
   (return value from function CalculateEmbeddedResolution with parameters (oiltempanalog with index M\_ZERO and LABEL\_317\_EOT\_ANG\_1) LEFT SHIFT by M\_SIX).
5. Set data of Tx\_a429\_bnr with index LABEL\_317\_EOT\_ANG\_1 to datapackage with index M\_ZERO.
6. Set status of Tx\_a429\_bnr with index LABEL\_317\_EOT\_ANG\_1 to OK when Dau\_eng\_status with index EOT1 is equal to OK, otherwise set status of Tx\_a429\_bnr with index LABEL\_317\_EOT\_ANG\_1 to CHAN\_ERR.

##### 10.41.52.7.2 daugwylogicairbus-TxProcessLabel317EOTAng-LLR-002

Requirement ID: H398-LLD-GWY-FNC-7024

The function shall perform the following:

1. Set id of Tx\_a429\_bnr with index LABEL\_317\_EOT\_ANG\_2 to M\_TWO.
2. Call LookupTableLookup with parameters (reference of ltable, Dau\_eng\_data with index EOT2 and reference of u8status)   
   and set the return value to oiltempanalog with index M\_ONE.
3. Set datapackage with index M\_ONE to BITWISE OR of datapackage with index M\_ONE and (topthinredline LEFTSHIFT by M\_THREE).
4. Update datapackage with index M\_ONE with BITWISE OR of datapackage with index M\_ONE and  
    (return value from function CalculateEmbeddedResolution with parameters (oiltempanalog with index M\_ONE and LABEL\_317\_EOT\_ANG\_2) LEFTSHIFT by M\_SIX).
5. Set data of Tx\_a429\_bnr with index LABEL\_317\_EOT\_ANG\_2 to datapackage with index M\_ONE.
6. Set status of Tx\_a429\_bnr with index LABEL\_317\_EOT\_ANG\_2 to OK when Dau\_eng\_status with index EOT2 is equal to OK, otherwise set status of Tx\_a429\_bnr with index LABEL\_317\_EOT\_ANG\_2 to CHAN\_ERR.

### 10.41.53 TxProcessLabel320EOP

Low Level Design Details about CSU TxProcessLabel320EOP will follow in the sub sections.

#### 10.41.53.1 Brief Description

This function prepares two data packets:

Label 320-01 - Engine 1 Engine Oil Pressure Digital Value Data Packet and

Label 320-10 - Engine 2 Engine Oil Pressure Digital Value Data Packet.

#### 10.41.53.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.41.53.3 List of global variables accessed and modified

Accessed : Dau\_eng\_data

Dau\_eng\_status

Modified : Tx\_a429\_bnr

#### 10.41.53.4 Parameter list (Input/Output)

None

#### 10.41.53.5 Return Value

None

#### 10.41.53.6 Other CSUs called by this CSU

CalculateEmbeddedResolution

#### 10.41.53.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TxProcessLabel320EOP

##### 10.41.53.7.1 daugwylogicairbus-TxProcessLabel320EOP-LLR-001

Requirement ID: H398-LLD-GWY-FNC-7033

The function shall loop from count M\_ZERO until M\_MAX\_ENGINE minus one and perform the following:

1. Set id of Tx\_a429\_bnr with index as (LABEL\_320\_EOP\_1 plus count) to count plus M\_ONE.
2. Set oilpress to DIGIT\_RED\_COLOR when (Dau\_eng\_data with index (EOP1 plus count) is lessthan M\_EOP\_LOW\_RED\_LIMIT\_DIGITAL) OR (Dau\_eng\_data with index (EOP1 plus count) is greater or equal to M\_EOP\_HIGH\_RED\_LIMIT\_DIGITAL), otherwise set to DIGIT\_NOMINAL.
3. Set datapackage with index count to BITWISE OR of datapackage with index count and (oilpress LEFTSHIFT by M\_ONE).
4. Update datapackage with index count with BITWISE OR of datapackage with index count and (numstatusblink LEFTSHIFT by M\_THREE).
5. Update datapackage with index count with BITWISE OR of datapackage with index count and (return value from function CalculateEmbeddedResolution with parameters (Dau\_eng\_data with index (EOP1 plus count) and (LABEL\_320\_EOP\_1 plus count) LEFTSHIFT by M\_SIX)
6. Set data of Tx\_a429\_bnr with index (LABEL\_320\_EOP\_1 plus count) to datapackage with index count.
7. Set status of Tx\_a429\_bnr with index (LABEL\_320\_EOP\_1 plus count) to OK when Dau\_eng\_status with index (EOP1 plus count) is equal to OK, otherwise set status of Tx\_a429\_bnr with index (LABEL\_320\_EOP\_1 plus count) to CHAN\_ERR.

### 10.41.54 TxProcessLabel321EOPAng

Low Level Design Details about CSU TxProcessLabel321EOPAng will follow in the sub sections.

#### 10.41.54.1 Brief Description

This function prepares two data packets:

Label 321-01 - Engine 1 Engine Oil Pressure Analog Value Data Packet and

Label 321-10 - Engine 2 Engine Oil Pressure Analog Value Data Packet.

#### 10.41.54.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.41.54.3 List of global variables accessed and modified

Accessed : Dau\_eng\_data

Dau\_eng\_status

Modified : Tx\_a429\_bnr

#### 10.41.54.4 Parameter list (Input/Output)

None

#### 10.41.54.5 Return Value

None

#### 10.41.54.6 Other CSUs called by this CSU

LookupTableLookup

#### 10.41.54.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TxProcessLabel321EOPAng

##### 10.41.54.7.1 daugwylogicairbus-TxProcessLabel321EOPAng-LLR-001

Requirement ID: H398-LLD-GWY-FNC-7043

The function shall loop from count M\_ZERO until M\_MAX\_ENGINE minus one and perform the following:

1. Set id of Tx\_a429\_bnr with index as (LABEL\_321\_EOP\_ANG\_1 plus count) to count plus M\_ONE.
2. Set btmthinredline with index count to TRUE
3. Set topthinredline with index count to TRUE
4. Set EOP\_signa with index count to product of Dau\_eng\_data with index

(EOP1 plus count) and M\_HUNDREAD.

e) Call LookupTableLookup with parameters (reference of ltable, EOP\_signal with index count

and reference of u8status) and set the return value to oilpressanalog with index count.

f) Set datapackage with index count to btmthinredline with index count.

g) Update datapackage with index count with BITWISE OR of datapackage with index count and (topthinredline with index count LEFT SHIFT by M\_THREE).

h) Update datapackage with index count with BITWISE OR of datapackage with index count and (return value from

function CalculateEmbeddedResolution with parameters (oilpressanalog with index count and (LABEL\_321\_EOP\_ANG\_1 plus count) LEFTSHIFT by M\_SIX)

i) Set data of Tx\_a429\_bnr with index (LABEL\_321\_EOP\_ANG\_1 plus count) to datapackage with index count.

j) Set status of Tx\_a429\_bnr with index (LABEL\_321\_EOP\_ANG\_1 plus count) to OK when Dau\_eng\_status with index (EOP1 plus count) is equal to OK, otherwise set status of Tx\_a429\_bnr with index (LABEL\_321\_EOP\_ANG\_1 plus count) to CHAN\_ERR.

### 10.41.55 DeltaNG1DisplayClass

Low Level Design Details about CSU DeltaNG1DisplayClass will follow in the sub sections.

#### 10.41.55.1 Brief Description

This function sets the display class limit for Engine 1 DeltaNG based on the composite engine state and handles the visibility of different

limits related to DeltaNG

#### 10.41.55.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.41.55.3 List of global variables accessed and modified

Accessed : Dau\_eng\_status

Modified : Displayclass\_ng1

#### 10.41.55.4 Parameter list (Input/Output)

Inputs: None

Outputs: None

#### 10.41.55.5 Return Value

None

#### 10.41.55.6 Other CSUs called by this CSU

None

#### 10.41.55.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to DeltaNG1\_DisplayClass

##### 10.41.55.7.1 daugwylogicairbus-DeltaNG1DisplayClass-LLR-001

Requirement ID: H398-LLD-GWY-FNC-7052

The function shall determine displayclass\_ng1 as follows:

When Composite\_engine\_state is set in the following format:

1. Set Displayclass\_ng1 to NO\_LIMITATION When Composite\_engine\_state is ONE.
2. Set Displayclass\_ng1 to NO\_LIMITATION When Composite\_engine\_state is TWO.
3. Set Displayclass\_ng1 to NO\_LIMITATION When Composite\_engine\_state is THREE.
4. Set Displayclass\_ng1 to NO\_LIMITATION When Composite\_engine\_state is FOUR.
5. Set Displayclass\_ng1 to OEI When Composite\_engine\_state is FIVE.
6. Set Displayclass\_ng1 to OEI When Composite\_engine\_state is SEVEN.
7. Set Displayclass\_ng1 to OEI When Composite\_engine\_state is EIGHT.
8. Set Displayclass\_ng1 to OEI When Composite\_engine\_state is TWELVE.
9. Set Displayclass\_ng1 to OEI\_AND\_AEO When Composite\_engine\_state is NINE.
10. Set Displayclass\_ng1 to OEI\_AND\_AEO When Composite\_engine\_state is TEN.
11. Set Displayclass\_ng1 to OEI\_AND\_AEO When Composite\_engine\_state is ELEVEN.
12. Set Displayclass\_ng1 to OEI\_AND\_AEO When Composite\_engine\_state is THIRTEEN.
13. Set Displayclass\_ng1 to OEI\_AND\_AEO When Composite\_engine\_state is FOURTEEN.
14. Set Displayclass\_ng1 to OEI\_AND\_AEO When Composite\_engine\_state is FIFTEEN.
15. Set Displayclass\_ng1 to OEI\_AND\_AEO When Composite\_engine\_state is SIXTEEN.
16. Set Displayclass\_ng1 to OEI\_AND\_AEO When Composite\_engine\_state is SEVENTEEN.
17. Set Displayclass\_ng1 to OEI\_AND\_AEO When Composite\_engine\_state is EIGHTEEN.
18. Set Displayclass\_ng1 to AEO When Composite\_engine\_state is SIX.
19. When Composite\_engine\_state is set to other than ONE, TWO, THREE, FOUR, FIVE, SEVEN, EIGHT, TWELVE, NINE, TEN,

ELEVEN, THIRTEEN, FOURTEEN, FIFTEEN, SIXTEEN, SEVENTEEN, EIGHTEEN and SIX then Displayclass\_ng1 remains

unchanged

##### 10.41.55.7.3 daugwylogicairbus-DeltaNG1DisplayClass-LLR-003

Requirement ID: H398-LLD-GWY-FNC-7054

The function shall perform the following:

* Set DeltaNG1LimitStatus to TRUE when Dau\_eng\_status of index DELTA\_NG1 is equal to OK otherwise Set DeltaNG1LimitStatus to FALSE

##### 10.41.55.7.4 daugwylogicairbus-DeltaNG1DisplayClass-LLR-004

Requirement ID: H398-LLD-GWY-FNC-7055

The function shall perform the following:

1. Set operational Limits with index L345\_DELTA\_NG\_AEO\_YEL\_LO\_SDI01\_1 to FALSE
2. Set operational Limits with index L345\_DELTA\_NG\_AEO\_YEL\_UP\_SDI10\_1 to FALSE
3. Set operational Limits with index L346\_DELTA\_NG\_AEO\_YEL\_REC\_1 to FALSE
4. Set operational Limits with index L347\_DELTA\_NG\_AEO\_RED\_1 to FALSE
5. Set operational Limits with index L350\_DELTA\_NG\_AEO\_RED\_DIO\_1 to FALSE
6. Set operational Limits with index L351\_DELTA\_NG\_OEI\_YEL\_DASH\_1 to FALSE
7. Set operational Limits with index L352\_DELTA\_NG\_OEI\_RED\_DASH\_1 to FALSE
8. Set operational Limits with index L353\_DELTA\_NG\_OEI\_RED\_LINE\_1 to FALSE

##### 10.41.55.7.5 daugwylogicairbus-DeltaNG1DisplayClass-LLR-005

Requirement ID: H398-LLD-GWY-FNC-7056

The function shall perform the following when Displayclass\_ng1 is equal to AEO:

1. Set operational Limits with index L345\_DELTA\_NG\_AEO\_YEL\_LO\_SDI01\_1 to BITWISE AND OF TRUE AND DeltaNG1LimitStatus
2. Set operational Limits with index L345\_DELTA\_NG\_AEO\_YEL\_UP\_SDI10\_1 to BITWISE AND OF TRUE AND DeltaNG1LimitStatus
3. Set operational Limits with index L347\_DELTA\_NG\_AEO\_RED\_1 to BITWISE AND OF TRUE AND DeltaNG1LimitStatus
4. Set operational Limits with index L350\_DELTA\_NG\_AEO\_RED\_DIO\_1 to BITWISE AND OF TRUE AND DeltaNG1LimitStatus when ENG\_PWR\_C\_Caution of Ext\_interfaces is equal to FALSE when Engine\_state\_logic with index of M\_ENGINE\_1 is not equal to STANDBY.

##### 10.41.55.7.6 daugwylogicairbus-DeltaNG1DisplayClass-LLR-006

Requirement ID: H398-LLD-GWY-FNC-7057

The function shall perform the following when Displayclass\_ng1 is equal to OEI:

a)Set operational Limits with index L351\_DELTA\_NG\_OEI\_YEL\_DASH\_1 to BITWISE AND OF TRUE AND DeltaNG1LimitStatus

b) Set operational Limits with index L352\_DELTA\_NG\_OEI\_RED\_DASH\_1 to BITWISE AND OF TRUE AND DeltaNG1LimitStatus

c)Set operational Limits with index L353\_DELTA\_NG\_OEI\_RED\_LINE\_1 to BITWISE AND OF TRUE AND DeltaNG1LimitStatus

##### 10.41.55.7.7 daugwylogicairbus-DeltaNG1DisplayClass-LLR-007

Requirement ID: H398-LLD-GWY-FNC-7058

The function shall perform the following when Displayclass\_ng1 is equal to OEI\_AND\_AEO:

a) Set operational Limits with index L345\_DELTA\_NG\_AEO\_YEL\_LO\_SDI01\_1 to BITWISE AND OF TRUE AND DeltaNG1LimitStatus

b) Set operational Limits with index L345\_DELTA\_NG\_AEO\_YEL\_UP\_SDI10\_1 to BITWISE AND OF TRUE AND DeltaNG1LimitStatus  
c) Set operational Limits with index L347\_DELTA\_NG\_AEO\_RED\_1 to BITWISE AND OF TRUE AND DeltaNG1LimitStatus

d)Set operational Limits with index L351\_DELTA\_NG\_OEI\_YEL\_DASH\_1 to BITWISE AND OF TRUE AND DeltaNG1LimitStatus  
e) Set operational Limits with index L352\_DELTA\_NG\_OEI\_RED\_DASH\_1 to BITWISE AND OF TRUE AND DeltaNG1LimitStatus

f) Set operational Limits with index L353\_DELTA\_NG\_OEI\_RED\_LINE\_1 to BITWISE AND OF TRUE AND DeltaNG1LimitStatus

### 10.41.56 DeltaNG1OEITopping

Low Level Design Details about CSU DeltaNG1OEITopping will follow in the sub sections.

#### 10.41.56.1 Brief Description

This function Sets the Engine 1 DeltaNG OEI Topping based on the current Displayclass\_ng1 value and the OEI2 and OEI30 discrete inputs and Delta NG1 LO and HI DIODE.

#### 10.41.56.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.41.56.3 List of global variables accessed and modified

Accessed : Xram\_data\_buffer

Displayclass\_ng1

Dau\_eng\_status

Modified : None

#### 10.41.56.4 Parameter list (Input/Output)

Inputs: None

Outputs: None

#### 10.41.56.5 Return Value

None

#### 10.41.56.6 Other CSUs called by this CSU

None

#### 10.41.56.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to DeltaNG1OEITopping

##### 10.41.56.7.1 daugwylogicairbus-DeltaNG1OEITopping-LLR-001

Requirement ID: H398-LLD-GWY-FNC-7067

The function shall does the following when

(Displayclass\_ng1 is equal to OEI) OR with (Displayclass\_ng1 is equal to OEI\_AND\_AEO) otherwise,

Set deltang1 oei hidiode and deltang1 oei lodiode to FALSE

1. Increment diode inactivetimer by one when logical NOT operation of disci with index DI\_2\_OEI30\_ENG1 of App A825 in Xram\_data\_buffer AND disci with index DI\_4\_OEI2\_ENG1 of App A825 in Xram\_data\_buffer otherwise set diode inactivetimer to M\_ZERO

2. Set prev deltang1 oei hidiode to deltang1 oei hidiode

3. Set prev deltang1 oei lodiode to deltang1 oei lodiode

4. Set deltang1 oei hidiode to TRUE and Set deltang1 oei lodiode to FALSE when

((disci of index DI\_2\_OEI30\_ENG1 of App A825 of Xram\_data\_buffer is TRUE) AND with NOT(disci of index DI\_4\_OEI2\_ENG1 of App A825 of Xram\_data\_buffer) is TRUE.

##### 10.41.56.7.2 daugwylogicairbus-DeltaNG1OEITopping-LLR-002

Requirement ID: H398-LLD-GWY-FNC-7068

The function shall perform the following when ((Displayclass\_ng1 is equal to OEI) OR with  (Displayclass\_ng1 is equal to OEI\_AND\_AEO )) :

1. Set deltang1 oei lodiode to TRUE

Set deltang1 oei hidiode to FALSE

When the following conditions are satisfied:

1. NOT of (disci of index DI\_2\_OEI30\_ENG1 of App\_A825 of Xram\_data\_buffer) is TRUE AND with NOT of (disci of index DI\_4\_OEI2\_ENG1 of App\_A825 of Xram\_data\_buffer) is TRUE .

##### 10.41.56.7.3 daugwylogicairbus-DeltaNG1OEITopping-LLR-003

Requirement ID: H398-LLD-GWY-FNC-7480

The function shall perform the following :

-Set Deltang oei lodiode with index M\_ENGINE\_1 to TRUE

-Set Deltang oei hidiode with index M\_ENGINE\_1 to FALSE

When the following conditions is satisfied:

deltang1 oei lodiode is equal to TRUE AND Dau\_eng\_status with index DELTA\_NG1 is equal to OK AND Opr\_limits with index L352\_DELTA\_NG\_OEI\_RED\_DASH\_1 to TRUE otherwise,

-Set Deltang oei lodiode with index M\_ENGINE\_1 to FALSE

-Set Deltang oei hidiode with index M\_ENGINE\_1 to FALSE

##### 10.41.56.7.4 daugwylogicairbus-DeltaNG1OEITopping-LLR-004

Requirement ID: H398-LLD-GWY-FNC-7489

The function shall perform the following When Displayclass\_ng1 is equal to OEI OR with Displayclass\_ng1 is equal to OEI\_AND\_AEO.

- Set diode inactivetimer to M\_300\_MILLISECONDS

- Set deltang1 oei hidiode to FALSE

- Set deltang1 oei lodiode to FALSE

When the following condition is satisfied:

diode inactivetimer is greater than or equal to M\_300\_MILLISECONDS otherwise,

-Set deltang1 oei hidiode to prev deltang1 oei\_hidiode

-Set deltang1 oei lodiode to prev deltang1 oei\_lodiode

##### 10.41.56.7.5 daugwylogicairbus-DeltaNG1OEITopping-LLR-005

Requirement ID: H398-LLD-GWY-FNC-7490

The function shall perform the following When Displayclass\_ng1 is equal to OEI OR with Displayclass\_ng1 is equal to OEI\_AND\_AEO.

- Set Deltang oei hidiode with index M\_ENGINE\_1 to TRUE

- Set Deltang oei lodiode with index M\_ENGINE\_1 to FALSE

When the following condition is satisfied:

deltang1 oei hidiode is equal to TRUE AND Dau\_eng\_status with index DELTA\_NG1 is equal to OK AND Opr\_limits with index L353\_DELTA\_NG\_OEI\_RED\_LINE\_1 to TRUE

### 10.41.57 DeltaNG2DisplayClass

Low Level Design Details about CSU DeltaNG2DisplayClass will follow in the sub sections.

#### 10.41.57.1 Brief Description

This function sets the Engine 2 DeltaNG display class limit based on the current Engine States Number and the OEI2 and OEI30 discrete inputs.

#### 10.41.57.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.41.57.3 List of global variables accessed and modified

Accessed : Dau\_eng\_status

Modified : Displayclass\_ng2

#### 10.41.57.4 Parameter list (Input/Output)

Inputs: None

Outputs: None

#### 10.41.57.5 Return Value

None

#### 10.41.57.6 Other CSUs called by this CSU

None

#### 10.41.57.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to DeltaNG2DisplayClass

##### 10.41.57.7.1 daugwylogicairbus-DeltaNG2DisplayClass-LLR-001

Requirement ID: H398-LLD-GWY-FNC-7078

The function shall determine Displayclass\_ng2 as follows:

When Composite\_engine\_state is set in the following format:

1. Set Displayclass\_ng2 to NO\_LIMITATION When Composite\_engine\_state is ONE.
2. Set Displayclass\_ng2 to NO\_LIMITATION When Composite\_engine\_state is TWELVE.
3. Set Displayclass\_ng2 to NO\_LIMITATION When Composite\_engine\_state is THIRTEEN .
4. Set Displayclass\_ng2 to NO\_LIMITATION When Composite\_engine\_state is FOURTEEN .
5. Set Displayclass\_ng2 to OEI When Composite\_engine\_state is TWO.
6. Set Displayclass\_ng2 to OEI When Composite\_engine\_state is FIFTEEN.
7. Set Displayclass\_ng2 to OEI When Composite\_engine\_state is SIXTEEN.
8. Set Displayclass\_ng2 to OEI When Composite\_engine\_state is SEVENTEEN .
9. Set Displayclass\_ng2 to OEI\_AND\_AEO When Composite\_engine\_state is THREE.
10. Set Displayclass\_ng2 to OEI\_AND\_AEO When Composite\_engine\_state is FOUR.
11. Set Displayclass\_ng2 to OEI\_AND\_AEO When Composite\_engine\_state is FIVE.
12. Set Displayclass\_ng2 to OEI\_AND\_AEO When Composite\_engine\_state is SEVEN
13. Set Displayclass\_ng2 to OEI\_AND\_AEO When Composite\_engine\_state is EIGHT.
14. Set Displayclass\_ng2 to OEI\_AND\_AEO When Composite\_engine\_state is NINE.
15. Set Displayclass\_ng2 to OEI\_AND\_AEO When Composite\_engine\_state is TEN.
16. Set Displayclass\_ng2 to OEI\_AND\_AEO When Composite\_engine\_state is ELEVEN.
17. Set Displayclass\_ng2 to OEI\_AND\_AEO When Composite\_engine\_state is EIGHTEEN.
18. Set Displayclass\_ng2 to AEO When Composite\_engine\_state is SIX.
19. When Composite\_engine\_state is set to other than ONE, TWELVE, THIRTEEN, FOURTEEN, TWO, FIFTEEN, SIXTEEN, SEVENTEEN, THREE, FOUR, FIVE, SEVEN,

             EIGHT, NINE, TEN, ELEVEN, EIGHTEEN and SIX then Displayclass\_ng2 remains unchanged

##### 10.41.57.7.3 daugwylogicairbus-DeltaNG2DisplayClass-LLR-003

Requirement ID: H398-LLD-GWY-FNC-7080

The function shall perform the following:

* Set DeltaNG2LimitStatus to TRUE When Dau\_eng\_status of index DELTA\_NG2 is OK otherwise set DeltaNG2LimitStatus to FALSE

##### 10.41.57.7.4 daugwylogicairbus-DeltaNG2DisplayClass-LLR-004

Requirement ID: H398-LLD-GWY-FNC-7081

The function shall perform the following:

1. Set operational Limits with index L345\_DELTA\_NG\_AEO\_YEL\_LO\_SDI01\_2 to FALSE
2. Set operational Limits with index L345\_DELTA\_NG\_AEO\_YEL\_UP\_SDI10\_2 to FALSE
3. Set operational Limits with index L346\_DELTA\_NG\_AEO\_YEL\_REC\_2 to FALSE
4. Set operational Limits with index L347\_DELTA\_NG\_AEO\_RED\_2 to FALSE
5. Set operational Limits with index L350\_DELTA\_NG\_AEO\_RED\_DIO\_2 to FALSE
6. Set operational Limits with index L351\_DELTA\_NG\_OEI\_YEL\_DASH\_2 to FALSE
7. Set operational Limits with index L352\_DELTA\_NG\_OEI\_RED\_DASH\_2 to FALSE

h) Set operational Limits with index L353\_DELTA\_NG\_OEI\_RED\_LINE\_2 to FALSE

##### 10.41.57.7.5 daugwylogicairbus-DeltaNG2DisplayClass-LLR-005

Requirement ID: H398-LLD-GWY-FNC-7082

The function shall perform the following when Displayclass\_ng2 is equal to AEO:

1. Set operational Limits with index L345\_DELTA\_NG\_AEO\_YEL\_LO\_SDI01\_2 to BITWISE AND of TRUE and DeltaNG2LimitStatus
2. Set operational Limits with index L345\_DELTA\_NG\_AEO\_YEL\_UP\_SDI10\_2 to BITWISE AND of TRUE and DeltaNG2LimitStatus
3. Set operational Limits with index L347\_DELTA\_NG\_AEO\_RED\_2 to BITWISE AND of TRUE and DeltaNG2LimitStatus
4. Set operational Limits with index L350\_DELTA\_NG\_AEO\_RED\_DIO\_2 to BITWISE AND of TRUE and DeltaNG2LimitStatus when ENG\_PWR\_C\_Caution of Ext interfaces is equal to FALSE and when Engine state logic with index of M\_ENGINE\_2 is not equal to STANDBY

##### 10.41.57.7.6 daugwylogicairbus-DeltaNG2DisplayClass-LLR-006

Requirement ID: H398-LLD-GWY-FNC-7083

The function shall perform the following when Displayclass\_ng2 is equal to OEI:

1. Set operational Limits with index L351\_DELTA\_NG\_OEI\_YEL\_DASH\_2 to BITWISE AND of TRUE and DeltaNG2LimitStatus
2. Set operational Limits with index L352\_DELTA\_NG\_OEI\_RED\_DASH\_2 to BITWISE AND of TRUE and DeltaNG2LimitStatus
3. Set operational Limits with index L353\_DELTA\_NG\_OEI\_RED\_LINE\_2 to BITWISE AND of TRUE and DeltaNG2LimitStatus

##### 10.41.57.7.7 daugwylogicairbus-DeltaNG2DisplayClass-LLR-007

Requirement ID: H398-LLD-GWY-FNC-7084

The function shall perform the following when Displayclass\_ng2 is equal to OEI\_AND\_AEO:

1. Set operational Limits with index L345\_DELTA\_NG\_AEO\_YEL\_LO\_SDI01\_2 to BITWISE AND of TRUE and DeltaNG2LimitStatus
2. Set operational Limits with index L345\_DELTA\_NG\_AEO\_YEL\_UP\_SDI10\_2 to BITWISE AND of TRUE and DeltaNG2LimitStatus
3. Set operational Limits with index L347\_DELTA\_NG\_AEO\_RED\_2 to BITWISE AND of TRUE and DeltaNG2LimitStatus
4. Set operational Limits with index L351\_DELTA\_NG\_OEI\_YEL\_DASH\_2 to BITWISE AND of TRUE and DeltaNG2LimitStatus
5. Set operational Limits with index L352\_DELTA\_NG\_OEI\_RED\_DASH\_2 to BITWISE AND of TRUE and DeltaNG2LimitStatus

g) Set operational Limits with index L353\_DELTA\_NG\_OEI\_RED\_LINE\_2 to BITWISE AND of TRUE and DeltaNG2LimitStatus

### 10.41.58 DeltaNG2OEITopping

Low Level Design Details about CSU DeltaNG2OEITopping will follow in the sub sections.

#### 10.41.58.1 Brief Description

This function sets the Engine 2 DeltaNG OEI Topping based on the current Displayclass\_ng2 value and the OEI2 and OEI30 discrete inputs.

#### 10.41.58.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.41.58.3 List of global variables accessed and modified

Accessed : Displayclass\_ng2

Xram\_data\_buffer

Modified : None

#### 10.41.58.4 Parameter list (Input/Output)

Inputs: None

Outputs: None

#### 10.41.58.5 Return Value

None

#### 10.41.58.6 Other CSUs called by this CSU

None

#### 10.41.58.7 Description of list of LLRs allocated

he following section will list the LLRs allocated to DeltaNG2OEITopping

##### 10.41.58.7.1 daugwylogicairbus-DeltaNG2OEITopping-LLR-001

Requirement ID: H398-LLD-GWY-FNC-7093

The function shall does the following when (Displayclass ng2 is equal to OEI) OR (Displayclass\_ng2 is equal to OEI\_AND\_AEO) returns TRUE otherwise,set deltang2 oei hidiode and deltang2 oei lodiode to FALSE

1. Increment diode inactivetimer by one when logical NOT operation of disci with index DI\_3\_OEI30\_ENG2 of App\_A825 in Xram\_data\_buffer AND logical NOT operation of disci with index DI\_5\_OEI2\_ENG2 of App A825 in Xram data buffer otherwise set diode inactivetimer to M\_ZERO

2. Set prev deltang2 oei hidiode to deltang2 oei hidiode

3. Set prev deltang2 oei lodiode to deltang2 oei Iodiode

4. a) Set deltang2 oei hidiode to TRUE

b) Set deltang2 oei lodiode to FALSE When

(disci of index DI\_3\_OEI30\_ENG2 of App A825 of Xram\_data\_buffer is TRUE ) AND NOT of (disci of index DI\_5\_OEI2\_ENG2 of App A825 of Xram\_data\_buffer) is TRUE

##### 10.41.58.7.2 daugwylogicairbus-DeltaNG2OEITopping-LLR-002

Requirement ID: H398-LLD-GWY-FNC-7094

The function shall does the following when (Displayclass\_ng2  is equal to OEI) OR (Displayclass\_ng2 is equal to OEI\_AND\_AEO) returns TRUE otherwise, set deltang2 oei hidiode and deltang2 oei lodiode to FALSE

1. Set deltang2 oei lodiode to TRUE
2. Set deltang2 oei hidiode to FALSE

When the following conditions are satisfied:

NOT of (disci of index DI\_3\_OEI30\_ENG2 of App\_A825 of Xram\_data\_buffer is TRUE) AND (disci of index DI\_5\_OEI2\_ENG2 of App A825 of  Xram\_data\_buffer) is TRUE

##### 10.41.58.7.3 daugwylogicairbus-DeltaNG2OEITopping-LLR-003

Requirement ID: H398-LLD-GWY-FNC-7095

The function shall does the following when (Displayclass ng2  is equal to OEI) OR (Displayclass\_ng2 is equal to OEI\_AND\_AEO) returns TRUE otherwise, set deltang2 oei hidiode and deltang2 oei lodiode to FALSE

a) Set diode inactivetimer to M\_300\_MILLISECONDS

b) Set deltang2 oei hidiode to FALSE

c) Set deltang2 oei lodiode to FALSE when diode inactivetimer is greater than or equal to M\_300\_MILLISECONDS, otherwise

a) deltang2 oei hidiode to prev deltang2 oei hidiode

b) deltang2 oei lodiode to prev deltang2 oei lodiode

##### 10.41.58.7.4 daugwylogicairbus-DeltaNG2OEITopping-LLR-004

Requirement ID: H398-LLD-GWY-FNC-7481

The function shall does the following:

1. a) Set Deltang oei hidiode with index M\_ENGINE\_2 to TRUE

b) Set Deltang oei lodiode with index M\_ENGINE\_2 to FALSE

when below condition is satisfied:

-deltang2 oei hidiode is equal to TRUE AND Dau\_eng\_status with index DELTA\_NG2 is equal to OK AND Opr\_limits with index L353\_DELTA\_NG\_OEI\_RED\_LINE\_2 is equal to TRUE

2. a) Set Deltang oei lodiode with index M\_ENGINE\_2 to TRUE

b) Set Deltang oei hidiode with index M\_ENGINE\_2 to FALSE

when below condition is satisfied:

- deltang2 oei lodiode is equal to TRUE AND Dau\_eng\_status with index DELTA\_NG2 is equal to OK AND Opr\_limits with index L353\_DELTA\_NG\_OEI\_RED\_DASH\_2 is equal to TRUE

otherwise,

a) Set Deltang oei hidiode with index M\_ENGINE\_2 to FALSE

b) Set Deltang oei lodiode with index M\_ENGINE\_2 to FALSE

### 10.41.59 TxProcessLabel343NG

Low Level Design Details about CSU TxProcessLabel343NG will follow in the sub sections.

#### 10.41.59.1 Brief Description

This function prepares two data packets:

Label 343-01 - Engine 1 Value Data Packet and

Label 343-10 - Engine 2 Value Data Packet.

#### 10.41.59.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.41.59.3 List of global variables accessed and modified

Accessed                 :  Engine\_oei\_state

          Composite\_engine\_state

         Dau\_eng\_data

         Dau\_eng\_status

Deltang\_sigvalue

Modified                   :  Tx\_a429\_bnr

#### 10.41.59.4 Parameter list (Input/Output)

None

#### 10.41.59.5 Return Value

None

#### 10.41.59.6 Other CSUs called by this CSU

CalculateEmbeddedResolution

#### 10.41.59.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TxProcessLabel343NG

##### 10.41.59.7.1 daugwylogicairbus-TxProcessLabel343NG-LLR-001

Requirement ID: H398-LLD-GWY-FNC-7105

The function shall loop from count M\_ZERO until M\_MAX\_ENGINE minus one and perform the following:

1. Set display class ng temp to Displayclass\_ng1 when loop index is equal to M\_ZERO otherwise set display class ng temp to Displayclass\_ng2
2. Set id of Tx\_a429\_bnr with index (LABEL\_343\_NG\_1 plus loop index) to (loop index plus M\_ONE).
3. Set datapackage to ANALOG\_SRC.
4. Set Deltang\_sigvalue with index loop index to the difference of ( the product of M\_FP\_SEVENTEEN\_P\_FIVE and ( the ratio of Dau\_eng\_data with index (DELTA\_NG1 plus loop index) to M\_HUNDRED)) and M\_FP\_NINE\_P\_SEVEN\_FIVE
5. Set numstatus to DIGIT\_NOMINAL

##### 10.41.59.7.2 daugwylogicairbus-TxProcessLabel343NG-LLR-002

Requirement ID: H398-LLD-GWY-FNC-7106

The function shall loop from count M\_ZERO until M\_MAX\_ENGINE minus one and set numstatus to DIGIT\_AMBER\_COLOR when

1. ((ENG\_OEI\_State with index (M\_ENGINE\_1 plus loop index) of Engine\_oei\_state is equal to TRUE OR ENG\_OEI\_Training\_State with index

(M\_ENGINE\_1 plus loop index) of Engine\_oei\_state is equal to TRUE) AND (Deltang\_sigvalue with index loop index is greater than M\_DELTA\_NG\_OEI\_CT\_LIMIT AND less than or equal to  M\_DELTA\_NG\_OEI\_LO\_LIMIT)) AND (ENG\_LO\_MSG with index of M\_ENGINE\_1 plus count of Ext\_interfaces is equal to FALSE) OR (Deltang oei lodiode with index of M\_ENGINE\_1 plus loop index is equal to TRUE) OR ((Composite\_engine\_state is equal to six OR OEI\_AND\_AEO is equal to display class ng temp)) AND (Deltang\_sigvalue with loop index is greater than M\_DELTA\_NG\_MCP\_LIMIT AND less than

M\_DELTA\_NG\_TOP\_LIMIT)) is TRUE.

##### 10.41.59.7.3 daugwylogicairbus-TxProcessLabel343NG-LLR-003

Requirement ID: H398-LLD-GWY-FNC-7107

The function shall loop from count M\_ZERO until M\_MAX\_ENGINE minus one and set numstatus to DIGIT\_RED\_COLOR when

1. ((ENG\_OEI\_State with index (M\_ENGINE\_1 plus count) of Engine\_oei\_state is equal to TRUE OR ENG\_OEI\_Training\_State with index (M\_ENGINE\_1 plus count) of Engine\_oei\_state is equal to TRUE) AND (Deltang\_sigvalue with index count is greater than M\_DELTA\_NG\_OEI\_LO\_LIMIT) AND ENG\_LO\_MSG with index of M\_ENGINE\_1 plus count of Ext\_interfaces is equal to FALSE)

OR

((Composite\_engine\_state is equal to SIX OR OEI\_AND\_AEO is equal to display class ng temp)) AND (Deltang\_sigvalue with index count is greater than M\_DELTA\_NG\_TOP\_LIMIT) is TRUE

##### 10.41.59.7.4 daugwylogicairbus-TxProcessLabel343NG-LLR-004

Requirement ID: H398-LLD-GWY-FNC-7108

The function shall loop from count M\_ZERO until M\_MAX\_ENGINE minus one and perform the following:

1. Update datapackage with BITWISE OR of datapackage and (numstatus LEFTSHIFT by M\_ONE).
2. set numstatusblink to NOT\_BLINKING.
3. Update datapackage with BITWISE OR of datapackage and (numstatusblink LEFTSHIFT by M\_THREE).
4. Set Dau\_eng\_data with index of (TACH\_NG1 plus count) to M\_ZERO when ((Dau\_eng\_data with index of (TACH\_NG1 plus count) less than variable ng forced to zero) AND (Engine\_state\_logic with index of (M\_ENGINE\_1 plus count) is equal to STANDBY) OR (Engine state logic with index of (M\_ENGINE\_1 plus count) is equal to OEI\_TI) OR (Engine state logic with index of (M\_ENGINE\_1 plus count) is equal to START)).
5. Update datapackage with BITWISE OR of datapackage and (return value from function CalculateEmbeddedResolution with parameters (Dau\_eng\_data with index (TACH\_NG1 plus count) and (LABEL\_343\_NG\_1 plus count) LEFTSHIFT by M\_FOUR)
6. Set data of Tx\_a429\_bnr with index (LABEL\_343\_NG\_1 plus count) to datapackage.
7. Set status of Tx a429 bnr with index (LABEL\_343\_NG\_1 plus count) to OK when Ng valid with index M\_ENGINE\_1 plus count is equal to OK otherwise set Tx a429 bnr with LABEL\_343\_NG\_1 plus count is equal to CHAN\_ERR

when Dau\_eng\_status with index (TACH\_NG1 plus count) is equal to OK, otherwise set status of Tx\_a429\_bnr with index (LABEL\_343\_NG\_1 plus count) to CHAN\_ERR.

### 10.41.60 TxProcessLabel344DeltaNGAngle

Low Level Design Details about CSU TxProcessLabel344DeltaNGAngle will follow in the sub sections.

#### 10.41.60.1 Brief Description

This function processes the Label 344-Delta NG Angle for both Engine 1 and Engine 2. It sets the ID, data, and status based on various conditions and engine states.

#### 10.41.60.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.41.60.3 List of global variables accessed and modified

Accessed                 :  Dau\_eng\_data

          Dau\_eng\_status

Deltang\_sigvalue

Modified                   :  Tx\_a429\_bnr

Deltang\_status

Deltang\_sigvalue

#### 10.41.60.4 Parameter list (Input/Output)

None

#### 10.41.60.5 Return Value

None

#### 10.41.60.6 Other CSUs called by this CSU

LookupTableLookup

CalculateEmbeddedResolution

#### 10.41.60.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TxProcessLabel344DeltaNGAngle

##### 10.41.60.7.2 daugwylogicairbus-TxProcessLabel344DeltaNGAngle-LLR-002

Requirement ID: H398-LLD-GWY-FNC-7120

The function performs the following:

1. Set pi32\_x of ltable to table x
2. Set pi32\_y of ltable to table y
3. Set DeltaNGStatus to OK when (Dau\_eng\_status of index DELTA\_NG1 is equal to OK) OR (Dau\_eng\_status of index DELTA\_NG2 is equal to OK))

otherwise set DeltaNGStatus to CHAN\_ERR

##### 10.41.60.7.3 daugwylogicairbus-TxProcessLabel344DeltaNGAngle-LLR-003

Requirement ID: H398-LLD-GWY-FNC-7121

The function shall loop from count M\_ZERO until M\_MAX\_ENGINE minus one and perform the following:

1. Set id of Tx\_a429\_bnr with index (LABEL\_344\_DELTA\_NG\_ANGLE\_1 plus count) to (count plus M\_ONE).
2. Set Deltang\_sigvalue of index count to  the difference of(product of    (M\_FP\_SEVENTEEN\_P\_FIVE and (fraction of Dau\_eng\_data of  index (DELTA\_NG1 plus count )and M\_HUNDRED)) minus M\_FP\_NINE\_P\_SEVEN\_FIVE))
3. Set Deltang\_sigvalue of index count to product of (Deltang\_sigvalue of index count and M\_HUNDRED).
4. Call LookupTableLookup with parameters (reference of ltable, Deltang\_sigvalue of index count and reference of u8status) and set the return value to delta\_ng\_angle.
5. Set datapackage of  index count to datapackage BITWISE OR ( by calling the function CalculateEmbeddedResolution    
   of parameters(delta\_ng\_angle of index count and LABEL\_344\_DELTA\_NG\_ANGLE\_1 plus loop index) LEFTSHIFT by M\_SEVEN).
6. Set data of Tx\_a429\_bnr with index (LABEL\_344\_DELTA\_NG\_ANGLE\_1 plus count) to datapackage of index count.
7. Set status of Tx\_a429\_bnr with index (LABEL\_344\_DELTA\_NG\_ANGLE\_1 plus count) to OK when Dau\_eng\_status with index (DELTA\_NG1 plus count) is equal to OK, otherwise set status of Tx\_a429\_bnr with index (LABEL\_344\_DELTA\_NG\_ANGLE\_1 plus count) to CHAN\_ERR.

### 10.41.61 TxProcessLabel345DltNGAEOYl

Low Level Design Details about CSU TxProcessLabel345DltNGAEOYl will follow in the sub sections.

#### 10.41.61.1 Brief Description

This function Transmits the DeltaNG AEO YELLOW values and the Engine 1 and Engine 2 DeltaNG operation limit - AEO Yellow zone visibility flags.

#### 10.41.61.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.41.61.3 List of global variables accessed and modified

Accessed : None

Modified : Tx\_a429\_bnr

#### 10.41.61.4 Parameter list (Input/Output)

None

#### 10.41.61.5 Return Value

None

#### 10.41.61.6 Other CSUs called by this CSU

None

#### 10.41.61.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TxProcessLabel345DltNGAEOYl

##### 10.41.61.7.1 daugwylogicairbus-TxProcessLabel345DltNGAEOYl-LLR-001

Requirement ID: H398-LLD-GWY-FNC-7130

The function shall call convertRange with parameter M\_DELTA\_NG\_AEO\_YEL\_LO and set the return value to deltangaeoyellowangle value with index M\_ZERO .

##### 10.41.61.7.2 daugwylogicairbus-TxProcessLabel345DltNGAEOYl-LLR-002

Requirement ID: H398-LLD-GWY-FNC-7131

The function shall call convertRange with parameter M\_DELTA\_NG\_AEO\_YEL\_UP and set the return value to deltangaeoyellowangle value with index M\_ONE .

##### 10.41.61.7.3 daugwylogicairbus-TxProcessLabel345DltNGAEOYl-LLR-003

Requirement ID: H398-LLD-GWY-FNC-7132

The function shall loop from count M\_ZERO until M\_TWO and perform the following:

1. Set id of Tx\_a429\_bnr with index (LABEL\_345\_DELTA\_NG\_AEO\_YEL\_LO plus count) to (M\_ONE plus count).
2. Set datapackage of size loop index to operation\_limit with index (L345\_DELTA\_NG\_AEO\_YEL\_LO\_SDI01\_1 plus count).
3. Update datapackage of size loop index with BITWISE OR of datapackage and (operation\_limit with index  
    (L345\_DELTA\_NG\_AEO\_YEL\_LO\_SDI01\_2 plus count) LEFTSHIFT by M\_ONE).
4. Update datapackage of size loop index with BITWISE OR of datapackage and   
   by calling function CalculateEmbeddedResolution with parameters(deltangaeoyellowangle value with index count and   
   LABEL\_345\_DELTA\_NG\_AEO\_YEL\_LO ) LEFTSHIFT by M\_SEVEN.
5. Set data of Tx\_a429\_bnr with index (LABEL\_345\_DELTA\_NG\_AEO\_YEL\_LO plus count) to datapackage of size loop index.
6. Set status of Tx\_a429\_bnr with index (LABEL\_345\_DELTA\_NG\_AEO\_YEL\_LO plus count) to

deltangstatus.

### 10.41.62 TxProcessLabel346DltNGAEOYlRc

Low Level Design Details about CSU TxProcessLabel346DltNGAEOYlRc will follow in the sub sections.

#### 10.41.62.1 Brief Description

This function sets the data, status, and ID for the Label 346-00 - DeltaNG AEO YELLOW Rectangle value Data Packet A429 message.

#### 10.41.62.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.41.62.3 List of global variables accessed and modified

Accessed : None

Modified : Tx\_a429\_bnr

#### 10.41.62.4 Parameter list (Input/Output)

None

#### 10.41.62.5 Return Value

None

#### 10.41.62.6 Other CSUs called by this CSU

None

#### 10.41.62.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TxProcessLabel346DltNGAEOYlRc

##### 10.41.62.7.1 daugwylogicairbus-TxProcessLabel346DltNGAEOYlRc-LLR-001

Requirement ID: H398-LLD-GWY-FNC-7142

The function shall perform the following:

1. Set id of Tx\_a429\_bnr with index LABEL\_346\_DELTA\_NG\_AEO\_YEL\_REC to M\_ZERO.
2. Set datapackage to operation\_limit with index L346\_DELTA\_NG\_AEO\_YEL\_REC\_1.
3. Update datapackage with BITWISE OR of datapackage and (operation\_limit with index (L346\_DELTA\_NG\_AEO\_YEL\_REC\_2) LEFTSHIFT by M\_ONE.
4. Update datapackage with BITWISE OR of datapackage and (return value from function   
   CalculateEmbeddedResolution with parameters (deltangaeoyellowrecvalue and LABEL\_346\_DELTA\_NG\_AEO\_YEL\_REC) LEFTSHIFT by M\_SEVEN).
5. Set data of Tx\_a429\_bnr with index LABEL\_346\_DELTA\_NG\_AEO\_YEL\_REC to datapackage.
6. Set status of Tx\_a429\_bnr with index LABEL\_346\_DELTA\_NG\_AEO\_YEL\_REC to deltangstatus.

### 10.41.63 TxProcessLabel347DeltaNGAEORed

Low Level Design Details about CSU TxProcessLabel347DeltaNGAEORed will follow in the sub sections.

#### 10.41.63.1 Brief Description

This function sets the data, status, and ID for the Label 347-00 - DeltaNG AEO Red value Data Packet A429 message.

#### 10.41.63.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD

#### 10.41.63.3 List of global variables accessed and modified

Accessed : None

Modified : Tx\_a429\_bnr

#### 10.41.63.4 Parameter list (Input/Output)

None

#### 10.41.63.5 Return Value

None

#### 10.41.63.6 Other CSUs called by this CSU

None

#### 10.41.63.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TxProcessLabel347DeltaNGAEORed

##### 10.41.63.7.1 daugwylogicairbus-TxProcessLabel347DeltaNGAEORed -LLR-001

Requirement ID: H398-LLD-GWY-FNC-7151

The function shall perform the following:

1. Call convertRange with parameter M\_DELTA\_NG\_AEO\_RED and set the return value to deltangaeoredangle value.
2. Set id of Tx\_a429\_bnr with index LABEL\_347\_DELTA\_NG\_AEO\_RED to M\_ZERO.
3. Set datapackage to operation\_limit with index L347\_DELTA\_NG\_AEO\_RED\_1.
4. Update datapackage with BITWISE OR of datapackage and (operation\_limit with index (L347\_DELTA\_NG\_AEO\_RED\_2) LEFTSHIFT by M\_ONE.
5. Update datapackage with BITWISE OR of datapackage and (return value from function  
    CalculateEmbeddedResolution with parameters deltangaeoredangle value and LABEL\_347\_DELTA\_NG\_AEO\_RED) LEFTSHIFT by M\_SEVEN).
6. Set data of Tx\_a429\_bnr with index LABEL\_347\_DELTA\_NG\_AEO\_RED to datapackage.
7. Set status of Tx\_a429\_bnr with index LABEL\_347\_DELTA\_NG\_AEO\_RED to deltangstatus.

### 10.41.64 TxProcessLabel350DltNGAEORdDi

Low Level Design Details about CSU TxProcssLabel350DeltaNGAEORedDiode will follow in the sub sections.

#### 10.41.64.1 Brief Description

This function sets the data, status, and ID for Label 350-00 - DeltaNG AEO Red Diode value Data Packet A429 message.

#### 10.41.64.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.41.64.3 List of global variables accessed and modified

Accessed: None

Modified: Tx\_a429\_bnr

#### 10.41.64.4 Parameter list (Input/Output)

None

#### 10.41.64.5 Return Value

None

#### 10.41.64.6 Other CSUs called by this CSU

None

#### 10.41.64.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TxProcessLabel350DltNGAEORdDi

##### 10.41.64.7.1 daugwylogicairbus-TxProcessLabel350DltNGAEORdDi-LLR-001

Requirement ID: H398-LLD-GWY-FNC-7161

The function shall perform the following:

i) Set delta Engine speed(NG) All Engine Operative Red Diode Value to return of convertRange with parameter M\_DELTA\_NG\_AEO\_RED\_DIO.

ii) Set id of Tx\_a429\_bnr of index LABEL\_350\_DELTA\_NG\_AEO\_RED\_DIO to M\_ZERO.

iii) Set data package to operation limits of index L350\_DELTA\_NG\_AEO\_RED\_DIO\_1.

iv) Set data package to Bitwise OR of data package and operation limits of index L350\_DELTA\_NG\_AEO\_RED\_DIO\_2 left shift M\_ONE.

v) Set delta Engine Speed1 All Engine Operative Red Diode Sel to DIODE.

vi) Set data package to data package bitwise OR of delta Engine Speed1 All Engine Operative Red Diode Sel left shift M\_TWO.

vii) Set delta Engine Speed2 All Engine Operative Red Diode Sel to DIODE.

viii) Set data package to data package bitwise OR of delta Engine Speed2 All Engine Operative Red Diode Sel left shift M\_THREE.

ix) Set data package to data package bitwise OR of return of CalculateEmbeddedResolution with

parameters delta Engine Speed All Engine Operative Red Diode Value and LABEL\_350\_DELTA\_NG\_AEO\_RED\_DIO left shift M\_SEVEN.

x) Set data of Tx\_a429\_bnr of index LABEL\_350\_DELTA\_NG\_AEO\_RED\_DIO to data package.

xi) Set status of Tx\_a429\_bnr of index LABEL\_350\_DELTA\_NG\_AEO\_RED\_DIO to Delta Engine Speed Status.

### 10.41.65 TxProcessLabel351DltNGOEIYlDsh

Low Level Design Details about CSU TxProcessLabel351DltNGOEIYlDsh will follow in the sub sections.

#### 10.41.65.1 Brief Description

This function sets the data, status, and ID for the Label 351-00 - DeltaNG OEI Yellow Dash value Data Packet a429 message.

#### 10.41.65.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.41.65.3 List of global variables accessed and modified

Accessed: None

Modified: Tx\_a429\_bnr

#### 10.41.65.4 Parameter list (Input/Output)

None

#### 10.41.65.5 Return Value

None

#### 10.41.65.6 Other CSUs called by this CSU

None

#### 10.41.65.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TxProcessLabel351DltNGOEIYlDsh

##### 10.41.65.7.1 daugwylogicairbus-TxProcessLabel351DltNGOEIYlDsh-LLR-001

Requirement ID: H398-LLD-GWY-FNC-7171

The function shall perform the following:

i) Set delta Engine Speed (NG) One Engine Inoperative Yellow Dash Value to return of convert Range

with parameter M\_DELTA\_NG\_OEI\_YEL\_DASH.

ii) Set id of transmitter a429 binary of index LABEL\_351\_DELTA\_NG\_OEI\_YEL\_DASH to M\_ZERO.

iii) Set data package to operation limits of index L351\_DELTA\_NG\_OEI\_YEL\_DASH\_1.

iv) Set data package to data package bitwise OR of operation limits of index L351\_DELTA\_NG\_OEI\_YEL\_DASH\_2 left shift M\_ONE.

v) Set data package to data package bitwise OR of return of Calculate Embedded Resolution with

parameters delta ENGINE SPEED One Engine Inoperative Yellow Dash Value and LABEL\_351\_DELTA\_NG\_OEI\_YEL\_DASH left shift M\_SEVEN.

vi) Set data of transmitter a429 binary of index LABEL\_351\_DELTA\_NG\_OEI\_YEL\_DASH to data package.

vii) Set status of transmitter a429 binary of index LABEL\_351\_DELTA\_NG\_OEI\_YEL\_DASH to Delta Engine Speed Status.

### 10.41.66 TxProcessLabel352DltNGOEIRdDsh

Low Level Design Details about CSU TxProcessLabel352DltNGOEIRdDshwill follow in the sub sections.

#### 10.41.66.1 Brief Description

This function sets the data, status, and ID for Label 352-00 - DeltaNG OEI Red Dash value Data Packet A429 message.

#### 10.41.66.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.41.66.3 List of global variables accessed and modified

Accessed: None

Modified: Tx\_a429\_bnr

#### 10.41.66.4 Parameter list (Input/Output)

None

#### 10.41.66.5 Return Value

None

#### 10.41.66.6 Other CSUs called by this CSU

None

#### 10.41.66.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TxProcessLabel352DltNGOEIRdDsh

##### 10.41.66.7.1 daugwylogicairbus-TxProcessLabel352DltNGOEIRdDsh-LLR-001

Requirement ID: H398-LLD-GWY-FNC-7180

The function shall perform the following:

i) Set delta Engine Speed (NG) One Engine Inoperative Red Dash Value to return of convert Range

with parameter M\_DELTA\_NG\_OEI\_RED\_DASH.

ii) Set id of transmitter a429 binary of index LABEL\_352\_DELTA\_NG\_OEI\_RED\_DASH to M\_ZERO.

iii) Set data package to operation limits of index L352\_DELTA\_NG\_OEI\_RED\_DASH\_1.

iv) Set data package to data package bitwise OR of operation limits of index L352\_DELTA\_NG\_OEI\_RED\_DASH\_2 left shift M\_ONE.

v) Set data package to data package bitwise OR of return of Calculate Embedded Resolution with parameters delta ENGINE

SPEED One Engine Inoperative Red Dash Value and LABEL\_352\_DELTA\_NG\_OEI\_RED\_DASH left shift M\_SEVEN.

vi) Set data of transmitter a429 binary of index LABEL\_352\_DELTA\_NG\_OEI\_RED\_DASH to data package.

vii) Set status of transmitter a429 binary of index LABEL\_352\_DELTA\_NG\_OEI\_RED\_DASH to Delta Engine Speed Status.

### 10.41.67 TxProcessLabel353DltNGOEIRdLn

Low Level Design Details about CSU TxProcessLabel353DltNGOEIRdLn will follow in the

sub sections.

#### 10.41.67.1 Brief Description

This function sets the data, status, and ID for the Label 353-00 - DeltaNG OEI Red Line value Data Packet A429 message.

#### 10.41.67.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.41.67.3 List of global variables accessed and modified

Accessed: None

Modified: Tx\_a429\_bnr

#### 10.41.67.4 Parameter list (Input/Output)

None

#### 10.41.67.5 Return Value

None

#### 10.41.67.6 Other CSUs called by this CSU

None

#### 10.41.67.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TxProcessLabel353DltNGOEIRdLn

##### 10.41.67.7.1 daugwylogicairbus-TxProcessLabel353DltNGOEIRdLn-LLR-001

Requirement ID: H398-LLD-GWY-FNC-7189

The function shall do the following:

i) Set delta Engine Speed (NG) One Engine Inoperative Red Line Value to return

of convert Range with parameter M\_DELTA\_NG\_OEI\_RED\_LINE.

ii) Set id of transmitter a429 binary of index LABEL\_353\_DELTA\_NG\_OEI\_RED\_LINE to M\_ZERO.

iii) Set data package to operation limits of index L353\_DELTA\_NG\_OEI\_RED\_LINE\_1.

iv) Set data package to data package bitwise OR of operation limits of index L353\_DELTA\_NG\_OEI\_RED\_LINE\_2left shift M\_ONE.

v) Set data package to data package bitwise OR of return of Calculate Embedded Resolution with parameters

delta ENGINE SPEED One Engine Inoperative Red Line Value and LABEL\_353\_DELTA\_NG\_OEI\_RED\_LINE left shift M\_SEVEN.

vi) Set data of transmitter a429 binary of index LABEL\_353\_DELTA\_NG\_OEI\_RED\_LINE to data package.

vii) Set status of transmitter a429 binary of index LABEL\_353\_DELTA\_NG\_OEI\_RED\_LINE to Delta Engine Speed Status.

### 10.41.68 TxProcessLabel012WarningStatus

Low Level Design Details about CSU TxProcessLabel012WarningStatus will follow in the sub sections.

#### 10.41.68.1 Brief Description

This function sets the data, status, and ID for the Label 012 Engine Status Data Packet A429 message.

#### 10.41.68.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.41.68.3 List of global variables accessed and modified

Accessed: Ext\_interfaces

Bleed\_value

Engine\_chip

Modified: Tx\_a429\_ddw

#### 10.41.68.4 Parameter list (Input/Output)

None

#### 10.41.68.5 Return Value

None

#### 10.41.68.6 Other CSUs called by this CSU

None

#### 10.41.68.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TxProcessLabel012WarningStatus

##### 10.41.68.7.1 daugwylogicairbus-TxProcessLabel012WarningStatus-LLR-001

Requirement ID: H398-LLD-GWY-FNC-7198

The function shall do the following.

1. Set id of Tx\_a429\_ddw of LABEL\_012\_ENGINE\_STATUS to M\_ZERO.
2. Set data package to M\_ZERO.
3. Set data package to data package BITWISE OR of M\_ZERO left shift M\_ONE.
4. Set data package to data package BITWISE OR of ENGINE 1 Chip Detector of Engine Chip left shift M\_TWO.
5. Set data package to data package BITWISE OR of ENG 2 Chip Detector of Engine Chip left shift M\_THREE.
6. Set data package to data package BITWISE OR of ENGINE THROTTLE Warning of EXTERNAL interfaces left shift M\_FOUR.
7. Set data package to data package BITWISE OR of ENGINE POWER C Caution of EXTERNAL interfaces. left shift M\_FIVE.
8. Set data package to data package BITWISE OR of ENGINE 1 Bleed Value Offset of Bleed Value left shift M\_SIX.
9. Set data package to data package BITWISE OR of ENGINE 2 Bleed Value Offset of Bleed Value left shift M\_SEVEN.
10. Set data package to data package BITWISE OR of ENGINE 1 Bleed Value Open of Bleed Value left shift M\_EIGHT.
11. Set data package to data package BITWISE OR of ENGINE 2 Bleed Value Open of Bleed Value left shift M\_NINE.
12. Set data of Tx\_a429\_ddw of index of LABEL\_012\_ENGINE\_STATUS to data package.
13. Set status of Tx\_a429\_ddw of index of LABEL\_012\_ENGINE\_STATUS to OK.

### 10.41.69 TxProcessLabel013WarningStatus

Low Level Design Details about CSU TxProcessLabel013WarningStatus will follow in the sub sections.

#### 10.41.69.1 Brief Description

This function sets the TxProcessLabel013WarningStatus

#### 10.41.69.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.41.69.3 List of global variables accessed and modified

Accessed: Hyd\_cautions

Mgb\_oil

Tgb\_oil\_temp

Mgb\_chip

Source\_selection\_sent

Modified: Tx\_a429\_ddw

#### 10.41.69.4 Parameter list (Input/Output)

None

#### 10.41.69.5 Return Value

None

#### 10.41.69.6 Other CSUs called by this CSU

None

#### 10.41.69.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TxProcessLabel013WarningStatus

##### 10.41.69.7.1 daugwylogicairbus-TxProcessLabel013WarningStatus-LLR-001

Requirement ID: H398-LLD-GWY-FNC-7207

The function shall do the following.

1. Set id of Tx\_a429\_ddw of index LABEL\_013\_ENGINE\_STATUS to M\_ZERO.
2. Set data package to HYDRAULIC aux pump low pressure alert of Hydraulic Cautions.
3. Set data package to data package of BITWISE OR of HYDRAULIC auto pilot circuit pressure alert of Hydraulic Cautions left shift M\_ONE.
4. Set data package to data package of BITWISE OR of HYDRAULIC 1 main pump low pressure alert of Hydraulic Cautions left shift M\_TWO.
5. Set data package to data package of BITWISE OR of HYDAULIC 1 low level alert of Hydraulic Cautions left shift M\_THREE.
6. Set data package to data package of BITWISE OR of HYDRAULIC 2 low level alert of Hydraulic Cautions left shift M\_FOUR.
7. Set data package to data package of BITWISE OR of HYDRAULIC 2 pump low pressure alert of Hydraulic Cautions left shift M\_FIVE.
8. Set data package to data package of BITWISE OR of HYDRAULIC 1 aux pump fail alert of Hydraulic Cautions left shift M\_SIX.
9. Set data package to data package of BITWISE OR of MGB oil temp alarm of MGB Oil left shift M\_SEVEN.
10. Set data package to data package of BITWISE OR of MGB Main pump low pressure alarm of MGB Oil left shift M\_EIGHT.
11. Set data package to data package of BITWISE OR of MGB Standby pump low pressure alarm of MGB Oil left shift M\_NINE.
12. Set data package to data package of BITWISE OR of IGB Oil Temperature left shift M\_TEN.
13. Set data package to data package of BITWISE OR of TGB Oil Temperature left shift M\_ELEVEN.
14. Set data package to data package of BITWISE OR of MGB Chip left shift M\_TWELVE.

##### 10.41.69.7.2 daugwylogicairbus-TxProcessLabel013WarningStatus-LLR-005

Requirement ID: H398-LLD-GWY-FNC-7339

The function shall do the following.

* Set EIU1status to FALSE
* Set EIU2status to FALSE

When reconfiguration\_state\_sent is DAU\_N\_MODE.

##### 10.41.69.7.3 daugwylogicairbus-TxProcessLabel013WarningStatus-LLR-002

Requirement ID: H398-LLD-GWY-FNC-7208

The function shall do the following.

* Set EIU1status to TRUE
* Set EIU2status to FALSE

When source\_selection\_sent is EIU\_2.

##### 10.41.69.7.4 daugwylogicairbus-TxProcessLabel013WarningStatus-LLR-003

Requirement ID: H398-LLD-GWY-FNC-7209

The function shall do the following.

* Set EIU1status to FALSE
* Set EIU2status to TRUE

When source\_selection\_sent is EIU\_1.

##### 10.41.69.7.5 daugwylogicairbus-TxProcessLabel013WarningStatus-LLR-004

Requirement ID: H398-LLD-GWY-FNC-7210

The function shall do the following.

1. Set data package to data package of BITWISE OR of EIU1status left shift M\_THIRTEEN.
2. Set data package to data package of BITWISE OR of EIU2status left shift M\_FOURTEEN.
3. Set data package to data package of BITWISE OR of HYDRAULIC SERVO alert of Hydraulic Cautions left shift M\_FIFTEEN\_A.
4. Set data of Tx\_a429\_ddw of index LABEL\_013\_ENGINE\_STATUS to data package.
5. Set status of Tx\_a429\_ddw of index LABEL\_013\_ENGINE\_STATUS to OK.

### 10.41.70 TxProcessLabel356EngineStatus1

Low Level Design Details about CSU TxProcessLabel356EngineStatus1 will follow in the sub sections.

#### 10.41.70.1 Brief Description

This function sets the data, status, and ID for the Label 356-01 & Label 356-10 - Engine 1 & 2 System Status Data Packet A429 messages.

#### 10.41.70.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.41.70.3 List of global variables accessed and modified

Accessed: Engine\_state

Engine\_oei\_state

Ext\_interfaces

Modified: Tx\_a429\_ddw

#### 10.41.70.4 Parameter list (Input/Output)

None

#### 10.41.70.5 Return Value

None

#### 10.41.70.6 Other CSUs called by this CSU

None

#### 10.41.70.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TxProcessLabel356EngineStatus1

##### 10.41.70.7.1 daugwylogicairbus-TxProcessLabel356EngineStatus1-LLR-001

Requirement ID: H398-LLD-GWY-FNC-7219

The function shall loops from M\_ZERO to M\_MAX\_ENGINE minus one and prforms the following :

1. Set id of Tx\_a429\_ddw of index LABEL\_356\_ENGINE\_STATUS1\_1 with loop index to loop index with M\_ONE.
2. Set data package of index loop index to OEI quarter Circle Amber Display.
3. Set data package of index loop index to data package of index loop index BITWISE OR of OEI quarter Circle Red Display left shift M\_ONE.
4. Set data package of index loop index to data package of index loop index BITWISE OR of OEI quarter Circle White Display left shift M\_TWO.
5. Set data package of index loop index to data package of index loop index BITWISE OR of OEI quarter Circle Blink Display left shift M\_THREE.
6. Set data package of index loop index to data package of index loop index BITWISE OR of OEI quarter Full Circle Display left shift M\_FOUR.
7. Set data package of index loop index to data package of index loop index BITWISE OR of ENGINE Training of index M\_ENGINE\_1 with loop index of Engine State left shift M\_FIVE.
8. Set data package of index loop index to data package of index loop index BITWISE OR of ENGINE OEI State of index M\_ENGINE\_1 sum up with loop index of Engine\_oei\_state left shift M\_SIX.
9. Set data package of index loop index to data package of index loop index BITWISE OR of ENGINE Failure Frame of index M\_ENGINE\_1 with loop index of Engine State left shift M\_TEN.
10. Set data of Tx\_a429\_ddw of index LABEL\_356\_ENGINE\_STATUS1\_1 with loop index to data package of loop index.
11. Set status of Tx\_a429\_ddw of index LABEL\_356\_ENGINE\_STATUS1\_1 with loop index to OK.

### 10.41.71 TxProcessLabel357EngineStatus2

Low Level Design Details about CSU TxProcessLabel357EngineStatus2 will follow in the sub sections.

#### 10.41.71.1 Brief Description

This function processes the Label 357 Engine Status for both Engine 1 and Engine 2. It sets the ID, data, and status based on various conditions and engine states.

#### 10.41.71.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.41.71.3 List of global variables accessed and modified

Accessed: Ext\_interfaces

deltang\_oei\_ctdiode

Engine\_state

Modified: Tx\_a429\_ddw

#### 10.41.71.4 Parameter list (Input/Output)

None

#### 10.41.71.5 Return Value

None

#### 10.41.71.6 Other CSUs called by this CSU

None

#### 10.41.71.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TxProcessLabel357EngineStatus2

##### 10.41.71.7.1 daugwylogicairbus-TxProcessLabel357EngineStatus2-LLR-001

Requirement ID: H398-LLD-GWY-FNC-7228

The function shall loops from M\_ZERO to M\_MAX\_ENGINE minus one and prforms the following :

i) Set id of Tx\_a429\_ddw of index LABEL\_357\_ENGINE\_STATUS2\_1 with

loop index to loop index with M\_ONE.

ii) Set data package of index loop index to ENGINE HIGH MESSAGE of index

M\_ENGINE\_1 with loop index of EXTERNAL interfaces.

iii) Set data package of index loop index to data package of index loop index BITWISE

OR of ENGINE LOW MESSAGE of index M\_ENGINE\_1 with loop index of

EXTERNAL interfaces left shift M\_ONE.

iv) Set data package of index loop index to data package of index loop index

BITWISE OR of ENGINE CT Message of M\_ENGINE with loop index left shift M\_TWO.

v) Set data package of index loop index to data package of index loop index BITWISE OR

of Delta Engine Speed (NG) One Engine Inoperative CT Diode of index

M\_ENGINE\_1 with loop index left shift M\_SIX.

vi) Set data package of index loop index to data package of index loop index BITWISE

OR of Delta Engine OEI One Engine inoperative Low Diode of M\_ENGINE\_1 with

loop index left shift M\_SEVEN.

vii) Set data package of index loop index to data package of index loop index

BITWISE OR of Delta Engine OEI One Engine inoperative High Diode of

M\_ENGINE\_1 with loop index left shift M\_EIGHT.

viii) Set data package of index loop index to data package of index loop index

BITWISE OR T4 One Engine inoperative diode Display left shift M\_NINE.

ix) Set data package of index loop index to data package of index loop

index BITWISE OR T4 One Engine inoperative diode Display left shift M\_TEN.

x) Set data package of index loop index to data package of index loop index

BITWISE OR T4 One Engine inoperative diode Display left shift M\_ELEVEN.

xi) Set data package of index loop index to data package of index loop index

BITWISE OR of ENGINE START ADVANCE of index M\_ENGINE with loop

index of Engine State left shift M\_TWELVE.

xii) Set data package of index loop index to data package of index loop index

BITWISE OR of Engine Start Blinking of index M\_ENGINE\_1 with loop index

left shift M\_THIRTEEN.

xiii) Set data package of index loop index to data package of index loop index

BITWISE OR of ENGINE GOV Caution of index M\_ENGINE\_1 with loop index

left shift M\_FOURTEEN.

xiv) Set data package of index loop index to data package of index loop index

BITWISE OR of Engine Gov blinking of M\_ENGINE with loop index left shift M\_FIFTEEN\_A.

xv) Set data of Tx\_a429\_ddw of index LABEL\_357\_ENGINE\_STATUS2\_1

with loop index to data package of loop index.

xvi) Set status of Tx\_a429\_ddw of index LABEL\_357\_ENGINE\_STATUS2\_1 with loop index to OK.

### 10.41.72 TxProcessLabel030ModeMgmt

Low Level Design Details about CSU TxProcessLabel030ModeMgmtwill follow in the sub sections.

#### 10.41.72.1 Brief Description

This function sets the data, status, and ID for the Label 030-01 & Label 030-10 - DU 1 & 2 Data Packet A429 messages.

#### 10.41.72.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.41.72.3 List of global variables accessed and modified

Accessed: Rx\_a429

Xram\_data\_buffer

Modified: Tx\_a429\_bnr

#### 10.41.72.4 Parameter list (Input/Output)

None

#### 10.41.72.5 Return Value

None

#### 10.41.72.6 Other CSUs called by this CSU

None

#### 10.41.72.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TxProcessLabel030ModeMgmt

##### 10.41.72.7.1 daugwylogicairbus-TxProcessLabel030ModeMgmt-LLR-001

Requirement ID: H398-LLD-GWY-FNC-7237

The function shall do the following

1. Set day night dis with NOT of (disci of index DI\_91\_DAY\_NIGHT of App\_A825 of Xram\_data\_buffer)
2. Set nvg normal dis with NOT of (disci of index DI\_92\_NVG\_NORMAL of App\_A825 of Xram\_data\_buffer

##### 10.41.72.7.3 daugwylogicairbus-TxProcessLabel030ModeMgmt-LLR-002

Requirement ID: H398-LLD-GWY-FNC-7238

The function shall do the following

* Set Page\_mode to PREFLIGHT
* Set Discrete\_outs with index of DISC\_OUT\_10 to ON.
* Set Discrete\_outs with index of DISC\_OUT\_11 to ON

When disci with index of DI\_1\_ON\_GROUND of disci of App\_A825 of Xram\_data\_buffer AND disci with index of DI\_31\_PFTTEST of disci of App\_A825 of Xram\_data\_buffer otherwise set Page\_mode to ENG\_AND\_VEH

##### 10.41.72.7.4 daugwylogicairbus-TxProcessLabel030ModeMgmt-LLR-003

Requirement ID: H398-LLD-GWY-FNC-7239

The function shall loop the count from M\_ZERO to M\_MAX\_ENGINE minus one and perform the following:

- Set id of  Tx\_a429\_bnr of index [LABEL\_030\_MODE\_MANAGEMENT\_1 plus count]  with sum of count and M\_ONE

- Set data\_package of index  count  is equal to day night dis

- Set data\_package of  index count with   BITWISE OR data\_package of  index count and (nvg normal dis shift by M\_ONE).

-Set data\_package of  index count with   BITWISE OR with data\_package of  index count and (Page\_mode left shifted by M\_TWELVE).

- Set   data of index [LABEL\_030\_MODE\_MANAGEMENT\_1 plus count] of Tx\_a429\_bnr with

  datapackage of  index count

- Set of  status of index  [LABEL\_030\_MODE\_MANAGEMENT\_1 plus count] of Tx\_a429\_bnr  to OK

### 10.41.73 TxProcessLabel077LightingState

Low Level Design Details about CSU TxProcessLabel077LightingState will follow in the sub sections.

#### 10.41.73.1 Brief Description

This function sets the data, status, and ID for the Label 077-01 & Label 077-10 - DU 1 & 2 Data Packet A429 messages.

#### 10.41.73.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.41.73.3 List of global variables accessed and modified

Accessed: Rx\_a429

Xram\_data\_buffer

Modified: Tx\_a429\_bnr

#### 10.41.73.4 Parameter list (Input/Output)

None

#### 10.41.73.5 Return Value

None

#### 10.41.73.6 Other CSUs called by this CSU

None

#### 10.41.73.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TxProcessLabel077LightingState

##### 10.41.73.7.1 daugwylogicairbus-TxProcessLabel077LightingState-LLR-002

Requirement ID: H398-LLD-GWY-FNC-7249

The function shall perform the following

- Set nvg normal label77 with BITWISE AND of ( data of Rx\_a429 of index

[Priority of index IP\_LABEL\_077\_LIGHTING\_2 and IP\_LABEL\_077\_LIGHTING\_2] is RIGHT SHIFT by M\_THIRTEEN )and M\_ONE

- Set day night label77 with by BITWISE AND of ( data of Rx\_a429 of index

[Priority of index IP\_LABEL\_077\_LIGHTING\_2 and IP\_LABEL\_077\_LIGHTING\_2] is RIGHT SHIFT by MM\_FOURTEEN)and M\_ONE

When status of Rx\_a429 of index [Priority of index IP\_LABEL\_077\_LIGHTING\_2 and IP\_LABEL\_077\_LIGHTING\_2] is equal to OK

##### 10.41.73.7.2 daugwylogicairbus-TxProcessLabel077LightingState-LLR-003

Requirement ID: H398-LLD-GWY-FNC-7250

The function shall loop count from M\_ZERO to M\_MAX\_ENGINE minus one and perform the following:

- Set id of Tx\_a429\_bnr of index [LABEL\_077\_LIGHTING\_1 plus count] with sum of count and M\_ONE

- Set data package of index count with nvg normal label77 LEFT shift by M\_THIRTEEN).

- Set datapackage of index count with BITWISE OR of (datapackage of index count and day night label77 LEFT shift by M\_FOURTEEN)

- Set data of index [LABEL\_077\_LIGHTING\_1 plus count] of Tx\_a429\_bnr with datapackage of index count

- Set of status of index [LABEL\_077\_LIGHTING\_1 plus count] of Tx\_a429\_bnr to OK

### 10.41.74 TxProcessLabel035DiscrepRpt

Low Level Design Details about CSU TxProcessLabel035DiscrepRptwill follow in the sub sections.

#### 10.41.74.1 Brief Description

This function sets the data, status, and ID for the Label 035-01 - Discrepancy Report A429 message.

#### 10.41.74.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.41.74.3 List of global variables accessed and modified

Accessed: Mis\_cmp\_fail

Modified: Tx\_a429\_ddw

#### 10.41.74.4 Parameter list (Input/Output)

None

#### 10.41.74.5 Return Value

None

#### 10.41.74.6 Other CSUs called by this CSU

None

#### 10.41.74.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TxProcessLabel035DiscrepRpt

##### 10.41.74.7.1 daugwylogicairbus-TxProcessLabel035DiscrepRpt-LLR-001

Requirement ID: H398-LLD-GWY-FNC-7259

The function shall perform the following

- Set id of Tx\_a429\_ddw of index LABEL\_035\_DISCREPANCY\_REPORT to M\_ZERO.

- Set datapackage to analog\_input\_miscompare of index EOP\_1 of Mis\_cmp\_fail.

- Set datapackage to BITWISE OR of datapackage and (analog\_input\_miscompare of index EOP\_2

of Mis\_cmp\_fail LEFT SHIFT by M\_ONE)

- Set datapackage to BITWISE OR of datapackage and (analog\_input\_miscompare of index EOT\_1

of Mis\_cmp\_fail LEFT SHIFT by M\_TWO)

- Set datapackage to BITWISE OR of datapackage and (analog\_input\_miscompare of index EOT\_2 of

Mis\_cmp\_fail LEFT SHIFT by M\_THREE)

- Set datapackage to BITWISE OR of datapackage and (analog\_input\_miscompare of

index FP\_1 of Mis\_cmp\_fail LEFT SHIFT by M\_FOUR)

- Set datapackage to BITWISE OR of datapackage and (analog\_input\_miscompare of

index FP\_2 of Mis\_cmp\_fail LEFT SHIFT by M\_FIVE)

- Set datapackage to BITWISE OR of datapackage and (analog\_input\_miscompare of

index HP\_1 of Mis\_cmp\_fail LEFT SHIFT by M\_SIX)

- Set datapackage to BITWISE OR of datapackage and (analog\_input\_miscompare of

index HP\_2 of Mis\_cmp\_fail LEFT SHIFT by M\_SEVEN)

- Set datapackage to BITWISE OR of datapackage and (analog\_input\_miscompare of

index MGBOP\_T of Mis\_cmp\_fail LEFT SHIFT by M\_EIGHT)

- Set datapackage to BITWISE OR of datapackage and (analog\_input\_miscompare of

index MGBT\_T of Mis\_cmp\_fail LEFT SHIFT by M\_NINE)

- Set datapackage to BITWISE OR of datapackage and (analog\_input\_miscompare of

index (M\_TEN plus TOTAL\_TRQ) of Mis\_cmp\_fail LEFT SHIFT by M\_TEN)

- Set datapackage to BITWISE OR of datapackage and (analog\_input\_miscompare of

index (M\_TEN plus TRQ\_1) of Mis\_cmp\_fail LEFT SHIFT by M\_ELEVEN)

- Set datapackage to BITWISE OR of datapackage and (analog\_input\_miscompare of

index (M\_TEN plus TRQ\_2) of Mis\_cmp\_fail LEFT SHIFT by M\_TWELVE)

- Set datapackage to BITWISE OR of datapackage and (analog\_input\_miscompare of

index (M\_TEN plus T4\_1) of Mis\_cmp\_fail LEFT SHIFT by M\_THIRTEEN)

- Set datapackage to BITWISE OR of datapackage and (analog\_input\_miscompare of

index (M\_TEN plus T4\_2) of Mis\_cmp\_fail LEFT SHIFT by M\_FOURTEEN)

- Set datapackage to BITWISE OR of datapackage and (analog\_input\_miscompare of

index (M\_TEN plus NG\_1) of Mis\_cmp\_fail LEFT SHIFT by M\_FIFTEEN\_A)

- Set datapackage to BITWISE OR of datapackage and (analog\_input\_miscompare of

index (M\_TEN plus NG\_2) of Mis\_cmp\_fail LEFT SHIFT by M\_SIXTEEN)

- Set datapackage to BITWISE OR of datapackage and (analog\_input\_miscompare of

index (M\_TEN plus DELTA\_NG\_1) of Mis\_cmp\_fail LEFT SHIFT by M\_SEVENTEEN)

- Set datapackage to BITWISE OR of datapackage and (analog\_input\_miscompare of

index (M\_TEN plus DELTA\_NG\_2) of Mis\_cmp\_fail LEFT SHIFT by M\_EIGHTEEN)

- Set data of Tx\_a429\_ddw of index LABEL\_035\_DISCREPANCY\_REPORT to datapackage.

- Set status of Tx\_a429\_ddw of index LABEL\_035\_DISCREPANCY\_REPORT to OK.

### 10.41.76 T4Roundup

Low Level Design Details about CSU T4Roundup will follow in the sub sections.

#### 10.41.76.1 Brief Description

This function rounds the input T4ActualValue to the nearest multiple of 5.

#### 10.41.76.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.41.76.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 10.41.76.4 Parameter list (Input/Output)

T\_FLOAT32 - T4ActualValue

#### 10.41.76.5 Return Value

T\_FLOAT32 - returns the rounded value

#### 10.41.76.6 Other CSUs called by this CSU

None

#### 10.41.76.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to T4Roundup

##### 10.41.76.7.1 daugwylogicairbus-T4Roundup-LLR-001

Requirement ID: H398-LLD-GWY-FNC-7277

The function shall do the following

- Set T4roundedto5 to the product of (fraction of T4ActualValue and five) and five

- Set T4roundedto5 to sum of T4Round5 and five

when T4ActualValue is greater than M\_FP\_ZERO and difference of T4 ActualValue and T4roundedto5) is greater than fraction FIVE by TWO

Otherwise,

Set T4roundedto5 to difference of T4roundedto5 and five when difference of T4 ActualValue and T4roundedto5 multiplied with M\_FP\_NEGATIVE\_ONE is greater than M\_FP\_FIVE divided by M\_FP\_TWO

##### 10.41.76.7.2 daugwylogicairbus-T4Roundup-LLR-002

Requirement ID: H398-LLD-GWY-FNC-7278

The function shall return the T4Round5

### 10.41.77 TxProcessLabel062DAUBIT

Low Level Design Details about CSU TxProcessLabel062DAUBIT will follow in the sub sections.

#### 10.41.77.1 Brief Description

This function Processes data for Label 062 (DAU BIT Discrete) transmission, handling fault detection, channel status

and configuration checks

#### 10.41.77.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.41.77.3 List of global variables accessed and modified

Accessed: Cmdownload\_lp\_cmd\_data

Rx\_a429

reconfiguration\_state\_sent

Analog\_data\_missing

Discrete\_data\_missing

RS422\_loop\_back\_fail\_flag

Nvm\_cmu\_tx

Cmu\_data\_missing

Rs232\_transmit\_data

Modified: Tx\_a429\_bnr

AnalogMiscompareFail

Mis\_cmp\_fail

#### 10.41.77.4 Parameter list (Input/Output)

None

#### 10.41.77.5 Return Value

None

#### 10.41.77.6 Other CSUs called by this CSU

None

#### 10.41.77.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TxProcessLabel062DAUBIT

##### 10.41.77.7.1 daugwylogicairbus-TxProcessLabel062DAUBIT-LLR-001

Requirement ID: H398-LLD-GWY-FNC-7349

The function shall perform the following:

1. Set id of Tx\_a429\_bnr of index LABEL\_062\_DAU\_BIT to M\_ZERO.
2. Set data of Tx\_a429\_bnr of index LABEL\_062\_DAU\_BIT to M\_ZERO.

##### 10.41.77.7.2 daugwylogicairbus-TxProcessLabel062DAUBIT-LLR-002

Requirement ID: H398-LLD-GWY-FNC-7350

The function shall loops from M\_ZERO to M\_A429\_MAX\_RX\_CHAN minus one and performs the following:

* Set Channel62 of index count to TRUE when status of Rx\_a429 of index [count and u8 labelindex] is MISSING and loops u8 labelindex from M\_DEC\_ZERO to numOf429in of Cmdownload\_lp\_cmd\_data otherwise Set Channel62 of index count FALSE

##### 10.41.77.7.3 daugwylogicairbus-TxProcessLabel062DAUBIT-LLR-003

Requirement ID: H398-LLD-GWY-FNC-7351

The function shall loops from M\_ZERO to M\_A429\_MAX\_RX\_CHAN minus one and loop from M\_DEC\_ZERO to u8\_label\_index less than numOf429in of Cmdownload\_lp\_cmd\_data following:

* Set Channel of index count to TRUE when status of Rx\_a429 with index (index count and index count ) is equal to MISSING returns otherwise,Set Channel of index count to FALSE.

##### 10.41.77.7.4 daugwylogicairbus-TxProcessLabel062DAUBIT-LLR-004

Requirement ID: H398-LLD-GWY-FNC-7352

The function shall perform the following:

* Set channel\_interconnect\_failure of Mis\_cmp\_fail to FALSE
* Set AnalogMiscompareFail to FALSE.
* Set discrete\_input\_miscompare of Mis\_cmp\_fail to FALSE

when (Reconfiguration\_state\_sent is equal to DAU\_1\_MODE)OR (Reconfiguration\_state\_sent is equal to DAU\_2\_MODE))

##### 10.41.77.7.5 daugwylogicairbus-TxProcessLabel062DAUBIT-LLR-005

Requirement ID: H398-LLD-GWY-FNC-7353

The function shall perform the following:

* Increment five seconds counter.
* Set AnalogMiscompareFail to FALSE.
* Set discrete\_input\_miscompare of Mis\_cmp\_fail to FALSE.
* Set channel\_interconnect\_failure of Mis\_cmp\_fail to FALSE.
* Set airbus\_analog\_fault of Mis\_cmp\_fail to FALSE.

When five seconds sounter is less than the ( product of M\_TIMER\_TEMP and M\_FIVE) otherwise Set channel\_interconnect\_failure of Mis\_cmp\_fail to ((NOT of Rs232\_transmit\_data)

OR A429\_loop\_back\_fail\_flag)

##### 10.41.77.7.6 daugwylogicairbus-TxProcessLabel062DAUBIT-LLR-006

Requirement ID: H398-LLD-GWY-FNC-7354

The function shall perform the following:

* Set IO\_processor\_failure of Mis\_cmp\_fail to TRUE when Analog\_data\_missing is equal to TRUE OR Discrete\_data\_missing is equal to TRUE

returns TRUE. Otherwise Set IO\_processor\_failure of Mis\_cmp\_fail to FALSE.

##### 10.41.77.7.7 daugwylogicairbus-TxProcessLabel062DAUBIT-LLR-007

Requirement ID: H398-LLD-GWY-FNC-7355

The function shall perform the following:

* Set NVM\_read\_write\_fail of Mis\_cmp\_fail to TRUE when (b\_nvm\_read\_write\_failed of bitwise\_status of pbit of NVM\_BuiltInTests of

Nvm\_cmu\_tx is equal to M\_ONE) OR (Cmu\_data\_missing is equal to TRUE ) returns TRUE.

Otherwise Set NVM\_read\_write\_fail of Mis\_cmp\_fail to FALSE.

##### 10.41.77.7.8 daugwylogicairbus-TxProcessLabel062DAUBIT-LLR-008

Requirement ID: H398-LLD-GWY-FNC-7356

The function shall perform the following:

* Set AnalogMiscompareFail to FALSE.
* Set discrete\_input\_miscompare of Mis\_cmp\_fail to FALSE.
* Set airbus\_analog\_fault of Mis\_cmp\_fail to FALSE.

When Rs232\_transmit\_data is equal to FALSE.

##### 10.41.77.7.9 daugwylogicairbus-TxProcessLabel062DAUBIT-LLR-009

Requirement ID: H398-LLD-GWY-FNC-7357

The function shall perform the following:

* Set data of Tx\_a429\_bnr of index LABEL\_062\_DAU\_BIT to airbus\_analog\_fault of Mis\_cmp\_fail OR with Analog\_miscompare\_fail BITWISE OR discrete\_input\_miscompare of Mis\_cmp\_fail LEFT shift by M\_ONE BITWISE OR

ChannelDU of index DU\_1 LEFT shift by M\_THREE BITWISE OR ChannelDU of index DU\_2 LEFT shift by M\_FOUR)

BITWISE OR channel\_interconnect\_failure of Mis\_cmp\_fail LEFT shift by M\_NINE

BITWISE OR IO\_processor\_failure of Mis\_cmp\_fail LEFT shift by M\_TWELVE BITWISE OR Reconfig\_capability\_test\_fail LEFT shift by M\_THIRTEEN) BITWISE OR

NVM\_read\_write\_fail of Mis\_cmp\_fail LEFT shift by M\_EIGHTEEN)

* Set status of Tx\_a429\_bnr of index LABEL\_062\_DAU\_BIT to OK

### 10.41.82 DeltaNGThreshDiscreteOutput

Low Level Design Details about CSU DeltaNGThreshDiscreteOutput will follow in the sub sections.

#### 10.41.82.1 Brief Description

This function performs the Delta NG Threshold Discrete output as ON or OFF.

#### 10.41.82.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.41.82.3 List of global variables accessed and modified

Accessed: Deltang\_sigvalue

Modified: Discrete\_outs

#### 10.41.82.4 Parameter list (Input/Output)

None

#### 10.41.82.5 Return Value

None

#### 10.41.82.6 Other CSUs called by this CSU

None

#### 10.41.82.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to DeltaNGThreshDiscreteOutput

##### 10.41.82.7.1 daugwylogicairbus-DeltaNGThreshDiscreteOutput-LLR-001

Requirement ID: H398-LLD-GWY-FNC-7402

The function shall perform the following:

1. Set Discrete outs of index DISC\_OUT\_10 and Discrete outs of index DISC\_OUT\_11 to ON when Deltang sigvalue of index M\_ZERO is greater than the M\_DELTA\_NG\_THRESHOLD OR Deltang sigvalue of index M\_ONE is greater than M\_DELTA\_NG\_THRESHOLD OR Dau\_eng\_status with index DELTA\_NG1 is not equal to OK OR Dau\_eng\_status with index DELTA\_NG2 is not equal to OK
2. Set Discrete\_outs of index DISC\_OUT\_10 and Discrete outs of index DISC\_OUT\_11 to OFF when Deltang sigvalue of index M\_ZERO is less than or equal to M\_DELTA\_NG\_THRESHOLD OR Deltang sigvalue of index M\_ONE is less than or equal to M\_DELTA\_NG\_THRESHOLD and PREFLIGHT is not equal to Page mode.

### 10.41.83 ArincLineStatus

Low Level Design Details about CSU ArincLineStatus will follow in the sub sections.

#### 10.41.83.1 Brief Description

The ArincLineStatus function checks the status of two ARINC lines (DAU1 and DAU2) based on messages received from DU and Remise.

#### 10.41.83.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.41.83.3 List of global variables accessed and modified

Accessed               : Rx\_a429

Modified                 :  None

#### 10.41.83.4 Parameter list (Input/Output)

Inputs    : None

Outputs : None

#### 10.41.83.5 Return Value

None.

#### 10.41.83.6 Other CSUs called by this CSU

GpioReadInputDataBit

#### 10.41.83.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to ArincLineStatus.

##### 10.41.83.7.1 daugwylogicairbus-ArincLineStatus-LLR-001

Requirement ID: H398-LLD-GWY-FNC-7455

The function shall do the following when status of Rx\_a429 with index of ((Priority with index of IP\_LABEL\_011\_SYSMSG\_1) and IP\_LABEL\_011\_SYSMSG\_1) is equal to OK.

1. Set eiu1 status with index of M\_ZERO to Bitwise AND of (data of Rx\_a429 with index of Priority with index of IP\_LABEL\_011\_SYSMSG\_1 and IP\_LABEL\_011\_SYSMSG\_1) and M\_ONE.
2. Set eiu2 status with index of M\_ZERO to Bitwise AND of ((data of Rx\_a429 with index of Priority with index of IP\_LABEL\_011\_SYSMSG\_1 and IP\_LABEL\_011\_SYSMSG\_1) right shifted by M\_ONE) and M\_ONE.

##### 10.41.83.7.2 daugwylogicairbus-ArincLineStatus-LLR-002

Requirement ID: H398-LLD-GWY-FNC-7456

The function shall do the following when status of Rx\_a429 with index of ((Priority with index of IP\_LABEL\_011\_SYSMSG\_2) and IP\_LABEL\_011\_SYSMSG\_2) is equal to OK.

1. Set eiu1 status with index of M\_ONE to Bitwise AND of (data of Rx\_a429 with index of Priority with index of IP\_LABEL\_011\_SYSMSG\_2 and IP\_LABEL\_011\_SYSMSG\_2) and M\_ONE.
2. Set eiu2 status with index of M\_ONE to Bitwise AND of ((data of Rx\_a429 with index of Priority with index of IP\_LABEL\_011\_SYSMSG\_2 and IP\_LABEL\_011\_SYSMSG\_2) right shifted by M\_ONE) and M\_ONE.

##### 10.41.83.7.3 daugwylogicairbus-ArincLineStatus-LLR-003

Requirement ID: H398-LLD-GWY-FNC-7457

The function shall do the following when Return value of function call GpioReadInputDataBit with parameters (M\_GPIOC, M\_GPIOC\_SYS\_SEL) through M\_HW\_GET\_RCI is equal to M\_ZERO.

1. Set arinc\_valid to TRUE when eiu1 status with index of M\_ZERO is equal to M\_ONE OR eiu1 status with index of M\_ONE is equal to M\_ONE otherwise set arinc\_valid to FALSE.

##### 10.41.83.7.4 daugwylogicairbus-ArincLineStatus-LLR-004

Requirement ID: H398-LLD-GWY-FNC-7458

The function shall do the following when Return value of function call GpioReadInputDataBit with parameters (M\_GPIOC, M\_GPIOC\_SYS\_SEL) through M\_HW\_GET\_RCI is equal to M\_ONE.

1. Set arinc\_valid to TRUE when eiu2 status with index of M\_ZERO is equal to M\_ONE OR eiu2 status with index of M\_ONE is equal to M\_ONE otherwise set arinc\_valid to FALSE.

### 10.41.84 TxProcessLabel064CKSUMMSB

Low Level Design Details about CSU TxProcessLabel064CKSUMMSB will follow in the sub sections.

#### 10.41.84.1 Brief Description

The TxProcessLabel064CKSUMMSB function processes transmission label .

#### 10.41.84.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.41.84.3 List of global variables accessed and modified

Accessed               : NONE

Modified                 :  Sys\_checksum, Tx\_a429\_ddw

#### 10.41.84.4 Parameter list (Input/Output)

Inputs    : None

Outputs : None

#### 10.41.84.5 Return Value

None.

#### 10.41.84.6 Other CSUs called by this CSU

CalcSysChecksum

#### 10.41.84.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TxProcessLabel064CKSUMMSB.

##### 10.41.84.7.1 daugwylogicairbus-TxProcessLabel064CKSUMMSB-LLR-001

Requirement ID: H398-LLD-GWY-FNC-7460

The function shall perform the following

1. Set Sys\_checksum with the return value of CalcSysChecksum.
2. Set checksum msb word to Sys\_checksum right shifted by M\_SIXTEEN.
3. Set id of Tx\_a429\_bnr with index of LABEL\_064\_SWCRC\_MSB\_1 to M\_HW\_GET\_RCI plus M\_ONE.
4. Set data of Tx\_a429\_bnr with index of LABEL\_064\_SWCRC\_MSB\_1 to BITWISE OR of (M\_ZERO and (checksum msb word left shifted by M\_THREE).
5. Set status of Tx\_a429\_bnr with index of LABEL\_064\_SWCRC\_MSB\_1 to OK.

### 10.41.85 TxProcessLabel065CKSUMLSB

Low Level Design Details about CSU TxProcessLabel065CKSUMLSB will follow in the sub sections.

#### 10.41.85.1 Brief Description

The TxProcessLabel065CKSUMLSB function processes transmission label .

#### 10.41.85.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.41.85.3 List of global variables accessed and modified

Accessed               : Sys\_checksum,

Modified                 :  Tx\_a429\_ddw

#### 10.41.85.4 Parameter list (Input/Output)

Inputs    : None

Outputs : None

#### 10.41.85.5 Return Value

None.



#### 10.41.85.6 Other CSUs called by this CSU

None

#### 10.41.85.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TxProcessLabel065CKSUMLSB.

##### 10.41.85.7.1 daugwylogicairbus -TxProcessLabel065CKSUMLSB-LLR-001

Requirement ID: H398-LLD-GWY-FNC-7461

The funtion shall do the following

* Set checksum lsb word to BITWISE AND of (Sys\_checksum and M\_SIXTEEN\_BIT\_MASK).
* Set id of Tx\_a429\_bnr with index of LABEL\_065\_SWCRC\_LSB\_1 to M\_HW\_GET\_RCI plus M\_ONE.
* Set data of Tx\_a429\_bnr with index of LABEL\_065\_SWCRC\_LSB\_1 to BITWISE OR of (M\_ZERO and (checksum lsb word left shifted by M\_THREE)).
* Set status of Tx\_a429\_bnr with index of LABEL\_065\_SWCRC\_LSB\_1 to OK.

### 10.41.86 TxProcessLabel031Units

Low Level Design Details about CSU TxProcessLabel031Units will follow in the sub sections.

#### 10.41.86.1 Brief Description

The TxProcessLabel031Units function transmits data packets for labels 031-01 and 031-10 on ARINC lines 1 and 2 (DU 1 & 2).

#### 10.41.86.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.41.86.3 List of global variables accessed and modified

Accessed               : None

Modified                 :  Tx\_a429\_bnr

#### 10.41.86.4 Parameter list (Input/Output)

Inputs    : None

Outputs : None

#### 10.41.86.5 Return Value

None.

#### 10.41.86.6 Other CSUs called by this CSU

None

#### 10.41.86.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TxProcessLabel031Units.

##### 10.41.86.7.1 daugwylogicairbus -TxProcessLabel031Units-LLR-001

Requirement ID: H398-LLD-GWY-FNC-7462

The function shall loop from count M\_ZERO until M\_MAX\_ENGINE minus one and perform the following:

1. Set id of Tx\_a429\_bnr with index of  LABEL\_031\_UNITS\_1 plus count to count plus M\_ONE.
2. Set data package with index of count to variable bar psi left shited by M\_FOUR.
3. Set data of Tx\_a429\_bnr with index of  LABEL\_031\_UNITS\_1 plus count to data package with index of count.
4. Set status of Tx\_a429\_bnr with index of  LABEL\_031\_UNITS\_1 plus count to OK.

### 10.41.87 HwAbsFlt

Low Level Design Details about CSU TxProcessLabel031Units will follow in the sub sections

#### 10.41.87.1 Brief Description

The HwAbsFlt function provides absolute value of given 32 bit input.

#### 10.41.87.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.41.87.3 List of global variables accessed and modified

Accessed               : None

Modified                 : None

#### 10.41.87.4 Parameter list (Input/Output)

Inputs    : T\_FLOAT32 input given input

Outputs : None

#### 10.41.87.5 Return Value

T\_FLOAT32 - absolute value of input.

#### 10.41.87.6 Other CSUs called by this CSU

None

#### 10.41.87.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to HwAbsFlt.

##### daugwylogicairbus-HwAbsFlt-LLR-001

Requirement ID: H398-LLD-GWY-FNC-7476

The function shall set the input to input subtracted by M\_DEC\_ZERO when given input is less than M\_DEC\_ZERO.

Otherwise, do nothing

and returns input.

### 10.41.88 TRQRoundup

Low Level Design Details about CSU TRQRoundup will follow in the sub sections

#### 10.41.88.1 Brief Description

The TRQRoundup function rounds the input Trq\_Actual\_Value to the nearest 0.5.

#### 10.41.88.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 10.41.88.3 List of global variables accessed and modified

Accessed               : None

Modified                 : None

#### 10.41.88.4 Parameter list (Input/Output)

Inputs    : T\_FLOAT32 Trq\_Actual\_Value - actual value to be rounded up

Outputs : None

#### 10.41.88.5 Return Value

T\_FLOAT32 - returns the rounded value.

#### 10.41.88.6 Other CSUs called by this CSU

None

#### 10.41.88.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TRQRoundup.

##### 10.41.88.7.1 daugwylogicairbus-TRQRoundup-LLR-001

Requirement ID: H398-LLD-GWY-FNC-7477

* The function shall do the following :

set the torque rouded to torque actual value multiplied with M\_FP\_TWO which upon divided by M\_FP\_TWO

* set the torque rouded to M\_FP\_ONE divided by M\_FP\_TWO which is added with torque rounded when torque rounded is subtracted by torque actual value is greater than M\_FP\_ONE is divided by M\_FP\_FOUR
* Returns torque rounded

# 11 Software Low Level Requirements - Gateway Module Library

This section specifies the Software Low Level Requirements for Gateway Module Library Software.

## 11.1 daulibmisc

This module provides all the miscellaneous firmware functions (add-on to CMSIS functions).

### 11.1.1 NvicPriorityGroupConfig

Low Level Design Details about CSU NvicPriorityGroupConfig will follow in the sub sections.

#### 11.1.1.1 Brief Description

The function NvicPriorityGroupConfig configures the priority grouping: pre-emption priority and sub priority.

#### 11.1.1.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 11.1.1.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 11.1.1.4 Parameter list (Input/Output)

Inputs: T\_UINT32 nvic\_priority\_group - specifies the priority grouping bits length.

Outputs: None

#### 11.1.1.5 Return Value

None

#### 11.1.1.6 Other CSUs called by this CSU

None

#### 11.1.1.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to NvicPriorityGroupConfig.

##### 11.1.1.7.1 daulibmisc-NvicPriorityGroupConfig-LLR-001

Requirement ID: H398-LLD-GWY-FNC-4781

The function shall set the priority group bits [10:8] in aircr (Application interrupt and reset control register) of M\_SCB(System control block) by performing Bitwise OR of M\_AIRCR\_VECTKEY\_MASK and nvic\_priority\_group value.

### 11.1.2 NvicInit

Low Level Design Details about CSU NvicInit will follow in the sub sections

#### 11.1.2.1 Brief Description

The function NvicInit initializes the NVIC peripheral according to the specifiedparameters in the nvic\_init\_struct

#### 11.1.2.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 11.1.2.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 11.1.2.4 Parameter list (Input/Output)

Inputs: T\_NVIC\_INIT \* nvic\_init\_struct - pointer to a T\_NVIC\_INIT structure that contains the configuration information for the specified NVIC peripheral.

Outputs: None

#### 11.1.2.5 Return Value

None

#### 11.1.2.6 Other CSUs called by this CSU

None

#### 11.1.2.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to NvicInit

##### 11.1.2.7.1 daulibmisc-NvicInit-LLR-001

Requirement ID: H398-LLD-GWY-FNC-4790

The function shall compute the interrupt priority and set the computed value to Interrupt Priority Register (IP) of NVIC if NVIC interrupt channel command (nvic\_irq\_channel\_cmd of nvic\_init\_struct) is equal to ENABLE i.e,

When DISABLE is not equal to nvic\_irq\_channel\_cmd of nvic\_init\_struct then compute the Corresponding IRQ Priority as per below logic:

1. Extracted Priority Group bits value (Priority group bits [10:8] of Application interrupt and reset control register) is extracted as M\_PRIORITY\_GROUP subtracted by ((aircr of M\_SCB) bitwise AND with M\_PRIORITY\_GROUP)) right shifted by M\_SHIFT\_BY\_8)
2. Preempt Priority value is set to M\_FOUR subtracted by Extracted Priority Group bits value.
3. Sub Priority value is set to Sub Priority value right shift by Extracted Priority Group bits value.
4. Extracted Priority Group bits value is set to (nvic\_irq\_channel\_preempt\_pri of nvic\_init\_struct left shift by Preempt Priority value)
5. Extracted Priority Group bits value is set to Extracted Priority Group bits value bitwise OR with (nvic\_irq\_channel\_subpriority of nvic\_init\_struct bitwise AND with Sub Priority value)
6. Extracted Priority Group bits value is set to Extracted Priority Group bits value left shift by M\_SHIFT\_BY\_4
7. ip\_reg of index (nvic\_irq\_channel of nvic\_init\_struct) of M\_NVIC is set to Extracted Priority Group bits value.

             Note: Refer DM00046982-with FPU-ref-manual.pdf for computing interrupt priority.

##### 11.1.2.7.2 daulibmisc-NvicInit-LLR-002

Requirement ID: H398-LLD-GWY-FNC-4791

The function shall configure interrupt set-enable register of NVIC to enable the selected interrupt channel by setting iser of M\_NVIC with index value (nvic\_irq\_channel of nvic\_init\_struct right shift by M\_SHIFT\_BY\_5) to (M\_SHIFT\_BY\_1 left shift by (nvic\_irq\_channel of nvic\_init\_struct   bitwise AND with M\_THIRTY\_ONE)).

##### 11.1.2.7.3 daulibmisc-NvicInit-LLR-003

Requirement ID: H398-LLD-GWY-FNC-4792

The function shall disable the selected interrupt channels by setting Interrupt clear-enable register for selected interrupt channel when NVIC interrupt channel command is disabled i.e,

Set icer of M\_NVIC with index value (nvic\_irq\_channel of nvic\_init\_struct right shift by M\_SHIFT\_BY\_5) to M\_SHIFT\_BY\_1 left shift by (nvic\_irq\_channel of nvic\_init\_struct bitwise AND with M\_THIRTY\_ONE) when DISABLE is equal to nvic\_irq\_channel\_cmd of nvic\_init\_struct.

### 11.1.3 NvicSetVectorTable

Low Level Design Details about CSU NvicSetVectorTable will follow in the sub sections

#### 11.1.3.1 Brief Description

The function NvicSetVectorTable sets the vector table location and Offset.

#### 11.1.3.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 11.1.3.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 11.1.3.4 Parameter list (Input/Output)

Inputs: T\_UINT32 nvic\_vect\_tab - specifies if the vector table is in RAM or FLASH memory.

T\_UINT32 offset - Vector Table base offset field.

Outputs: None

#### 11.1.3.5 Return Value

None

#### 11.1.3.6 Other CSUs called by this CSU

None

#### 11.1.3.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to NvicSetVectorTable

##### 11.1.3.7.1 daulibmisc-NvicSetVectorTable-LLR-001

Requirement ID: H398-LLD-GWY-FNC-4801

The function shall configure vector table offset register of system control block with vector table location (RAM or FLASH, bit 29) and offset (bits [29:7]) by setting vtor of M\_SCB to (nvic\_vect\_tab bitwise OR with (offset bitwise AND with M\_NVIC\_VTOR))

## 11.2 daulibstm32f4xxadc

This module provides firmware functions to manage the following functionalities of the Analog to Digital Convertor (ADC) peripheral:

* Initialization and Configuration
* Regular Channels Configuration
* Regular Channels DMA Configuration.

### 11.2.1 AdcInit

Low Level Design Details about CSU AdcInit will follow in the sub sections

#### 11.2.1.1 Brief Description

The function AdcInit initializes the ADC peripheral according to the specified parameters in the initialization structure

#### 11.2.1.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY)

#### 11.2.1.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 11.2.1.4 Parameter list (Input/Output)

Inputs: T\_ADC \*adc\_x - Where x can be 1, 2 or 3 to select the ADC peripheral.

T\_ADC\_INIT \*adc\_init\_struct - Pointer to a T\_ADC\_INIT structure that contains the configuration information for the specified ADC peripheral.

Outputs: T\_ADC \*adc\_x - where x can be 1, 2 or 3 to select the ADC peripheral

#### 11.2.1.5 Return Value

None

#### 11.2.1.6 Other CSUs called by this CSU

None

#### 11.2.1.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to AdcInit.

##### 11.2.1.7.1 daulibstm32f4xxadc-AdcInit-LLR-001

Requirement ID: H398-LLD-GWY-FNC-4821

The function shall clear the RES (resolution) and SCAN (scan mode) bits of ADC control register 1 and configure with adc\_scan\_convmode and adc\_resolution of adc\_init\_struct as follows:

* + Set Temp Register to cr1 of adc\_x.
  + Set Temp Register to Temp Register bitwise AND with M\_CR1\_CLEAR\_MASK
  + Set Temp Register to Temp Register bitwise OR with ((adc\_scan\_convmode of adc\_init\_struct left shift by M\_SHIFT\_BY\_8) bitwise OR with adc\_resolution of adc\_init\_struct)
  + Set cr1 of adc\_x to Temp Register

##### 11.2.1.7.2 daulibstm32f4xxadc-AdcInit-LLR-002

Requirement ID: H398-LLD-GWY-FNC-4822

The function shall clear the CONT, ALIGN, EXTEN and EXTSEL bits of ADC control register 2 and configure with adc\_dataalign, adc\_external\_trigconv, adc\_external\_trig\_convedge and adc\_continuous\_convmode of adc\_init\_struct i.e,

* + Set Temp Register to cr2 of adc\_x.
  + Set Temp Register to Temp Register bitwise AND with M\_CR2\_CLEAR\_MASK.
  + Set Temp Register to Temp Register bitwise OR with (adc\_dataalign of adc\_init\_struct bitwise OR with adc\_external\_trigconv of adc\_init\_struct bitwise OR with adc\_external\_trig\_convedge of adc\_init\_struct bitwise OR with (adc\_continuous\_convmode of adc\_init\_struct left shift by M\_SHIFT\_BY\_1)).
  + Set cr2 of adc\_x to Temp Register

##### 11.2.1.7.3 daulibstm32f4xxadc-AdcInit-LLR-003

Requirement ID: H398-LLD-GWY-FNC-4823

The function shall clear the L (Regular channel sequence length) bits of ADC regular sequence register 1 and configure total number of conversions in the regular channel conversion sequence i.e,

* + Set Temp Register to sqr1 of adc\_x.
  + Set Temp Register to (Temp Register bitwise AND with M\_SQR1\_L\_RESET).
  + Set Temp Register 2 to Temp Register 2 bitwise OR with (adc\_nbr\_of\_conversion of adc\_init\_struct subtracted by M\_ONE).
  + Set Temp Register to Temp Register bitwise OR with (Temp Register 2 left shift by M\_SHIFT\_BY\_20).
  + Set sqr1 of adc\_x to Temp Register

### 11.2.2 AdcCommonInit

Low Level Design Details about CSU AdcCommonInit will follow in the sub sections

#### 11.2.2.1 Brief Description

The function AdcCommonInit initializes the ADCs peripherals according to the specified parameters in the ADC initialization structure

#### 11.2.2.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY)

#### 11.2.2.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 11.2.2.4 Parameter list (Input/Output)

Inputs: T\_ADC\_COMMON\_INIT \*adc\_common\_init\_struct - pointer to a T\_ADC\_COMMON\_INIT structure that contains the configuration information for all ADCs peripherals.

Outputs: None

#### 11.2.2.5 Return Value

None

#### 11.2.2.6 Other CSUs called by this CSU

None

#### 11.2.2.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to AdcCommonInit

##### 11.2.2.7.1 daulibstm32f4xxadc-AdcCommonInit-LLR-001

Requirement ID: H398-LLD-GWY-FNC-4832

The function shall clear the MULTI, DELAY, DMA and ADCPRE bits of ADC common control register and configure with ADC mode, prescaler, dma access mode and adc sampling delay.

* + Set Temp Register to ccr of M\_ADC.
  + Set Temp Register to (Temp Register bitwise AND with M\_CR\_CLEAR\_MASK\_ADC.
  + Set Temp Register to (Temp Register bitwise OR with (adc\_mode of adc\_common\_init\_struct bitwise OR with adc\_prescaler of adc\_common\_init\_struct bitwise OR with adc\_dma\_access\_mode of adc\_common\_init\_struct bitwise OR with adc\_two\_sampling\_delay of adc\_common\_init\_struct)).
  + Set ccr of M\_ADC to Temp Register

### 11.2.3 AdcCmd

Low Level Design Details about CSU AdcCmd will follow in the sub sections

#### 11.2.3.1 Brief Description

The function AdcCmd enables or disables the specified ADC peripheral.

#### 11.2.3.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 11.2.3.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 11.2.3.4 Parameter list (Input/Output)

Inputs: T\_ADC \*adc\_x - Where x can be 1, 2 or 3 to select the ADC peripheral.

T\_FUNCTIONAL\_STATE new\_state - New state of the adc\_x peripheral.

Outputs: T\_ADC \*adc\_x - Where x can be 1, 2 or 3 to select the ADC peripheral.

#### 11.2.3.5 Return Value

None

#### 11.2.3.6 Other CSUs called by this CSU

None

#### 11.2.3.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to AdcCmd.

##### 11.2.3.7.1 daulibstm32f4xxadc-AdcCmd-LLR-001

Requirement ID: H398-LLD-GWY-FNC-4841

The function shall set the ADON (A/D Converter ON/OFF) bit of ADC control register 2 to wake up the ADC from power down mode i.e, set cr2 of adc\_x to (cr2 of adc\_x bitwise OR with M\_ADC\_CR2\_ADON) when new\_state is equal to ENABLE

##### 11.2.3.7.2 daulibstm32f4xxadc-AdcCmd-LLR-002

Requirement ID: H398-LLD-GWY-FNC-4842

The function shall reset the ADON (A/D Converter ON/OFF) bit of ADC control register 2 to disable the selected ADC peripheral i.e, set cr2 of adc\_x to (cr2 of adc\_x bitwise AND with (negated value of M\_ADC\_CR2\_ADON)) when new\_state is equal to DISABLE.

### 11.2.4 AdcRegularChannelConfig

Low Level Design Details about CSU AdcRegularChannelConfig will follow in the sub sections.

#### 11.2.4.1 Brief Description

The function AdcRegularChannelConfig configures for the selected ADC regular channel, its corresponding rank in the sequencer and its sample time.

#### 11.2.4.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY)

#### 11.2.4.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 11.2.4.4 Parameter list (Input/Output)

Inputs: T\_ADC \*adc\_x – Where x can be 1, 2 or 3 to select the ADC peripheral.

T\_UINT8 adc\_channel - The ADC channel to configure.

T\_UINT8 rank -The rank in the regular group sequencer.

T\_UINT8 adc\_sampletime - The sample time value to be set for the selected channel.

Outputs: T\_ADC \*adc\_x - Where x can be 1, 2 or 3 to select the ADC peripheral

#### 11.2.4.5 Return Value

None

#### 11.2.4.6 Other CSUs called by this CSU

None

#### 11.2.4.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to AdcRegularChannelConfig

##### 11.2.4.7.1 daulibstm32f4xxadc-AdcRegularChannelConfig-LLR-001

Requirement ID: H398-LLD-GWY-FNC-4851

The function shall clear the old sample time for the specified ADC channel bits in ADC sample time register 1 and configure the sample time when ADC channel selected (adc\_channel) is greater than 9 i.e,

Perform as follows when M\_ADC\_CHANNEL\_9 is less than adc\_channel

* + Set Temp Register 1 to smpr1 of adc\_x.
  + Set Temp Register 2 to (M\_SMPR1\_SMP\_SET left shift by (M\_THREE multiplied by (adc\_channel subtracted by M\_TEN))).
  + Set Temp Register 1 to (Temp Register 1 bitwise AND with negated value of Temp register 2).
  + Set Temp Register 2 to (adc\_sampletime left shift by (M\_THREE multiplied by (adc\_channel subtracted by M\_TEN))).
  + Set Temp Register 1 to (Temp Register 1 bitwise OR with Temp Register 2).
  + Set smpr1 of adc\_x to Temp Register 1.

##### 11.2.4.7.2 daulibstm32f4xxadc-AdcRegularChannelConfig-LLR-002

Requirement ID: H398-LLD-GWY-FNC-4852

The function shall clear the old sample time for the specified ADC channel adc\_channel by clearing the SMPx (where x=adc\_channel) bits in ADC sample time register 2 and configure sample time when ADC channel selected is less than or equal to 9.

Perform as follows when M\_ADC\_CHANNEL\_9 is not less than adc\_channel

* + Set Temp Register 1 to smpr2 of adc\_x.
  + Set Temp Register 2 to (M\_SMPR2\_SMP\_SET left shift by (M\_THREE multiplied by adc\_channel)).
  + Set Temp Register 1 to (Temp Register 1 bitwise AND with negated value of Temp register 2).
  + Set Temp Register 2 to (adc\_sampletime left shift by (M\_THREE multiplied by adc\_channel)).
  + Set Temp Register 1 to (Temp Register 1 bitwise OR with Temp Register 2).
  + Set smpr2 of adc\_x to Temp Register 1.

##### 11.2.4.7.3 daulibstm32f4xxadc-AdcRegularChannelConfig-LLR-003

Requirement ID: H398-LLD-GWY-FNC-4853

The function shall clear the sequence bits for the selected rank in ADC regular sequence register 3 and configure the rank when rank is less than 7 i.e,

Perform as follows when M\_SEVEN is greater than rank

* + Set Temp Register 1 to sqr3 of adc\_x.
  + Set Temp Register 2 to (M\_SQR3\_SQ\_SET left shift by (M\_FIVE multiplied by (rank subtracted by M\_ONE))).
  + Set Temp Register 1 to (Temp Register 1 bitwise AND with negated value of Temp register 2).
  + Set Temp Register 2 to (adc\_channel left shift by (M\_FIVE multiplied by (rank subtracted by M\_ONE))).
  + Set Temp Register 1 to (Temp Register 1 bitwise OR with Temp Register 2).
  + Set sqr3 of adc\_x to Temp Register 1.

##### 11.2.4.7.4 daulibstm32f4xxadc-AdcRegularChannelConfig-LLR-004

Requirement ID: H398-LLD-GWY-FNC-4854

The function shall clear the sequence bits for the selected rank in ADC regular sequence register 2 and configure the rank when rank is greater than or equal to 7 and less than 13 i.e,

Perform as follows when rank is greater than or equal to M\_SEVEN AND less than M\_THIRTEEN

* + Set Temp Register 1 to sqr2 of adc\_x.
  + Set Temp Register 2 to (M\_SQR2\_SQ\_SET left shift by (M\_FIVE multiplied by (rank subtracted by M\_SEVEN))).
  + Set Temp Register 1 to (Temp Register 1 bitwise AND with negated value of Temp register 2).
  + Set Temp Register 2 to (adc\_channel left shift by (M\_FIVE multiplied by (rank subtracted by M\_SEVEN))).
  + Set Temp Register 1 to (Temp Register 1 bitwise OR with Temp Register 2).
  + Set sqr2 of adc\_x to Temp Register 1.

##### 11.2.4.7.5 daulibstm32f4xxadc-AdcRegularChannelConfig-LLR-005

Requirement ID: H398-LLD-GWY-FNC-4855

The function shall clear the sequence bits for the selected rank in ADC regular sequence register 1 and configure the rank when rank is greater than or equal 13 i.e,

Perform as follows when rank is greater than or equal to M\_THIRTEEN

* + Set Temp Register 1 to sqr1 of adc\_x.
  + Set Temp Register 2 to (M\_SQR1\_SQ\_SET left shift by (M\_FIVE multiplied by (rank subtracted by M\_THIRTEEN))).
  + Set Temp Register 1 to (Temp Register 1 bitwise AND with negated value of Temp register 2).
  + Set Temp Register 2 to (adc\_channel left shift by (M\_FIVE multiplied by (rank subtracted by M\_THIRTEEN))).
  + Set Temp Register 1 to (Temp Register 1 bitwise OR with Temp Register 2).
  + Set sqr1 of adc\_x to Temp Register 1.

### 11.2.5 AdcSoftwareStartConv

Low Level Design Details about CSU AdcSoftwareStartConv will follow in the sub sections.

#### 11.2.5.1 Brief Description

The function AdcSoftwareStartConv enables the selected ADC software start conversion of the regular channels.

#### 11.2.5.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 11.2.5.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 11.2.5.4 Parameter list (Input/Output)

Inputs: T\_ADC \*adc\_x - Where x can be 1, 2 or 3 to select the ADC peripheral.

Outputs: T\_ADC \*adc\_x - Where x can be 1, 2 or 3 to select the ADC peripheral.

#### 11.2.5.5 Return Value

None

#### 11.2.5.6 Other CSUs called by this CSU

None

#### 11.2.5.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to AdcSoftwareStartConv.

##### 11.2.5.7.1 daulibstm32f4xxadc-AdcSoftwareStartConv-LLR-001

Requirement ID: H398-LLD-GWY-FNC-4864

The function shall enable the selected ADC conversion for regular group by setting the SWSTART bit of ADC control register 2( i.e, set cr2 of adc\_x to (cr2 of adc\_x bitwise OR with M\_ADC\_CR2\_SWSTART.))

### 11.2.6 AdcDmaCmd

Low Level Design Details about CSU AdcDmaCmd will follow in the sub sections.

#### 11.2.6.1 Brief Description

The function AdcDmaCmd enables or disables the ADC DMA request.

#### 11.2.6.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 11.2.6.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 11.2.6.4 Parameter list (Input/Output)

Inputs: T\_ADC \*adc\_x - Where x can be 1, 2 or 3 to select the ADC peripheral.

T\_FUNCTIONAL\_STATE new\_state - New state of the selected ADC DMA transfer.

Outputs: T\_ADC \*adc\_x - Where x can be 1, 2 or 3 to select the ADC peripheral.

#### 11.2.6.5 Return Value

None

#### 11.2.6.6 Other CSUs called by this CSU

None

#### 11.2.6.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to AdcDmaCmd.

##### 11.2.6.7.1 daulibstm32f4xxadc-AdcDmaCmd-LLR-001

Requirement ID: H398-LLD-GWY-FNC-4873

The function shall enable the ADC DMA request by setting the DMA mode bit of ADC control register 2 when the new state of ADC peripheral is enable i.e, set cr2 of adc\_x to (cr2 of adc\_x bitwise OR with M\_ADC\_CR2\_DMA) when new\_state is equal to ENABLE.

##### 11.2.6.7.2 daulibstm32f4xxadc-AdcDmaCmd-LLR-002

Requirement ID: H398-LLD-GWY-FNC-4874

The function shall disable the ADC DMA request by resetting the DMA mode bit of ADC control register 2 when the new state of ADC peripheral is disable i.e, set cr2 of adc\_x to (cr2 of adc\_x bitwise AND with negated value of M\_ADC\_CR2\_DMA) when new\_state is equal to DISABLE.

### 11.2.7 AdcDmaReqAfterLastTransferCmd

Low Level Design Details about CSU AdcDmaReqAfterLastTransferCmd will follow in the sub sections.

#### 11.2.7.1 Brief Description

The function AdcDmaReqAfterLastTransferCmd enables or disables the ADC DMA request after last transfer (Single-ADC mode).

#### 11.2.7.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 11.2.7.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 11.2.7.4 Parameter list (Input/Output)

Inputs: T\_ADC \*adc\_x - Where x can be 1, 2 or 3 to select the ADC peripheral.

T\_FUNCTIONAL\_STATE new\_state - New state of the selected ADC DMA request after last transfer

Outputs: T\_ADC \* adc\_x - Where x can be 1, 2 or 3 to select the ADC peripheral.

#### 11.2.7.5 Return Value

None

#### 11.2.7.6 Other CSUs called by this CSU

None

#### 11.2.7.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to AdcDmaReqAfterLastTransferCmd.

##### 11.2.7.7.1 daulibstm32f4xxadc-AdcDmaReqAfterLastTransferCmd-LLR-001

Requirement ID: H398-LLD-GWY-FNC-4883

The function shall set the DDS (DMA disable selection) bit of ADC control register 2 when new state of Adc peripheral is enable i.e, set cr2 of adc\_x to (cr2 of adc\_x bitwise OR with M\_ADC\_CR2\_DDS).

##### 11.2.7.7.2 daulibstm32f4xxadc-AdcDmaReqAfterLastTransferCmd-LLR-002

Requirement ID: H398-LLD-GWY-FNC-4884

The function shall reset the DDS (DMA disable selection) bit of ADC control register 2 when new state of Adc peripheral is disable i.e, set cr2 of adc\_x to (cr2 of adc\_x bitwise AND with negated value of M\_ADC\_CR2\_DDS).

## 11.3 daulibstm32f4xxcan

This module provides firmware functions to manage the following functionalities of the Controller area network (CAN) peripheral:

* Initialization and Configuration
* CAN Frames Transmission
* CAN Frames Reception
* Interrupts and flags

### 11.3.1 CanDeInit

Low Level Design Details about CSU CanDeInit will follow in the sub sections.

#### 11.3.1.1 Brief Description

The function CanDeInit deinitializes the CAN peripheral registers to their default reset values.

#### 11.3.1.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 11.3.1.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 11.3.1.4 Parameter list (Input/Output)

Inputs: T\_CAN\_TYPEDEF \*can\_x - Where x can be 1 or 2 to select the CAN peripheral.

Outputs: None

#### 11.3.1.5 Return Valu

None

#### 11.3.1.6 Other CSUs called by this CSU

RccApb1PeriphResetCmd

#### 11.3.1.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to CanDeInit

##### 11.3.1.7.1 daulibstm32f4xxcan-CanDeInit-LLR-001

Requirement ID: H398-LLD-GWY-FNC-4894

The function shall do the following when M\_CAN1 is equal to can\_x:

a) Enable CAN1 reset state by calling the function RccApb1PeriphResetCmd with parameters M\_RCC\_APB1PERIPH\_CAN1 and ENABLE.

b) Release CAN1 from reset state by calling the function RccApb1PeriphResetCmd with parameters M\_RCC\_APB1PERIPH\_CAN1 and DISABLE.

##### 11.3.1.7.2 daulibstm32f4xxcan-CanDeInit-LLR-002

Requirement ID: H398-LLD-GWY-FNC-5780

The function shall do the following when M\_CAN1 is not equal to can\_x:

a) Enable CAN2 reset state by calling the function RccApb1PeriphResetCmd with parameters M\_RCC\_APB1PERIPH\_CAN2 and ENABLE

b) Release CAN2 from reset state by calling the function RccApb1PeriphResetCmd with parameters M\_RCC\_APB1PERIPH\_CAN2 and DISABLE.

### 11.3.2 CanInit

Low Level Design Details about CSU CanInit will follow in the sub sections.

#### 11.3.2.1 Brief Description

The function CanInit initializes the CAN peripheral according to the specified parameters in the CAN initialization structure.

#### 11.3.2.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 11.3.2.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 11.3.2.4 Parameter list (Input/Output)

Inputs: T\_CAN\_TYPEDEF\* can\_x - where x can be 1 or 2 to select the CAN peripheral.

T\_CAN\_INIT\* can\_init\_struct - pointer to a T\_CAN\_INIT structure that contains the configuration information for the CAN peripheral

Outputs: T\_CAN\_TYPEDEF\* can\_x - where x can be 1 or 2 to select the CAN peripheral.

#### 11.3.2.5 Return Value

T\_UINT8 Returns the constant which indicates initialization status.(M\_CAN\_INITSTATUS\_FAILED or M\_CAN\_INITSTATUS\_SUCCESS.)

#### 11.3.2.6 Other CSUs called by this CSU

None

#### 11.3.2.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to CanInit

##### 11.3.2.7.1 daulibstm32f4xxcan-CanInit-LLR-001

Requirement ID: H398-LLD-GWY-FNC-4903

The function shall configure CAN master control register to exit from sleep mode (set bit 1 to 0) i.e,

set mcr of can\_x to (mcr of can\_x bitwise AND with negated value of M\_CAN\_MCR\_SLEEP).

##### 11.3.2.7.2 daulibstm32f4xxcan-CanInit-LLR-002

Requirement ID: H398-LLD-GWY-FNC-4904

The function shall configure CAN master control register to request initialization (set bit 0 to 1) i.e, set mcr of can\_x to (mcr of can\_x bitwise OR with M\_CAN\_MCR\_INRQ).

##### 11.3.2.7.3 daulibstm32f4xxcan-CanInit-LLR-003

Requirement ID: H398-LLD-GWY-FNC-4905

The function shall loop till INAK bit in CAN master control register is 0 (M\_CAN\_MSR\_INAK is not equal to to (msr of can\_x bitwise AND with M\_CAN\_MSR\_INAK)) and loop count is not equal to M\_INAK\_TIMEOUT and increment the loop counter.

##### 11.3.2.7.4 daulibstm32f4xxcan-CanInit-LLR-004

Requirement ID: H398-LLD-GWY-FNC-4906

The function shall set the initialization status to M\_CAN\_INITSTATUS\_FAILED when INAK bit in CAN master control register is 0 (M\_CAN\_MSR\_INAK not equal to msr of can\_x bitwise AND with M\_CAN\_MSR\_INAK).

##### 11.3.2.7.5 daulibstm32f4xxcan-CanInit-LLR-005

Requirement ID: H398-LLD-GWY-FNC-4907

The function shall enable time triggered communication mode (set TTCM bit in master control register to 1) i.e, set mcr of can\_x to (mcr of can\_x bitwise OR with M\_CAN\_MCR\_TTCM) when can\_ttcm of can\_init\_struct is equal to ENABLE and (M\_CAN\_MSR\_INAK is equal to (msr of can\_x bitwise AND with M\_CAN\_MSR\_INAK ) (INAK bit in msr of can\_x is 1)).

##### 11.3.2.7.6 daulibstm32f4xxcan-CanInit-LLR-006

Requirement ID: H398-LLD-GWY-FNC-4908

The function shall disable time triggered communication mode (reset TTCM bit in mcr of can\_x to 0) i.e, set mcr of can\_x to (mcr of can\_x bitwise AND with negated value of M\_CAN\_MCR\_TTCM) when can\_ttcm of can\_init\_struct is not equal to ENABLE and (M\_CAN\_MSR\_INAK is equal to (msr of can\_x bitwise AND with M\_CAN\_MSR\_INAK ) (INAK bit in msr of can\_x is 1)).

##### 11.3.2.7.7 daulibstm32f4xxcan-CanInit-LLR-007

Requirement ID: H398-LLD-GWY-FNC-4909

The function shall enable automatic bus-off management (set ABOM bit in master control register to 1) i.e, set mcr of can\_x to (mcr of can\_x bitwise OR with M\_CAN\_MCR\_ABOM) when can\_abom of can\_init\_struct is ENABLE and (M\_CAN\_MSR\_INAK is equal to (msr of can\_x bitwise AND with M\_CAN\_MSR\_INAK ) (INAK bit in msr of can\_x is 1)).

##### 11.3.2.7.8 daulibstm32f4xxcan-CanInit-LLR-008

Requirement ID: H398-LLD-GWY-FNC-4910

The function shall disable automatic bus-off management (reset ABOM bit in master control register to 0) i.e., set mcr of can\_x to (mcr of can\_x bitwise AND with negated value of M\_CAN\_MCR\_ABOM) when can\_abom of can\_init\_struct is DISABLE and (M\_CAN\_MSR\_INAK is equal to (msr of can\_x bitwise AND with M\_CAN\_MSR\_INAK ) (INAK bit in msr of can\_x is 1)).

##### 11.3.2.7.9 daulibstm32f4xxcan-CanInit-LLR-009

Requirement ID: H398-LLD-GWY-FNC-4911

The function shall enable automatic wake-up mode (set AWUM bit in master control register to 1) i.e, set mcr of can\_x to (mcr of can\_x bitwise OR with M\_CAN\_MCR\_AWUM) when can\_awum of can\_init\_struct is ENABLE and (M\_CAN\_MSR\_INAK is equal to (msr of can\_x bitwise AND with M\_CAN\_MSR\_INAK ) (INAK bit in msr of can\_x is 1)).

##### 11.3.2.7.10 daulibstm32f4xxcan-CanInit-LLR-010

Requirement ID: H398-LLD-GWY-FNC-4912

The function shall disable automatic wake-up mode (set AWUM bit in master control register to 0) i.e, set mcr of can\_x to (mcr of can\_x bitwise AND with negated value of M\_CAN\_MCR\_AWUM) when can\_awum of can\_init\_struct is DISABLE and (M\_CAN\_MSR\_INAK is equal to (msr of can\_x bitwise AND with M\_CAN\_MSR\_INAK ) (INAK bit in msr of can\_x is 1)).

##### 11.3.2.7.11 daulibstm32f4xxcan-CanInit-LLR-011

Requirement ID: H398-LLD-GWY-FNC-4913

The function shall enable no automatic retransmission (set NART bit in master control register to 1) i.e, set mcr of can\_x to (mcr of can\_x bitwise OR with M\_CAN\_MCR\_NART) when can\_nart of can\_init\_struct is ENABLE and (M\_CAN\_MSR\_INAK is equal to (msr of can\_x bitwise AND with M\_CAN\_MSR\_INAK ) (INAK bit in msr of can\_x is 1)).

##### 11.3.2.7.12 daulibstm32f4xxcan-CanInit-LLR-012

Requirement ID: H398-LLD-GWY-FNC-4914

The function shall disable no automatic retransmission (set NART bit in master control register to 0) i.e, set mcr of can\_x to (mcr of can\_x bitwise AND with negated value of M\_CAN\_MCR\_NART) when can\_nart of can\_init\_struct is DISABLE and (M\_CAN\_MSR\_INAK is equal to (msr of can\_x bitwise AND with M\_CAN\_MSR\_INAK ) (INAK bit in msr of can\_x is 1)).

##### 11.3.2.7.13 daulibstm32f4xxcan-CanInit-LLR-013

Requirement ID: H398-LLD-GWY-FNC-4915

The function shall enable receive FIFO locked mode (set RFLM bit in mter control register to 1) i.e, set mcr of can\_x to (mcr of can\_x bitwise OR with M\_CAN\_MCR\_RFLM) when can\_rflm of can\_init\_struct is ENABLE and (M\_CAN\_MSR\_INAK is equal to (msr of can\_x bitwise AND with M\_CAN\_MSR\_INAK ) (INAK bit in msr of can\_x is 1)).

##### 11.3.2.7.14 daulibstm32f4xxcan-CanInit-LLR-014

Requirement ID: H398-LLD-GWY-FNC-4916

The function shall disable receive FIFO locked mode (set RFLM bit in mter control register to 0) i.e, set mcr of can\_x to (mcr of can\_x bitwise AND with negated value of M\_CAN\_MCR\_RFLM) when can\_rflm of can\_init\_struct is DISABLE and (M\_CAN\_MSR\_INAK is equal to (msr of can\_x bitwise AND with M\_CAN\_MSR\_INAK ) (INAK bit in msr of can\_x is 1)).

##### 11.3.2.7.15 daulibstm32f4xxcan-CanInit-LLR-015

Requirement ID: H398-LLD-GWY-FNC-4917

The function shall set transmit FIFO priority (set TXFP bit in master control register to 1) i.e, set mcr of can\_x to (mcr of can\_x bitwise OR with M\_CAN\_MCR\_TXFP) when can\_txfp of can\_init\_struct is ENABLE and (M\_CAN\_MSR\_INAK is equal to (msr of can\_x bitwise AND with M\_CAN\_MSR\_INAK ) (INAK bit in msr of can\_x is 1)).

##### 11.3.2.7.16 daulibstm32f4xxcan-CanInit-LLR-016

Requirement ID: H398-LLD-GWY-FNC-4918

The function shall reset transmit FIFO priority (set TXFP bit in master control register to 0) i.e, set mcr of can\_x to (mcr of can\_x bitwise AND with negated value of M\_CAN\_MCR\_TXFP) when can\_txfp of can\_init\_struct is DISABLE and (M\_CAN\_MSR\_INAK is equal to (msr of can\_x bitwise AND with M\_CAN\_MSR\_INAK ) (INAK bit in msr of can\_x is 1)).

##### 11.3.2.7.17 daulibstm32f4xxcan-CanInit-LLR-017

Requirement ID: H398-LLD-GWY-FNC-4919

The function shall set bit timing register as below when (M\_CAN\_MSR\_INAK is equal to (msr of can\_x bitwise AND with M\_CAN\_MSR\_INAK ) (INAK bit in msr of can\_x is 1)).

Set btr of can\_x to (can\_mode of can\_init\_struct left shift by M\_SHIFT\_30 bitwise OR with

can\_sjw of can\_init\_struct left shift by M\_SHIFT\_24 bitwise OR with

can\_bs1 of can\_init\_struct left shift by M\_SHIFT\_BY\_16 bitwise OR with

can\_bs2 of can\_init\_struct left shift by M\_SHIFT\_20 bitwise OR with

(can\_prescaler of can\_init\_struct minus M\_ONE))

Note:

a) set Bit 30 with Loop back mode

b) set Bit 24 and 25 with Resynchronization jump width

c) set Bit 16-19 with Time segment 1

d) set Bit 20-22 with Time segment 2

e) set Bit 0-9 with Baud rate prescaler

##### 11.3.2.7.18 daulibstm32f4xxcan-CanInit-LLR-018

Requirement ID: H398-LLD-GWY-FNC-4920

The function shall Request leave initialization (set INRQ bit of Master control register 0) i.e, set mcr of can\_x to (mcr of can\_x bitwise AND with Negation of M\_CAN\_MCR\_INRQ) when (M\_CAN\_MSR\_INAK is equal to (msr of can\_x bitwise AND with M\_CAN\_MSR\_INAK ) (INAK bit in msr of can\_x is 1)).

##### 11.3.2.7.19 daulibstm32f4xxcan-CanInit-LLR-019

Requirement ID: H398-LLD-GWY-FNC-4921

The function shall do the following when (M\_CAN\_MSR\_INAK is equal to (msr of can\_x bitwise AND with M\_CAN\_MSR\_INAK ) (INAK bit in msr of can\_x is 1)).

1. Set wait acknowledgement counter to M\_ZERO.
2. Loop till (M\_CAN\_MSR\_INAK is equal to msr of can\_X bitwise AND with M\_CAN\_MSR\_INAK) AND (M\_INAK\_TIMEOUT is not equal to wait acknowledgement counter.
3. Set the initialization status to M\_CAN\_INITSTATUS\_FAILED when INAK bit in MSR of can\_x is 1 (M\_CAN\_MSR\_INAK is equal to msr of can\_X bitwise AND with M\_CAN\_MSR\_INAK) and timeout has occurred (wait acknowledgement counter is equal to M\_INAK\_TIMEOUT).
4. Set the initialization status to M\_CAN\_INITSTATUS\_SUCCESS when INAK bit in MSR of can\_x is 0 (M\_CAN\_MSR\_INAK is not equal to msr of can\_X bitwise AND with M\_CAN\_MSR\_INAK) .

##### 11.3.2.7.20 daulibstm32f4xxcan-CanInit-LLR-020

Requirement ID: H398-LLD-GWY-FNC-4922

The function shall return the initialization status.

### 11.3.3 CanFilterInit

Low Level Design Details about CSU CanFilterInit will follow in the sub sections.

#### 11.3.3.1 Brief Description

The function CanFilterInit configures the CAN reception filter according to the specified parameters in the can\_filter\_init\_struct.

#### 11.3.3.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-DSC).

#### 11.3.3.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 11.3.3.4 Parameter list (Input/Output)

Inputs: T\_CAN\_FILTER\_INIT\* can\_filter\_init\_struct - pointer to a T\_CAN\_FILTER\_INIT structure that contains the configuration information.

Outputs: None

#### 11.3.3.5 Return Value

None

#### 11.3.3.6 Other CSUs called by this CSU

None

#### 11.3.3.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to CanFilterInit

##### 11.3.3.7.1 daulibstm32f4xxcan-CanFilterInit-LLR-001

Requirement ID: H398-LLD-GWY-FNC-4931

The function shall set (FINIT - bit 0) in filter master register to 1 i.e, set fmr of M\_CAN1 to (fmr of M\_CAN1 bitwise OR with M\_FMR\_FINIT).

##### 11.3.3.7.2 daulibstm32f4xxcan-CanFilterInit-LLR-002

Requirement ID: H398-LLD-GWY-FNC-4932

The function shall deactivate the filter x (where x = can\_filter\_number) in filter activation register by setting the corresponding bit to 0 i.e, falr of M\_CAN1 to (falr of M\_CAN1 bitwise AND with negated value of (M\_ONE left shift by can\_filter\_number of can\_filter\_init\_struct)).

##### 11.3.3.7.3 daulibstm32f4xxcan-CanFilterInit-LLR-003

Requirement ID: H398-LLD-GWY-FNC-4933

The function shall do the following when can\_filter\_scale of can\_filter\_init\_struct is equal to M\_CAN\_FILTERSCALE\_16BIT otherwise do nothing.

1. Configure fslr of M\_CAN1 for Dual 16-bit scale configuration by setting the corresponding FSCx bit to 0 (where x= can\_filter\_number of can\_filter\_init\_struct) i.e, set fslr of M\_CAN1 to (fslr of M\_CAN1 bitwise AND with (M\_ONE left shift by can\_filter\_number of can\_filter\_init\_struct).
2. set fr1 of sfilterregister of index (can\_filter\_number of can\_filter\_init\_struct) of M\_CAN1 to ((M\_MASK\_16 bitwise AND with (can\_filter\_mask\_id\_low of can\_filter\_init\_struct left shift by M\_SHIFT\_BY\_16)) bitwise OR with (M\_MASK\_16 bitwise AND with (can\_filter\_id\_low of can\_filter\_init\_struct))).
3. set fr2 of sfilterregister of index (can\_filter\_number of can\_filter\_init\_struct) of M\_CAN1 to ((M\_MASK\_16 bitwise AND with (can\_filter\_mask\_id\_high of can\_filter\_init\_struct left shift by M\_SHIFT\_BY\_16)) bitwise OR with (M\_MASK\_16 bitwise AND with (can\_filter\_id\_high of can\_filter\_init\_struct))).

##### 11.3.3.7.4 daulibstm32f4xxcan-CanFilterInit-LLR-004

Requirement ID: H398-LLD-GWY-FNC-4934

The function shall do the following when can\_filter\_scale of can\_filter\_init\_struct is equal to M\_CAN\_FILTERSCALE\_32BIT otherwise do nothing.

1. Configure fs1r of M\_CAN1 for Dual 32-bit scale configuration by setting the corresponding FSCx bit to 1 (where x=can\_filter\_init\_struct->can\_filter\_number) i.e, set fslr of M\_CAN1 to (fslr of M\_CAN1 bitwise OR with (M\_ONE left shift by can\_filter\_number of can\_filter\_init\_struct).
2. set fr1 of sfilterregister of index (can\_filter\_number of can\_filter\_init\_struct) of M\_CAN1 to ((M\_MASK\_16 bitwise AND with (can\_filter\_id\_high of can\_filter\_init\_struct left shift by M\_SHIFT\_BY\_16)) bitwise OR with (M\_MASK\_16 bitwise AND with (can\_filter\_id\_low of can\_filter\_init\_struct))).
3. set fr2 of sfilterregister of index (can\_filter\_number of can\_filter\_init\_struct) of M\_CAN1 to ((M\_MASK\_16 bitwise AND with (can\_filter\_mask\_id\_high of can\_filter\_init\_struct left shift by M\_SHIFT\_BY\_16)) bitwise OR with (M\_MASK\_16 bitwise AND with (can\_filter\_mask\_id\_low of can\_filter\_init\_struct))).

##### 11.3.3.7.5 daulibstm32f4xxcan-CanFilterInit-LLR-005

Requirement ID: H398-LLD-GWY-FNC-4935

The function shall set the FBMx bit (where x=can\_filter\_init\_struct->can\_filter\_number) in filter mode register to 0 i.e, set fmlr of M\_CAN1 to (fmlr of M\_CAN1 bitwise AND with negated value (M\_ONE left shift by can\_filter\_number of can\_filter\_init\_struct)) when can\_filter\_mode of can\_filter\_init\_struct is equal to M\_CAN\_FILTERMODE\_IDMASK.

##### 11.3.3.7.6 daulibstm32f4xxcan-CanFilterInit-LLR-006

Requirement ID: H398-LLD-GWY-FNC-4936

The function shall set the FBMx bit (where x=can\_filter\_init\_struct->can\_filter\_number) in filter mode register to 1 i.e, set fmlr of M\_CAN1 to (fmlr of M\_CAN1 bitwise OR with (M\_ONE left shift by can\_filter\_number of can\_filter\_init\_struct)) when can\_filter\_mode of can\_filter\_init\_struct is not equal to M\_CAN\_FILTERMODE\_IDMASK.

##### 11.3.3.7.7 daulibstm32f4xxcan-CanFilterInit-LLR-007

Requirement ID: H398-LLD-GWY-FNC-4937

The function shall set the FFAx bit (where x=can\_filter\_init\_struct->can\_filter\_number) in CAN filter FIFO assignment register to 0 i.e, set ffalr of M\_CAN1 to (ffalr of M\_CAN1bitwise AND with negated value of (M\_ONE left shift by can\_filter\_number of can\_filter\_init\_struct)) when can\_filter\_fifo\_assignment of can\_filter\_init\_struct is equal to M\_CAN\_FILTER\_FIFO0 otherwise do nothing.

##### 11.3.3.7.8 daulibstm32f4xxcan-CanFilterInit-LLR-008

Requirement ID: H398-LLD-GWY-FNC-4938

The function shall set the FFAx bit (where x=can\_filter\_init\_struct->can\_filter\_number) in CAN filter FIFO assignment register to 1 i.e, set ffalr of M\_CAN1 to (ffalr of M\_CAN1bitwise OR with (M\_ONE left shift by can\_filter\_number of can\_filter\_init\_struct)) when can\_filter\_fifo\_assignment of can\_filter\_init\_struct is equal to M\_CAN\_FILTER\_FIFO1 otherwise do nothing.

##### 11.3.3.7.9 daulibstm32f4xxcan-CanFilterInit-LLR-009

Requirement ID: H398-LLD-GWY-FNC-4939

The function shall set the FACTx bit (where x=can\_filter\_init\_struct->can\_filter\_number) in CAN filter activation register to 1 i.e, set falr of M\_CAN1 to (falr of M\_CAN1 bitwise OR with (M\_ONE left shift by can\_filter\_number of can\_filter\_init\_struct) when can\_filter\_activation of can\_filter\_init\_struct is ENABLE otherwise do nothing.

##### 11.3.3.7.10 daulibstm32f4xxcan-CanFilterInit-LLR-010

Requirement ID: H398-LLD-GWY-FNC-4940

The function shall set the FINIT bit to 0 (active filters mode) of filter master register i.e, set fmr of M\_CAN1 to (fmr of M\_CAN1 bitwise AND with negated value of M\_FMR\_FINIT).

### 11.3.4 CanTransmit

Low Level Design Details about CSU CanTransmit will follow in the sub sections.

#### 11.3.4.1 Brief Description

The function CanTransmit initiates and transmits a CAN frame message.

#### 11.3.4.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-DSC).

#### 11.3.4.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 11.3.4.4 Parameter list (Input/Output)

Inputs: T\_CAN\_FILTER\_INIT\* can\_x - Where x can be 1 or 2 to select the CAN peripheral.

T\_CAN\_TX\_MSG\* tx\_message - Pointer to a structure which contains CAN Id, CAN dlc and CAN data.

Outputs: T\_CAN\_FILTER\_INIT\* can\_x - Where x can be 1 or 2 to select the CAN peripheral.

T\_CAN\_TX\_MSG\* tx\_message - Pointer to a structure which contains CAN Id, CAN dlc and CAN data.

#### 11.3.4.5 Return Value

T\_UINT8 - The number of the mailbox that is used for transmission to

M\_CAN\_TXSTATUS\_NOMAILBOX if there is no empty mailbox.

#### 11.3.4.6 Other CSUs called by this CSU

None

#### 11.3.4.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to CanTransmit

##### 11.3.4.7.1 daulibstm32f4xxcan-CanTransmit-LLR-001

Requirement ID: H398-LLD-GWY-FNC-4949

The function shall set transmit mailbox to M\_ZERO when TME0 bit of transmit status register is set i.e, (M\_CAN\_TSR\_TME0 is equal to (tsr of can\_x bitwise AND with M\_CAN\_TSR\_TME0)).

##### 11.3.4.7.2 daulibstm32f4xxcan-CanTransmit-LLR-002

Requirement ID: H398-LLD-GWY-FNC-4950

The function shall set transmit mailbox to M\_ONE when TME0 bit of transmit status register is set to 0 i.e, (M\_CAN\_TSR\_TME0 is not equal to (tsr of can\_x bitwise AND with M\_CAN\_TSR\_TME0)) AND (TME1 bit of transmit status register is set to 1 i.e, (M\_CAN\_TSR\_TME1 is equal to (tsr of can\_x bitwise AND with M\_CAN\_TSR\_TME1))).

##### 11.3.4.7.3 daulibstm32f4xxcan-CanTransmit-LLR-003

Requirement ID: H398-LLD-GWY-FNC-4951

The function shall set transmit mailbox to M\_TWO when TME0 bit of transmit status register is set to 0 i.e, (M\_CAN\_TSR\_TME0 is not equal to (tsr of can\_x bitwise AND with M\_CAN\_TSR\_TME0)) AND (TME1 bit of transmit status register is set to 0 i.e, (M\_CAN\_TSR\_TME1 is not equal to (tsr of can\_x bitwise AND with M\_CAN\_TSR\_TME1))) AND (TME2 bit of transmit status register is set to 1 i.e, (M\_CAN\_TSR\_TME2 is equal to (tsr of can\_x bitwise AND with M\_CAN\_TSR\_TME2))).

##### 11.3.4.7.4 daulibstm32f4xxcan-CanTransmit-LLR-004

Requirement ID: H398-LLD-GWY-FNC-4952

The function shall set transmit mailbox to M\_CAN\_TXSTATUS\_NOMAILBOX when TME0 bit of transmit status register is set to 0 i.e, (M\_CAN\_TSR\_TME0 is not equal to (tsr of can\_x bitwise AND with M\_CAN\_TSR\_TME0)) AND (TME1 bit of transmit status register is set to 0 i.e, (M\_CAN\_TSR\_TME1 is not equal to (tsr of can\_x bitwise AND with M\_CAN\_TSR\_TME1))) AND (TME2 bit of transmit status register is set to 0 i.e, (M\_CAN\_TSR\_TME2 is not equal to (tsr of can\_x bitwise AND with M\_CAN\_TSR\_TME2))).

##### 11.3.4.7.5 daulibstm32f4xxcan-CanTransmit-LLR-005

Requirement ID: H398-LLD-GWY-FNC-4953

The function shall reset TXRQ bit of TX mailbox identifier register i.e, set tir of stxmailbox with index transmit mailbox of can\_x to (set tir of stxmailbox with index transmit mailbox of can\_x bitwise AND with M\_TMIDXR\_TXRQ) when any of the transmit mailbox (TME0, TME1 or TME2) is empty i.e, M\_CAN\_TXSTATUS\_NOMAILBOX is not equal to transmit mailbox otherwise no nothing.

##### 11.3.4.7.6 daulibstm32f4xxcan-CanTransmit-LLR-006

Requirement ID: H398-LLD-GWY-FNC-4954

The function shall configure TX mailbox identifier register with Standard identifier and RTR data i.e, set tir of stxmailbox with index as transmit mailbox of can\_x to (tir of stxmailbox with index as transmit mailbox of can\_x bitwise OR with ((std\_id of tx\_message left shift by M\_SHIFT\_21) bitwsie OR with (rtr of tx\_message))) when (any of the transmit mailbox (TME0, TME1 or TME2) is empty i.e, M\_CAN\_TXSTATUS\_NOMAILBOX is not equal to transmit mailbox) AND (M\_CAN\_ID\_STD is equal to ide of tx\_message).

##### 11.3.4.7.7 daulibstm32f4xxcan-CanTransmit-LLR-007

Requirement ID: H398-LLD-GWY-FNC-4955

The function shall configure TX mailbox identifier register with Extended identifier, identifier extension and RTR data i.e, set tir of stxmailbox with index as transmit mailbox of can\_x to (tir of stxmailbox with index as transmit mailbox of can\_x bitwise OR with ((ext\_id of tx\_message left shift by M\_SHIFT\_3) bitwsie OR with (ide of tx\_message) bitwise OR with (rtr of tx\_message))) when (any of the transmit mailbox (TME0, TME1 or TME2) is empty i.e, M\_CAN\_TXSTATUS\_NOMAILBOX is not equal to transmit mailbox) AND (M\_CAN\_ID\_STD is not equal to ide of tx\_message).

##### 11.3.4.7.8 daulibstm32f4xxcan-CanTransmit-LLR-008

Requirement ID: H398-LLD-GWY-FNC-4956

The function shall clear the tdtr bits from 4-31 and configure with Data length code of CAN mailbox data length control and time stamp register i.e,

1. set dlc of tx\_message to (dlc of tx\_message bitwise AND with M\_MASK\_4) when (any of the transmit mailbox (TME0, TME1 or TME2) is empty i.e, M\_CAN\_TXSTATUS\_NOMAILBOX is not equal to transmit mailbox).
2. Set tdtr of stxmailbox with index as transmit mailbox of can\_x to (tdtr of stxmailbox with index as transmit mailbox of can\_x bitwise AND with M\_MASK\_28.
3. Set tdtr of stxmailbox with index as transmit mailbox of can\_x to (tdtr of stxmailbox with index as transmit mailbox of can\_x bitwise OR with dlc of tx\_message.

when (any of the transmit mailbox (TME0, TME1 or TME2) is empty i.e, M\_CAN\_TXSTATUS\_NOMAILBOX is not equal to transmit mailbox)

##### 11.3.4.7.9 daulibstm32f4xxcan-CanTransmit-LLR-009

Requirement ID: H398-LLD-GWY-FNC-4957

The function shall configure CAN mailbox data high register with data field i.e,

1. set tdlr with index as transmit mailbox of can\_x to ((data with index M\_THREE of tx\_message left shift by M\_SHIFT\_24) bitwise OR with ((data with index M\_TWO of tx\_message left shift by M\_SHIFT\_BY\_16) bitwise OR with ((data with index M\_ONE of tx\_message left shift by M\_SHIFT\_BY\_8) bitwise OR with ((data with index M\_ZERO of tx\_message)).
2. set tdhr with index as transmit mailbox of can\_x to ((data with index M\_SEVEN of tx\_message left shift by M\_SHIFT\_24) bitwise OR with ((data with index M\_SIX of tx\_message left shift by M\_SHIFT\_BY\_16) bitwise OR with ((data with index M\_FIVE of tx\_message left shift by M\_SHIFT\_BY\_8) bitwise OR with ((data with index M\_FOUR of tx\_message)).

when (any of the transmit mailbox (TME0, TME1 or TME2) is empty i.e, M\_CAN\_TXSTATUS\_NOMAILBOX is not equal to transmit mailbox)

##### 11.3.4.7.10 daulibstm32f4xxcan-CanTransmit-LLR-010

Requirement ID: H398-LLD-GWY-FNC-4958

The function shall Request for transmission (set TXRQ bit in CAN TX mailbox identifier register) i.e, set tir of stxmailbox with index transmit mailbox of can\_x to (tir of stxmailbox with index transmit mailbox of can\_x bitwise OR with M\_TMIDXR\_TXRQ) when (any of the transmit mailbox (TME0, TME1 or TME2) is empty i.e, M\_CAN\_TXSTATUS\_NOMAILBOX is not equal to transmit mailbox).

##### 11.3.4.7.11 daulibstm32f4xxcan-CanTransmit-LLR-011

Requirement ID: H398-LLD-GWY-FNC-4959

The function shall return the transmit mailbox.

### 11.3.5 CanReceive

Low Level Design Details about CSU CanReceive will follow in the sub sections.

#### 11.3.5.1 Brief Description

The function CanReceive Receives a correct CAN frame.

#### 11.3.5.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-DSC).

#### 11.3.5.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 11.3.5.4 Parameter list (Input/Output)

Inputs: T\_CAN\_TYPEDEF\* can\_x - Where x can be 1 or 2 to select the CAN peripheral.

T\_UINT8 fifo\_number: Receive FIFO number, M\_CAN\_FIFO0 or M\_CAN\_FIFO1.

Outputs: T\_CAN\_RX\_MSG\* rx\_message - Pointer to a structure receive frame which contains CAN Id, CAN DLC, CAN data and FMI number.

T\_CAN\_TYPEDEF\* can\_x - Where x can be 1 or 2 to select the CAN peripheral.

#### 11.3.5.5 Return Value

None

#### 11.3.5.6 Other CSUs called by this CSU

None

#### 11.3.5.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to CanReceive

##### 11.3.5.7.1 daulibstm32f4xxcan-CanReceive-LLR-001

Requirement ID: H398-LLD-GWY-FNC-4968

The function shall extract the IDE (identifier extension) bit value in receive FIFO mailbox identifier register i.e, set ide of rx\_message to (M\_FOUR bitwise AND with rir of sfifomailbox in fifo\_number index of can\_x).

##### 11.3.5.7.2 daulibstm32f4xxcan-CanReceive-LLR-002

Requirement ID: H398-LLD-GWY-FNC-4969

The function shall extract Standard identifier value in receive FIFO mailbox identifier register i.e, set std\_id of rx\_message to (M\_MASK\_11 bitwise AND with (rir of sfifomailbox with index fifo number of can\_x right shift by M\_SHIFT\_21)) when ide of rx\_message is equal to M\_CAN\_ID\_STD (standard ID).

##### 11.3.5.7.3 daulibstm32f4xxcan-CanReceive-LLR-003

Requirement ID: H398-LLD-GWY-FNC-4970

The function shall extract extended identifier value in receive FIFO mailbox identifier register i.e, set ext\_id of rx\_message to (M\_MASK\_29 bitwsie AND with (rir of sfifomailbox with index fifo number of can\_x right shift by M\_SHIFT\_3 ) if ide of rx\_message is not equal to M\_CAN\_ID\_STD (Extended ID).

##### 11.3.5.7.4 daulibstm32f4xxcan-CanReceive-LLR-004

Requirement ID: H398-LLD-GWY-FNC-4971

The function shall extract remote transmission request (frame type) value in receive FIFO mailbox identifier register i.e, set rtr of rx\_message to (M\_TWO bitwise AND with rir of sfifomailbox with index fifo number of can\_x).

##### 11.3.5.7.5 daulibstm32f4xxcan-CanReceive-LLR-005

Requirement ID: H398-LLD-GWY-FNC-4972

The function shall extract dlc value in receive FIFO mailbox identifier register i.e, set dlc of rx\_message to (M\_MASK\_4 bitwise AND with rdtr of sfifomailbox with index fifo number of can\_x).

##### 11.3.5.7.6 daulibstm32f4xxcan-CanReceive-LLR-006

Requirement ID: H398-LLD-GWY-FNC-4973

The function shall extract FMI (Filter match index) in receive FIFO mailbox identifier register i.e, set fmi of rx\_message to (M\_MASK\_8 bitwise AND with (rdtr of sfifomailbox with index fifo number of can\_x right shift by M\_SHIFT\_8)).

##### 11.3.5.7.7 daulibstm32f4xxcan-CanReceive-LLR-007

Requirement ID: H398-LLD-GWY-FNC-4974

The function shall extract data field data in receive FIFO mailbox identifier register i.e,

1. Set data of index M\_ZERO of rx\_message to (M\_MASK\_8 bitwise AND with rdlr of sfifomailbox with index fifo number of can\_x).
2. Set data of index M\_ONE of rx\_message to (M\_MASK\_8 bitwise AND with rdlr of sfifomailbox with index fifo number of can\_x right shift by M\_SHIFT\_BY\_8).
3. Set data of index M\_TWO of rx\_message to (M\_MASK\_8 bitwise AND with rdlr of sfifomailbox with index fifo number of can\_x right shift by M\_SHIFT\_BY\_16).
4. Set data of index M\_THREE of rx\_message to (M\_MASK\_8 bitwise AND with rdlr of sfifomailbox with index fifo number of can\_x right shift by M\_SHIFT\_BY\_24).
5. Set data of index M\_FOUR of rx\_message to (M\_MASK\_8 bitwise AND with rdhr of sfifomailbox with index fifo number of can\_x).
6. Set data of index M\_FIVE of rx\_message to (M\_MASK\_8 bitwise AND with rdhr of sfifomailbox with index fifo number of can\_x right shift by M\_SHIFT\_BY\_8).
7. Set data of index M\_SIX of rx\_message to (M\_MASK\_8 bitwise AND with rdhr of sfifomailbox with index fifo number of can\_x right shift by M\_SHIFT\_BY\_16).
8. Set data of index M\_SEVEN of rx\_message to (M\_MASK\_8 bitwise AND with rdhr of sfifomailbox with index fifo number of can\_x right shift by M\_SHIFT\_BY\_24).

##### 11.3.5.7.8 daulibstm32f4xxcan-CanReceive-LLR-008

Requirement ID: H398-LLD-GWY-FNC-4975

The function shall release FIFO 0 (set RFOM0 bit in receive FIFO 0 register (rf0r)) i.e, set rf0r of can\_x to (rf0r of can\_x bitwise OR with M\_CAN\_RF0R\_RFOM0) when M\_CAN\_FIFO0 is equal to fifo number.

##### 11.3.5.7.9 daulibstm32f4xxcan-CanReceive-LLR-009

Requirement ID: H398-LLD-GWY-FNC-4976

The function shall release FIFO 1 (set RFOM1 bit in receive FIFO 1 register (rf1r)) i.e, set rflr of can\_x to (rflr of can\_x bitwise OR with M\_CAN\_RF1R\_RFOM1) when M\_CAN\_FIFO0 is not equal to fifo number.

### 11.3.6 CanItConfig

Low Level Design Details about CSU CanItConfig will follow in the sub sections.

#### 11.3.6.1 Brief Description

The function CanItConfig enables or disables the specified CANx interrupts.

#### 11.3.6.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-DSC).

#### 11.3.6.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 11.3.6.4 Parameter list (Input/Output)

Inputs: T\_CAN\_TYPEDEF\* can\_x - Where x can be 1 or 2 to select the CAN peripheral.

T\_UINT32 can\_it - Specifies the CAN interrupt sources to be enabled or disabled.

T\_FUNCTIONAL\_STATE new\_state - New state of the CAN interrupts.

Outputs: T\_CAN\_TYPEDEF\* can\_x - Where x can be 1 or 2 to select the CAN peripheral.

#### 11.3.6.5 Return Value

None

#### 11.3.6.6 Other CSUs called by this CSU

None

#### 11.3.6.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to CanItConfig.

##### 11.3.6.7.1 daulibstm32f4xxcan-CanItConfig-LLR-001

Requirement ID: H398-LLD-GWY-FNC-4985

The function shall configure interrupt enable register (ier) to enable the selected can interrupt i.e, set ier of can\_x to (ier of can\_x bitwise OR with can\_it) when the new\_state is ENABLE.

##### 11.3.6.7.2 daulibstm32f4xxcan-CanItConfig-LLR-002

Requirement ID: H398-LLD-GWY-FNC-4986

The function shall configure interrupt enable register (ier) to disable the selected can interrupt i.e, set ier of can\_x to (ier of can\_x bitwise AND with negated value of can\_it when the new\_state is DISABLE.

## 11.4 daulibstm32f4xxcrc

The module provides all the CRC firmware functions.

### 11.4.1 CrcResetDr

Low Level Design Details about CSU CrcResetDr will follow in the sub sections.

#### 11.4.1.1 Brief Description

The function CrcResetDr resets the CRC control register (cr).

#### 11.4.1.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 11.4.1.3 List of global variables accessed and modified

None

#### 11.4.1.4 Parameter list (Input/Output)

Inputs: None.

Outputs: None.

#### 11.4.1.5 Return Value

None

#### 11.4.1.6 Other CSUs called by this CSU

None

#### 11.4.1.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to CrcResetDr.

##### 11.4.1.7.1 daulibstm32f4xxcrc-CrcResetDr-LLR-001

Requirement ID: H398-LLD-GWY-FNC-4996

The function shall set the RESET bit of CRC Control register (cr) to reset the CRC calculation unit i.e, set cr of M\_CRC to M\_CRC\_CR\_RESET.

### 11.4.2 CrcCalcBlockCrc

Low Level Design Details about CSU CrcCalcBlockCrc will follow in the sub sections.

#### 11.4.2.1 Brief Description

The function computes the 32-bit M\_CRC of a given buffer of data word (32-bit).

#### 11.4.2.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 11.4.2.3 List of global variables accessed and modified

None

#### 11.4.2.4 Parameter list (Input/Output)

Inputs: T\_UINT32 pbuffer: buffer containing the data to be computed.

T\_UINT32 buffer\_length: length of the buffer to be computed.

Outputs: None.

#### 11.4.2.5 Return Value

T\_UINT32 - returns 32-bit CRC.

#### 11.4.2.6 Other CSUs called by this CSU

None

#### 11.4.2.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to CrcCalcBlockCrc.

##### 11.4.2.7.1 daulibstm32f4xxcrc-CrcCalcBlockCrc-LLR-001

Requirement ID: H398-LLD-GWY-FNC-5005

The function shall copy the data from pbuffer with index as loop counter index to CRC data register (dr of M\_CRC) for loop counter index from M\_ZERO to buffer\_length and returns the calculated crc (dr of M\_CRC).

## 11.5 daulibstm32f4xxdac

This module provides firmware functions to manage the following functionalities of the Digital-to-Analog Converter (DAC) peripheral

* DAC channels configuration
* DMA management

### 11.5.1 DacInit

Low Level Design Details about CSU DacInit will follow in the sub sections.

#### 11.5.1.1 Brief Description

The function DacInit initializes the DAC peripheral according to the specified parameters in the DAC initialization structure.

#### 11.5.1.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 11.5.1.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 11.5.1.4 Parameter list (Input/Output)

Inputs: T\_UINT32 dac\_channel - Selected DAC channel.

T\_DAC\_INIT\* dac\_init\_struct - Pointer to a T\_DAC\_INIT structure that contains the configuration information for the specified DAC channel.

Outputs: None

#### 11.5.1.5 Return Value

None

#### 11.5.1.6 Other CSUs called by this CSU

None

#### 11.5.1.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to DacInit.

##### 11.5.1.7.1 daulibstm32f4xxdac-DacInit-LLR-001

Requirement ID: H398-LLD-GWY-FNC-5015

The function shall clear the DAC control register (BOFF, TEN, TSEL, WAVE and MAMP bits) for the selected dac\_channel i.e,

1. set Temp register 1 to cr of M\_DAC
2. set Temp register 1 to (Temp register 1 bitwise AND with negated value of (M\_CR\_CLEAR\_MASK\_DAC left shift by dac\_channel)).

##### 11.5.1.7.2 daulibstm32f4xxdac-DacInit-LLR-002

Requirement ID: H398-LLD-GWY-FNC-5016

The function shall configure the buffer output (BOFFx bit), trigger wave generation (TSELx and TENx bits),mask/Amplitude (MAMPx bits) for the selected wave generation (WAVEx bits) for the selected DAC channel, calculate control register value depending on dac\_channel and update the DAC control register, i.e,

1. Set Temp register 2 to (dac\_trigger value of dac\_init\_struct bitwise OR with dac\_wavegeneration of dac\_init\_struct bitwise OR with dac\_lfsrunmask\_triangleamplitude of dac\_init\_struct bitwise OR with dac\_outputbuffer of dac\_init\_struct).
2. Set Temp register 1 to (Temp register 1 bitwise OR with (Temp register 2 left shift by dac\_channel)).
3. Set cr of M\_DAC to Temp register 1.

### 11.5.2 DacStructInit

Low Level Design Details about CSU DacStructInit will follow in the sub sections.

#### 11.5.2.1 Brief Description

The function DacStructInit fills each DAC initialization structure member with its default value.

#### 11.5.2.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 11.5.2.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 11.5.2.4 Parameter list (Input/Output)

Inputs: None.

Outputs: T\_DAC\_INIT\* dac\_init\_struct - pointer to a T\_DAC\_INIT structure which will be initialized.

#### 11.5.2.5 Return Value

None

#### 11.5.2.6 Other CSUs called by this CSU

None

#### 11.5.2.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to DacStructInit.

##### 11.5.2.7.1 daulibstm32f4xxdac-DacStructInit-LLR-001

Requirement ID: H398-LLD-GWY-FNC-5025

The function shall reset the DAC Initialization structure values i.e,

1. Set dac\_trigger of dac\_init\_struct to M\_DAC\_TRIGGER\_NONE.
2. Set dac\_wavegeneration of dac\_init\_struct to M\_DAC\_WAVEGENERATION\_NONE.
3. Set dac\_lfsrunmask\_triangleamplitude of dac\_init\_struct to M\_DAC\_LFSRUNMASK\_BIT0.
4. Set dac\_outputbuffer of dac\_init\_struct to M\_DAC\_OUTPUTBUFFER\_ENABLE.

### 11.5.3 DacCmd

Low Level Design Details about CSU DacCmd will follow in the sub sections.

#### 11.5.3.1 Brief Description

The function DacCmd enables or disables the specified DAC channel.

#### 11.5.3.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 11.5.3.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 11.5.3.4 Parameter list (Input/Output)

Inputs: T\_UINT32 dac\_channel - Selected DAC channel.

T\_FUNCTIONAL\_STATE new\_state - New state of the DAC channel.

Outputs: None

#### 11.5.3.5 Return Value

None

#### 11.5.3.6 Other CSUs called by this CSU

None

#### 11.5.3.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to DacCmd.

##### 11.5.3.7.1 daulibstm32f4xxdac-DacCmd-LLR-001

Requirement ID: H398-LLD-GWY-FNC-5034

The function shall enable the selected DAC channel when new\_state is ENABLE by performing the below operation.

Set cr of M\_DAC to (cr of M\_DAC bitwise OR with (M\_DAC\_CR\_EN1 left shift by dac\_channel)).

##### 11.5.3.7.2 daulibstm32f4xxdac-DacCmd-LLR-002

Requirement ID: H398-LLD-GWY-FNC-5035

The function shall disable the selected DAC channel when new\_state is DISABLE by performing the below operation.

Set cr of M\_DAC to (cr of M\_DAC bitwise AND with negated value of (M\_DAC\_CR\_EN1 left shift by dac\_channel)).

## 11.6 daulibstm32f4xxdma

This module provides firmware functions to manage the following functionalities of the Direct Memory Access controller (DMA):

* Initialization and Configuration
* Interrupts and flags management

### 11.6.1 DmaDeInit

Low Level Design Details about CSU DmaDeInit will follow in the sub sections.

#### 11.6.1.1 Brief Description

The function DmaDeInit deinitializes the DMAy STREAMx registers to their default reset values.

#### 11.6.1.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 11.6.1.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 11.6.1.4 Parameter list (Input/Output)

Inputs: T\_DMA\_STREAM\*dmay\_streamx - DMA Stream, where y can be 1 or 2 to select the DMA and x can be 0 to 7 to select the DMA Stream.

Outputs: T\_DMA\_STREAM\* dmay\_streamx - DMA Stream, where y can be 1 or 2 to select the DMA and x can be 0 to 7 to select the DMA Stream.

#### 11.6.1.5 Return Value

None

#### 11.6.1.6 Other CSUs called by this CSU

None

#### 11.6.1.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to DmaDeInit.

##### 11.6.1.7.1 daulibstm32f4xxdma-DmaDeInit-LLR-001

Requirement ID: H398-LLD-GWY-FNC-5045

The function shall disable the selected DMAy Streamx by resetting EN bit in DMA stream x configuration register i.e set cr of dmay\_streamx to (cr of dmay\_streamx Bitwise AND with Negated value of M\_DMA\_SXCR\_EN).

##### 11.6.1.7.2 daulibstm32f4xxdma-DmaDeInit-LLR-002

Requirement ID: H398-LLD-GWY-FNC-5046

The function shall:

a) Set cr of dmay\_streamx to M\_ZERO.

b) Set ndtr of dmay\_streamx to M\_ZERO.

c) Set par of dmay\_streamx to M\_ZERO.

d) Set m0ar of dmay\_streamx to M\_ZERO.

e) Set m1ar of dmay\_streamx to M\_ZERO.

f) Set fcr of dmay\_streamx to M\_DMA\_FCR\_REG\_RESET.

##### 11.6.1.7.3 daulibstm32f4xxdma-DmaDeInit-LLR-003

Requirement ID: H398-LLD-GWY-FNC-5047

The function shall set lifcr of M\_DMA1 to M\_DMA\_STREAM0\_IT\_MASK when dmay\_streamx is M\_DMA1\_STREAM0.

##### 11.6.1.7.4 daulibstm32f4xxdma-DmaDeInit-LLR-004

Requirement ID: H398-LLD-GWY-FNC-5048

The function shall clear the interrupt pending bits for DMA1 Stream1 by setting the CFEIF1, CDMEIF1, CTEIF1, CHTIF1 and CTCIF1 bits of DMA1 low interrupt flag clear register (i.e set lifcr of M\_DMA1 to M\_DMA\_STREAM1\_IT\_MASK) when dmay\_streamx is M\_DMA1\_STREAM1.

##### 11.6.1.7.5 daulibstm32f4xxdma-DmaDeInit-LLR-005

Requirement ID: H398-LLD-GWY-FNC-5049

The function shall clear the interrupt pending bits for DMA1 Stream2 by setting the CFEIF2, CDMEIF2, CTEIF2, CHTIF2 and CTCIF2 bits of DMA1 low interrupt flag clear register (i.e set lifcr of M\_DMA1 to M\_DMA\_STREAM2\_IT\_MASK) when dmay\_streamx is M\_DMA1\_STREAM2.

11.6.1.7.6 daulibstm32f4xxdma-DmaDeInit-LLR-006

Requirement ID: H398-LLD-GWY-FNC-5050

The function shall clear the interrupt pending bits for DMA1 Stream3 by setting the CFEIF3, CDMEIF3, CTEIF3, CHTIF3 and CTCIF3 bits of DMA1 low interrupt flag clear register (i.e set lifcr of M\_DMA1 to M\_DMA\_STREAM3\_IT\_MASK) when dmay\_streamx is M\_DMA1\_STREAM3.

##### 11.6.1.7.7 daulibstm32f4xxdma-DmaDeInit-LLR-007

Requirement ID: H398-LLD-GWY-FNC-5051

The function shall clear the interrupt pending bits for DMA1 Stream4 by setting the CFEIF4, CDMEIF4, CTEIF4, CHTIF4 and CTCIF4 bits of DMA1 high interrupt flag clear register (i.e set hifcr of M\_DMA1 to M\_DMA\_STREAM4\_IT\_MASK) when dmay\_streamx is M\_DMA1\_STREAM4.

##### 11.6.1.7.8 daulibstm32f4xxdma-DmaDeInit-LLR-008

Requirement ID: H398-LLD-GWY-FNC-5052

The function shall clear the interrupt pending bits for DMA1 Stream5 by setting the CFEIF5, CDMEIF5, CTEIF5, CHTIF5 and CTCIF5 bits of DMA1 high interrupt flag clear register (i.e set hifcr of M\_DMA1 to M\_DMA\_STREAM5\_IT\_MASK) when dmay\_streamx is M\_DMA1\_STREAM5.

##### 11.6.1.7.9 daulibstm32f4xxdma-DmaDeInit-LLR-009

Requirement ID: H398-LLD-GWY-FNC-5053

The function shall clear the interrupt pending bits for DMA1 Stream6 by setting the CFEIF6, CDMEIF6, CTEIF6, CHTIF6 and CTCIF6 bits of DMA1 high interrupt flag clear register (i.e set hifcr of M\_DMA1 to M\_DMA\_STREAM6\_IT\_MASK) when dmay\_streamx is M\_DMA1\_STREAM6.

##### 11.6.1.7.10 daulibstm32f4xxdma-DmaDeInit-LLR-010

Requirement ID: H398-LLD-GWY-FNC-5054

The function shall clear the interrupt pending bits for DMA1 Stream7 by setting the CFEIF7, CDMEIF7, CTEIF7, CHTIF7 and CTCIF7 bits of DMA1 high interrupt flag clear register (i.e set hifcr of M\_DMA1 to M\_DMA\_STREAM7\_IT\_MASK) when dmay\_streamx is M\_DMA1\_STREAM7.

##### 11.6.1.7.11 daulibstm32f4xxdma-DmaDeInit-LLR-011

Requirement ID: H398-LLD-GWY-FNC-5055

The function shall clear the interrupt pending bits for DMA2 Stream0 by setting the CFEIF0, CDMEIF0, CTEIF0, CHTIF0 and CTCIF0 bits of DMA2 low interrupt flag clear register (i.e set lifcr of M\_DMA2 to M\_DMA\_STREAM0\_IT\_MASK) when dmay\_streamx is M\_DMA2\_STREAM0.

##### 11.6.1.7.12 daulibstm32f4xxdma-DmaDeInit-LLR-012

Requirement ID: H398-LLD-GWY-FNC-5056

The function shall clear the interrupt pending bits for DMA2 Stream1 by setting the CFEIF1, CDMEIF1, CTEIF1, CHTIF1 and CTCIF1 bits of DMA2 low interrupt flag clear register (i.e set lifcr of M\_DMA2 to M\_DMA\_STREAM1\_IT\_MASK) when dmay\_streamx is M\_DMA2\_STREAM1.

##### 11.6.1.7.13 daulibstm32f4xxdma-DmaDeInit-LLR-013

Requirement ID: H398-LLD-GWY-FNC-5057

The function shall clear the interrupt pending bits for DMA2 Stream2 by setting the CFEIF2, CDMEIF2, CTEIF2, CHTIF2 and CTCIF2 bits of DMA2 low interrupt flag clear register (i.e set lifcr of M\_DMA2 to M\_DMA\_STREAM2\_IT\_MASK) when dmay\_streamx is M\_DMA2\_STREAM2.

##### 11.6.1.7.14 daulibstm32f4xxdma-DmaDeInit-LLR-014

Requirement ID: H398-LLD-GWY-FNC-5058

The function shall clear the interrupt pending bits for DMA2 Stream3 by setting the CFEIF3, CDMEIF3, CTEIF3, CHTIF3 and CTCIF3 bits of DMA2 low interrupt flag clear register (i.e set lifcr of M\_DMA2 to M\_DMA\_STREAM3\_IT\_MASK) when dmay\_streamx is M\_DMA2\_STREAM3.

##### 11.6.1.7.15 daulibstm32f4xxdma-DmaDeInit-LLR-015

Requirement ID: H398-LLD-GWY-FNC-5059

The function shall clear the interrupt pending bits for DMA2 Stream4 by setting the CFEIF4, CDMEIF4, CTEIF4, CHTIF4 and CTCIF4 bits of DMA2 high interrupt flag clear register (i.e set hifcr of M\_DMA2 to M\_DMA\_STREAM4\_IT\_MASK) when dmay\_streamx is M\_DMA2\_STREAM4.

##### 11.6.1.7.16 daulibstm32f4xxdma-DmaDeInit-LLR-016

Requirement ID: H398-LLD-GWY-FNC-5060

The function shall clear the interrupt pending bits for DMA2 Stream5 by setting the CFEIF5, CDMEIF5, CTEIF5, CHTIF5 and CTCIF5 bits of DMA2 high interrupt flag clear register (i.e set hifcr of M\_DMA2 to M\_DMA\_STREAM5\_IT\_MASK) when dmay\_streamx is M\_DMA2\_STREAM5.

##### 11.6.1.7.17 daulibstm32f4xxdma-DmaDeInit-LLR-017

Requirement ID: H398-LLD-GWY-FNC-5061

The function shall clear the interrupt pending bits for DMA2 Stream6 by setting the CFEIF6, CDMEIF6, CTEIF6, CHTIF6 and CTCIF6 bits of DMA2 high interrupt flag clear register (i.e set hifcr of M\_DMA2 to M\_DMA\_STREAM6\_IT\_MASK) when dmay\_streamx is M\_DMA2\_STREAM6.

##### 11.6.1.7.18 daulibstm32f4xxdma-DmaDeInit-LLR-018

Requirement ID: H398-LLD-GWY-FNC-5062

The function shall do the following when dmay\_streamx is other than (M\_DMA1\_STREAM0, M\_DMA1\_STREAM1, M\_DMA1\_STREAM2, M\_DMA1\_STREAM3, M\_DMA1\_STREAM4, M\_DMA1\_STREAM5, M\_DMA1\_STREAM6, M\_DMA1\_STREAM7, M\_DMA2\_STREAM0, M\_DMA2\_STREAM1, M\_DMA2\_STREAM2, M\_DMA2\_STREAM3, M\_DMA2\_STREAM4, M\_DMA2\_STREAM5 and M\_DMA2\_STREAM6) :

a. Clear the interrupt pending bits for DMA2 Stream7 by setting the CFEIF7, CDMEIF7, CTEIF7, CHTIF7 and CTCIF7 bits of DMA2 high interrupt flag clear register (i.e set hifcr of M\_DMA2 to M\_DMA\_STREAM7\_IT\_MASK) when dmay\_streamx is M\_DMA2\_STREAM7.

b. Otherwise do nothing.

### 11.6.2 DmaInit

Low Level Design Details about CSU DmaInit will follow in the sub sections.

#### 11.6.2.1 Brief Description

The function DmaInit initializes the DMAy Streamx according to the specified parameters in the dma\_init\_struct structure.

#### 11.6.2.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 11.6.2.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 11.6.2.4 Parameter list (Input/Output)

Inputs: T\_DMA\_STREAM\* dmay\_streamx - Where y can be 1 or 2 to select the DMA and x can be 0 to 7 to select the DMA STREAM.

T\_DMA\_INIT\* dma\_init\_struct - Pointer to a T\_DMA\_INIT structure that contains the configuration information for the specified DMA STREAM.

Outputs: T\_DMA\_STREAM\* dmay\_streamx - Where y can be 1 or 2 to select the DMA and x can be 0 to 7 to select the DMA STREAM.

#### 11.6.2.5 Return Value

None

#### 11.6.2.6 Other CSUs called by this CSU

None

#### 11.6.2.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to DmaInit.

##### 11.6.2.7.1 daulibstm32f4xxdma-DmaInit-LLR-001

Requirement ID: H398-LLD-GWY-FNC-5071

The function shall clear the CHSEL, MBURST, PBURST, PL, MSIZE, PSIZE, MINC, PINC, CIRC and DIR bits of DMA stream x configuration register (cr of dmay\_streamx) and configure the same with dma\_channel, dma\_dir, dma\_peripheral\_inc, dma\_memory\_inc, dma\_peripheral\_datasize, dma\_memory\_datasize, dma\_mode, dma\_priority, dma\_memory\_burst and dma\_peripheral\_burst of dma\_init\_struct.

##### 11.6.2.7.2 daulibstm32f4xxdma-DmaInit-LLR-002

Requirement ID: H398-LLD-GWY-FNC-5072

The function shall clear the DMDIS and FTH bits of DMA stream x FIFO control register (fcr of dmay\_streamx) and configure the same with dma\_fifo\_mode and dma\_fifo\_threshold of dma\_init\_struct.

##### 11.6.2.7.3 daulibstm32f4xxdma-DmaInit-LLR-003

Requirement ID: H398-LLD-GWY-FNC-5073

The function shall configure DMA stream x number of data register (ndtr of dmay\_streamx), DMA stream x peripheral address register (par of dmay\_streamx) and DMA stream x memory 0 address register (m0ar of dmay\_streamx) with dma\_buffersize, dma\_peripheral\_baseaddr and dma\_memory0\_baseaddr of dma\_init\_struct respectively.

### 11.6.3 DmaStructInit

Low Level Design Details about CSU DmaStructInit will follow in the sub sections.

#### 11.6.3.1 Brief Description

The function DmaStructInit fills each dma\_init\_struct member with its default value.

#### 11.6.3.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 11.6.3.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 11.6.3.4 Parameter list (Input/Output)

Inputs: None

Outputs: T\_DMA\_INIT\* dma\_init\_struct - pointer to a T\_DMA\_INIT structure which will be initialized.

#### 11.6.3.5 Return Value

None

#### 11.6.3.6 Other CSUs called by this CSU

None

#### 11.6.3.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to DmaStructInit.

##### 11.6.3.7.1 daulibstm32f4xxdma-DmaStructInit-LLR-001

Requirement ID: H398-LLD-GWY-FNC-5082

The function shall initialize the DMA init structure (dma\_init\_struct) parameters values as below:

a) dma\_channel with M\_ZERO

b) dma\_peripheral\_baseaddr with M\_ZERO

c) dma\_memory0\_baseaddr with M\_ZERO

d) dma\_dir with M\_DMA\_DIR\_PERIPHERAL\_TO\_MEMORY

e) dma\_buffersize with M\_ZERO

f) dma\_peripheral\_inc with M\_DMA\_PERIPHERALINC\_DISABLE

g) dma\_memory\_inc with M\_DMA\_MEMORYINC\_DISABLE

h) dma\_peripheral\_datasize with M\_DMA\_PERIPHERAL\_DATASIZE\_BYTE

i) dma\_memory\_datasize with M\_DMA\_MEMORY\_DATA\_SIZE\_BYTE

j) dma\_mode with M\_DMA\_MODE\_NORMAL

k) dma\_priority with M\_DMA\_PRIORITY\_LOW

l) dma\_fifo\_mode with M\_DMA\_FIFOMODE\_DISABLE

m) dma\_fifo\_threshold with M\_DMA\_FIFO\_THR\_1QUARTERFULL

n) dma\_memory\_burst with M\_DMA\_MEMORYBURST\_SINGLE

o) dma\_peripheral\_burst with M\_DMA\_PERIPHERALBURST\_SINGLE

### 11.6.4 DmaCmd

Low Level Design Details about CSU DmaCmd will follow in the sub sections.

#### 11.6.4.1 Brief Description

The function DmaCmd enables or disables the specified DMAy Streamx.

#### 11.6.4.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 11.6.4.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 11.6.4.4 Parameter list (Input/Output)

Inputs: T\_DMA\_STREAM\* dmay\_streamx - Where y can be 1 or 2 to select the DMAand x can be 0 to 7 to select the DMA STREAM.

T\_FUNCTIONAL\_STATE new\_state - New state of the DMAy Streamx.

Outputs: T\_DMA\_STREAM\* dmay\_streamx - Where y can be 1 or 2 to select the DMAand x can be 0 to 7 to select the DMA STREAM.

#### 11.6.4.5 Return Value

None

#### 11.6.4.6 Other CSUs called by this CSU

None

#### 11.6.4.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to DmaCmd

##### 11.6.4.7.1 daulibstm32f4xxdma-DmaCmd-LLR-001

Requirement ID: H398-LLD-GWY-FNC-5091

The function shall set the selected DMAy Streamx by setting EN bit in DMA stream x configuration register (cr) (i.e Set cr of dmay\_streamx to cr of dmay\_streamx Bitwise OR with M\_DMA\_SXCR\_EN)if new\_state is ENABLE.

##### 11.6.4.7.2 daulibstm32f4xxdma-DmaCmd-LLR-002

Requirement ID: H398-LLD-GWY-FNC-5092

The function shall reset the selected DMAy Streamx by resetting the EN bit in DMA stream x configuration register (cr) (i.e Set cr of dmay\_streamx to cr of dmay\_streamx Bitwise AND with Negation of M\_DMA\_SXCR\_EN)if new\_state is DISABLE.

### 11.6.5 DmaClearItPendingBit

Low Level Design Details about CSU DmaClearItPendingBit will follow in the sub sections.

#### 11.6.5.1 Brief Description

The function DmaClearItPendingBit clears the DMAy Streamx's interrupt pending bits.

#### 11.6.5.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 11.6.5.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 11.6.5.4 Parameter list (Input/Output)

Inputs: T\_DMA\_STREAM \* dmay\_streamx - Where y can be 1 or 2 to select the DMA and x can be 0 to 7 to select the DMA STREAM.

T\_UINT32 dma\_interrupt - Specifies the DMA interrupt pending bit to clear.

Outputs: None.

#### 11.6.5.5 Return Value

None

#### 11.6.5.6 Other CSUs called by this CSU

None

#### 11.6.5.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to DmaClearItPendingBit

##### 11.6.5.7.1 daulibstm32f4xxdma-DmaClearItPendingBit-LLR-001

Requirement ID: H398-LLD-GWY-FNC-5101

The function shall set the DMAy as M\_DMA1 when dmay\_streamx is less than M\_DMA2\_STREAM0.

##### 11.6.5.7.2 daulibstm32f4xxdma-DmaClearItPendingBit-LLR-002

Requirement ID: H398-LLD-GWY-FNC-5102

The function shall set the DMAy as M\_DMA2 when dmay\_streamx is greater than or equal to M\_DMA2\_STREAM0.

##### 11.6.5.7.3 daulibstm32f4xxdma-DmaClearItPendingBit-LLR-003

Requirement ID: H398-LLD-GWY-FNC-5103

The function shall set the hifcr of DMAy register to (dma\_interrupt Bitwise AND with M\_RESERVED\_MASK) when (dma\_interrupt Bitwise AND with M\_HIGH\_ISR\_MASK) is equal to SET.

##### 11.6.5.7.4 daulibstm32f4xxdma-DmaClearItPendingBit-LLR-004

Requirement ID: H398-LLD-GWY-FNC-5104

The function shall set the lifcr of DMAy register to (dma\_interrupt Bitwise AND with M\_RESERVED\_MASK) when (dma\_interrupt Bitwise AND with M\_HIGH\_ISR\_MASK) is equal to RESET.

### 11.6.6 DmaItConfig

Low Level Design Details about CSU DmaItConfig will follow in the sub sections.

#### 11.6.6.1 Brief Description

The function DmaItConfig enables or disables the specified DMAy Streamx interrupts.

#### 11.6.6.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 11.6.6.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 11.6.6.4 Parameter list (Input/Output)

Inputs: T\_DMA\_STREAM\* dmay\_streamx - DMA Stream, where y can be 1 or 2 to select the DMA and x can be 0 to 7 to select the DMA Stream

T\_UINT32 dma\_interrupt - Specifies the DMA interrupt sources to be enabled or disabled.

T\_FUNCTIONAL\_STATE new\_state - New state of the DMAy Streamx.

Outputs: T\_DMA\_STREAM\* dmay\_streamx - DMA Stream, where y can be 1 or 2 to select the DMA and x can be 0 to 7 to select the DMA Stream

#### 11.6.6.5 Return Value

None

#### 11.6.6.6 Other CSUs called by this CSU

None

#### 11.6.6.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to DmaItConfig

##### 11.6.6.7.1 daulibstm32f4xxdma-DmaItConfig-LLR-001

Requirement ID: H398-LLD-GWY-FNC-5113

The function shall enable the selected DMA FIFO interrupts by setting the FEIE bit in DMA stream x FIFO control register (i.e set fcr of dmay\_streamx to fcr of dmay\_streamx Bitwise OR with M\_DMA\_IT\_FE) when dma\_interrupt contains FIFO interrupt(bit 7 is set) (i.e (dma\_interrupt Bitwise AND with M\_DMA\_IT\_FE) is not equal to M\_ZERO) and new\_state is ENABLE.

##### 11.6.6.7.2 daulibstm32f4xxdma-DmaItConfig-LLR-002

Requirement ID: H398-LLD-GWY-FNC-5114

The function shall

a. Enable the selected DMA FIFO interrupts by setting the FEIE bit in DMA stream x FIFO control register (i.e set fcr of dmay\_streamx to fcr of dmay\_streamx Bitwise AND with Negation of M\_DMA\_IT\_FE) when dma\_interrupt contains FIFO interrupt(bit 7 is set) (i.e (dma\_interrupt Bitwise AND with M\_DMA\_IT\_FE) is not equal to M\_ZERO) and new\_state is DISABLE.

b. Do nothing when (dma\_interrupt Bitwise AND with M\_DMA\_IT\_FE) is equal to M\_ZERO.

##### 11.6.6.7.3 daulibstm32f4xxdma-DmaItConfig-LLR-003

Requirement ID: H398-LLD-GWY-FNC-5115

The function shall enable the selected DMA transfer interrupts by setting the TCIE, HTIE,TEIE and DMEIE bits of DMA stream x configuration register for the received dma\_interrupt(i.e set cr of dmay\_streamx to cr of dmay\_streamx Bitwise OR with (dma\_interrupt Bitwise AND with M\_TRANSFER\_IT\_ENABLE\_MASK)) when dma\_interrupt is not M\_DMA\_IT\_FE and new\_state is ENABLE.

##### 11.6.6.7.4 daulibstm32f4xxdma-DmaItConfig-LLR-004

Requirement ID: H398-LLD-GWY-FNC-5116

The function shall

a. Enable the selected DMA transfer interrupts by setting the TCIE, HTIE,TEIE and DMEIE bits of DMA stream x configuration register for the received dma\_interrupt(i.e set cr of dmay\_streamx to cr of dmay\_streamx Bitwise AND with Negation of (dma\_interrupt Bitwise AND with M\_TRANSFER\_IT\_ENABLE\_MASK)) when dma\_interrupt is not M\_DMA\_IT\_FE and new\_state is DISABLE.

b. Do nothing when dma\_interrupt is M\_DMA\_IT\_FE.

## 11.7 daulibstm32f4xxflash

This module provides firmware functions to manage the following functionalities of the FLASH peripheral:

* FLASH Interface configuration

a) Set the latency

b) Enable/Disable the prefetch buffer

c) Enable/Disable the Instruction cache and the Data cache

* Interrupts and flags management

### 11.7.1 FlashSetLatency

Low Level Design Details about CSU FlashSetLatency will follow in the sub sections.

#### 11.7.1.1 Brief Description

The function FlashSetLatency sets the code latency value.

#### 11.7.1.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 11.7.1.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 11.7.1.4 Parameter list (Input/Output)

Inputs: T\_UINT32 flash\_latency - specifies the FLASH Latency value.

Outputs: None

#### 11.7.1.5 Return Value

None

#### 11.7.1.6 Other CSUs called by this CSU

None

#### 11.7.1.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to FlashSetLatency.

##### 11.7.1.7.1 daulibstm32f4xxflash-FlashSetLatency-LLR-001

Requirement ID: H398-LLD-GWY-FNC-5126

The function shall set the code latency value in acr (Flash access control register ([8:0])) i.e,

Set content of the address M\_ACR\_BYTE0\_ADDRESS to flash\_latency.

### 11.7.2 FlashPrefetchBufferCmd

Low Level Design Details about CSU FlashPrefetchBufferCmd will follow in the sub sections.

#### 11.7.2.1 Brief Description

The function FlashPrefetchBufferCmd enables or disables the Prefetch Buffer.

#### 11.7.2.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 11.7.2.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 11.7.2.4 Parameter list (Input/Output)

Inputs: T\_FUNCTIONAL\_STATE new\_state - New state of the Prefetch Buffer.

Outputs: None

#### 11.7.2.5 Return Value

None

#### 11.7.2.6 Other CSUs called by this CSU

None

#### 11.7.2.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to FlashPrefetchBufferCmd

##### 11.7.2.7.1 daulibstm32f4xxflash-FlashPrefetchBufferCmd-LLR-001

Requirement ID: H398-LLD-GWY-FNC-5135

The function shall enable the prefetch buffer (set PRFTEN – bit 8) in acr (Flash access control register) when new\_state is ENABLE i.e, set acr of M\_FLASH with (acr of M\_FLASH bitwise OR with M\_FLASH\_ACR\_PRFTEN).

##### 11.7.2.7.2 daulibstm32f4xxflash-FlashPrefetchBufferCmd-LLR-002

Requirement ID: H398-LLD-GWY-FNC-5136

The function shall disable the prefetch buffer (reset PRFTEN – bit 8) in acr (Flash access control register) when new\_state is DISABLE i.e, set acr of M\_FLASH with (acr of M\_FLASH bitwise AND with negated value of M\_FLASH\_ACR\_PRFTEN).

### 11.7.3 FlashInstructionCacheCmd

Low Level Design Details about CSU FlashInstructionCacheCmd will follow in the sub sections.

#### 11.7.3.1 Brief Description

The function FlashInstructionCacheCmd enables or disables the Instruction Cache feature.

#### 11.7.3.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 11.7.3.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 11.7.3.4 Parameter list (Input/Output)

Inputs: T\_FUNCTIONAL\_STATE new\_state - New state of the Instruction Cache.

Outputs: None

#### 11.7.3.5 Return Value

None

#### 11.7.3.6 Other CSUs called by this CSU

None

#### 11.7.3.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to FlashInstructionCacheCmd

##### 11.7.3.7.1 daulibstm32f4xxflash-FlashInstructionCacheCmd-LLR-001

Requirement ID: H398-LLD-GWY-FNC-5145

The function shall enable the Instruction Cache (set ICEN – bit 9) in acr (Flash access control register) when new\_state is ENABLE i.e, set acr of M\_FLASH with (acr of M\_FLASH bitwise OR with M\_FLASH\_ACR\_ICEN).

##### 11.7.3.7.2 daulibstm32f4xxflash-FlashInstructionCacheCmd-LLR-002

Requirement ID: H398-LLD-GWY-FNC-5146

The function shall disable the Instruction Cache (reset ICEN – bit 9) in acr (Flash access control register) when new\_state is DISABLE i.e, set acr of M\_FLASH with (acr of M\_FLASH bitwise AND with negated value of M\_FLASH\_ACR\_ICEN).

### 11.7.4 FlashDataCacheCmd

Low Level Design Details about CSU FlashDataCacheCmd will follow in the sub sections.

#### 11.7.4.1 Brief Description

The function FlashDataCacheCmd enables or disables the Data Cache feature.

#### 11.7.4.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 11.7.4.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 11.7.4.4 Parameter list (Input/Output)

Inputs: T\_FUNCTIONAL\_STATE new\_state - New state of the data Cache.

Outputs: None

#### 11.7.4.5 Return Value

None

#### 11.7.4.6 Other CSUs called by this CSU

None

#### 11.7.4.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to FlashDataCacheCmd

##### 11.7.4.7.1 daulibstm32f4xxflash-FlashDataCacheCmd-LLR-001

Requirement ID: H398-LLD-GWY-FNC-5155

The function shall enable the Data Cache (set DCEN - bit 10) in acr (Flash access control register) when new\_state is ENABLE i.e, set acr of M\_FLASH with (acr of M\_FLASH bitwise OR with M\_FLASH\_ACR\_DCEN).

##### 11.7.4.7.2 daulibstm32f4xxflash-FlashDataCacheCmd-LLR-002

Requirement ID: H398-LLD-GWY-FNC-5156

The function shall disable the Data Cache (reset DCEN - bit 10) of acr (Flash access control register) when new\_state is DISABLE i.e, set acr of M\_FLASH with (acr of M\_FLASH bitwise AND with negated value of M\_FLASH\_ACR\_DCEN).

## 11.8 daulibstm32f4xxfsmc

This module provides firmware functions to manage the following functionality of the FSMC peripheral:

* Interface with SRAM and NOR memories.

### 11.8.1 FsmcNorSramInit

Low Level Design Details about CSU FsmcNorSramInit will follow in the sub sections.

#### 11.8.1.1 Brief Description

The function FsmcNorSramInit initializes the FSMC NOR/SRAM Banks according to the specified parameters in the FSMC NOR/SRAM initialization structure.

#### 11.8.1.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 11.8.1.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 11.8.1.4 Parameter list (Input/Output)

Inputs: T\_FSMC\_NORSRAM\_INIT\* fsmc\_norsraminit\_struct - Pointer to a T\_FSMC\_NORSRAM\_INIT structure that contains the configuration information for the FSMC NOR/SRAM specified Banks.

Outputs: None

#### 11.8.1.5 Return Value

None

#### 11.8.1.6 Other CSUs called by this CSU

None

#### 11.8.1.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to FsmcNorSramInit

##### 11.8.1.7.1 daulibstm32f4xxfsmc-FsmcNorSramInit-LLR-001

Requirement ID: H398-LLD-GWY-FNC-5166

The function shall configure the FSMC BANK1 NOR/SRAM FSMC Bank control register with the received configuration information (Address/data multiplexing enable bit, Memory Type, Memory databus width, Burst enable bit, Wait signal during asynchronous transfers, Wait signal polarity bit, Wrapped burst mode, Wait timing configuration, Write enable bit, Wait enable bit, Extended mode enable, Write burst enable) i.e,

Set btcr of M\_FSMC\_BANK1 with index (fsmc\_bank of fsmc\_norsraminit\_struct) to (fsmc\_data\_address\_mux of fsmc\_norsraminit\_struct bitwise OR with

fsmc\_memory\_type of fsmc\_norsraminit\_struct bitwise OR with

fsmc\_memory\_datawidth of fsmc\_norsraminit\_struct bitwise OR with

fsmc\_burst\_accessmode of fsmc\_norsraminit\_struct bitwise OR with

fsmc\_asynchronous\_wait of fsmc\_norsraminit\_struct bitwise OR with

fsmc\_waitsignal\_polarity of fsmc\_norsraminit\_struct bitwise OR with

fsmc\_wrap\_mode of fsmc\_norsraminit\_struct bitwise OR with

fsmc\_waitsignal\_active of fsmc\_norsraminit\_struct bitwise OR with

fsmc\_write\_operation of fsmc\_norsraminit\_struct bitwise OR with

fsmc\_waitsignal of fsmc\_norsraminit\_struct bitwise OR with

fsmc\_extended\_mode of fsmc\_norsraminit\_struct bitwise OR with

fsmc\_write\_burst of fsmc\_norsraminit\_struct).

##### 11.8.1.7.2 daulibstm32f4xxfsmc-FsmcNorSramInit-LLR-002

Requirement ID: H398-LLD-GWY-FNC-5167

The function shall enable NOR Flash memory access operations in BCR1 register when FACCEN - bit 6 in BCR1 is set i.e,

Set btcr of M\_FSMC\_BANK1 with index (fsmc\_bank of fsmc\_norsraminit\_struct) to (btcr of M\_FSMC\_BANK1 with index (fsmc\_bank of fsmc\_norsraminit\_struct) bitwise OR with M\_BCR\_FACCEN\_SET) when fsmc\_memory\_type of fsmc\_norsraminit\_struct is equal to M\_FSMC\_MEMORYTYPE\_NOR otherwise do nothing.

##### 11.8.1.7.3 daulibstm32f4xxfsmc-FsmcNorSramInit-LLR-003

Requirement ID: H398-LLD-GWY-FNC-5168

The function shall configure the FSMC BANK1 NOR/SRAM Bank Timing register with the received configuration information (Address setup phase duration, Address-hold phase duration, Memory databus width, Data-phase duration, Bus turnaround phase duration, Clock divide ratio, Data latency, Access mode) i.e,

Set btcr of M\_FSMC\_BANK1 with index (fsmc\_bank of fsmc\_norsraminit\_struct added with M\_ONE) to (fsmc\_address\_setuptime of fsmc\_readwrite\_timing\_struct of fsmc\_norsraminit\_struct bitwise OR with (fsmc\_address\_holdtime of fsmc\_readwrite\_timing\_struct of fsmc\_norsraminit\_struct left shift by M\_SHIFT\_4) bitwise OR with

(fsmc\_data\_setuptime of fsmc\_readwrite\_timing\_struct of fsmc\_norsraminit\_struct left shift by M\_SHIFT\_8) bitwise OR with

(fsmc\_bus\_turnaround\_duration of fsmc\_readwrite\_timing\_struct of fsmc\_norsraminit\_struct left shift by M\_SHIFT\_16) bitwise OR with

(fsmc\_clk\_division of fsmc\_readwrite\_timing\_struct of fsmc\_norsraminit\_struct left shift by M\_SHIFT\_20) bitwise OR with

(fsmc\_data\_latency of fsmc\_readwrite\_timing\_struct of fsmc\_norsraminit\_struct left shift by M\_SHIFT\_24) bitwise OR with

(fsmc\_access\_mode of fsmc\_readwrite\_timing\_struct of fsmc\_norsraminit\_struct)).

##### 11.8.1.7.4 daulibstm32f4xxfsmc-FsmcNorSramInit-LLR-004

Requirement ID: H398-LLD-GWY-FNC-5169

The function shall configure Bank1 NOR/SRAM Flash write timing register for write configuration with the received Timing parameters (fsmc address setuptime, fsmc holdtime, fsmc data setuptime, fsmc clock division, fsmc data latency and fsmc access mode) when extended mode is enabled i.e,

Set bwtr of M\_FSMC\_BANK1E with index (fsmc\_bank of fsmc\_norsraminit\_struct) to

((fsmc\_address\_setuptime of fsmc\_write\_timing\_struct of fsmc\_norsraminit\_struct) bitwise OR with (fsmc\_address\_holdtime of fsmc\_write\_timing\_struct of fsmc\_norsraminit\_struct left shift by M\_SHIFT\_4) bitwise OR with

(fsmc\_data\_setuptime of fsmc\_write\_timing\_struct of fsmc\_norsraminit\_struct left shift by M\_SHIFT\_8) bitwise OR with

(fsmc\_clk\_division of fsmc\_write\_timing\_struct of fsmc\_norsraminit\_struct left shift by M\_SHIFT\_20) bitwise OR with

(fsmc\_data\_latency of fsmc\_write\_timing\_struct of fsmc\_norsraminit\_struct left shift by M\_SHIFT\_24) bitwise OR with

(fsmc\_access\_mode of fsmc\_write\_timing\_struct of fsmc\_norsraminit\_struct)) when fsmc\_extended\_mode of fsmc\_norsraminit\_struct is equal to M\_FSMC\_EXTENDEDMODE\_ENABLE.

##### 11.8.1.7.5 daulibstm32f4xxfsmc-FsmcNorSramInit-LLR-005

Requirement ID: H398-LLD-GWY-FNC-5170

The function shall reset Bank1 NOR/SRAM Flash write timing register when extended mode is not used i.e,

Set bwtr of M\_FSMC\_BANK1E with index (fsmc\_bank of fsmc\_norsraminit\_struct) to M\_BWTR\_RESET\_VALUE when fsmc\_extended\_mode of fsmc\_norsraminit\_struct is not equal to M\_FSMC\_EXTENDEDMODE\_ENABLE.

### 11.8.2 FsmcNorSramCmd

Low Level Design Details about CSU FsmcNorSramCmd will follow in the sub sections.

#### 11.8.2.1 Brief Description

The function FsmcNorSramCmd enables or disables the specified NOR/SRAM Memory Bank.

#### 11.8.2.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 11.8.2.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 11.8.2.4 Parameter list (Input/Output)

Inputs: T\_UINT32 fsmc\_bank - Specifies the FSMC Bank to be used

T\_FUNCTIONAL\_STATE new\_state - New state of the fsmc\_bank.

Outputs: None

#### 11.8.2.5 Return Value

None

#### 11.8.2.6 Other CSUs called by this CSU

None

#### 11.8.2.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to FsmcNorSramCmd

##### 11.8.2.7.1 daulibstm32f4xxfsmc-FsmcNorSramCmd-LLR-001

Requirement ID: H398-LLD-GWY-FNC-5179

The function shall enable selected NOR/SRAM Memory Bank (MBKEN - bit 0) in the Bank Control Register when new state of FSMC bank is ENABLE i.e,

Set btcr of M\_FSMC\_BANK1 with index fsmc\_bank to Enable (btcr of M\_FSMC\_BANK1 with index fsmc\_bank bitwise OR with M\_BCR\_MBKEN\_SET).

##### 11.8.2.7.2 daulibstm32f4xxfsmc-FsmcNorSramCmd-LLR-002

Requirement ID: H398-LLD-GWY-FNC-5180

The function shall disable selected NOR/SRAM Memory Bank (MBKEN - bit 0) in the Bank Control Register when new state of FSMC bank is DISABLE i.e,

Set btcr of M\_FSMC\_BANK1 with index fsmc\_bank to Disable(btcr of M\_FSMC\_BANK1 with index fsmc\_bank bitwise AND with M\_BCR\_MBKEN\_RESET).

## 11.9 daulibstm32f4xxgpio

This module provides the implementation of firmware functions to manage the following functionalities of the GPIO peripheral:

* Initialization and Configuration
* GPIO Read and Write
* GPIO Alternate functions configuration

### 11.9.1 GpioInit

Low Level Design Details about CSU GpioInit will follow in the sub sections.

#### 11.9.1.1 Brief Description

The function GpioInit initializes the gpio\_x peripheral according to the specified parameters in the GPIO initialization structure.

#### 11.9.1.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 11.9.1.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 11.9.1.4 Parameter list (Input/Output)

Inputs: T\_GPIO\* gpio\_x - GPIO peripheral, where x can be (A...I) to select the GPIO peripheral.

T\_GPIO\_INIT\* gpio\_init\_struct - Pointer to a T\_GPIO\_INIT structure that contains the configuration information for the specified GPIO peripheral.

Outputs: T\_GPIO\* gpio\_x - GPIO peripheral, where x can be (A...I) to select the GPIO peripheral.

#### 11.9.1.5 Return Value

None

#### 11.9.1.6 Other CSUs called by this CSU

None

#### 11.9.1.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to GpioInit

##### 11.9.1.7.1 daulibstm32f4xxgpio-GpioInit-LLR-001

Requirement ID: H398-LLD-GWY-FNC-5190

The function shall loop for all gpio pin position (M\_ZERO to M\_GPIO\_PERIPHERALS) and does the following:

1. Set position variable to M\_ONE left shift by gpio pin position
2. Calculate the current pin position i.e, set current pin position to (gpio\_pin of gpio\_init\_struct bitwise AND with position variable).
3. when current pin position is equal to position variable, perform as follows otherwise do nothing.

* Clear the GPIO\_MODER register for the current pin position i.e, set moder of gpio\_x to (moder of gpio\_x bitwise AND with negated value of (M\_GPIO\_MODER\_MODER0 left shift by (gpio pin position multiplied with M\_TWO))).
* Set the GPIO\_MODER register with the received gpio mode i.e, set moder of gpio\_x to (moder of gpio\_x bitwise OR with (gpio\_mode of gpio\_init\_struct left shift by (gpio pin position multiplied by M\_TWO))).
* When the received gpio mode is output or alternate function(GPIO\_MODE\_OUT is equal to gpio\_mode of gpio\_init\_struct OR GPIO\_MODE\_AF is equal to gpio\_mode of gpio\_init\_struct), perform as follows otherwise do nothing.
* Clear the GPIO\_OSPEEDR register for the current pin position i.e, set ospeedr of gpio\_x to (ospeedr of gpio\_x bitwise AND with negated value of (M\_GPIO\_OSPEEDER\_OSPEEDR0 left shift by (gpio pin position multiplied by M\_TWO))).
* Set the GPIO\_OSPEEDR register with the received gpio speed i.e, set ospeedr of gpio\_x to (ospeedr of gpio\_x bitwise OR with (gpio\_speed of gpio\_init\_struct left shift by (gpio pin position multiplied by M\_TWO))).
* Clear the GPIO\_OTYPER register for the current pin position i.e, set otyper of gpio\_x to (otyper of gpio\_x bitwise AND with negated value of (M\_GPIO\_OTYPER\_OT\_0 left shift by gpio pin position)).
* Set the GPIO\_OTYPER register with the received gpio output type i.e, set otyper of gpio\_x to (otyper of gpio\_x bitwise OR with (gpio\_otype of gpio\_init\_struct left shift by gpio pin position)).
* Clear the GPIO\_PUPDR register for the current pin position i.e, set pupdr of gpio\_x to (pupdr of gpio\_x bitwise AND with negated value of (M\_GPIO\_PUPDR\_PUPDR0 left shift by (gpio pin position multiplied by M\_TWO))).
* Set the GPIOx\_PUPDR register with Pull-up Pull down resistor configuration, i.e, set pupdr of gpio\_x to (pupdr of gpio\_x bitwise OR with gpio\_pupd of gpio\_init\_struct left shift by (gpio pin position multiplied by M\_TWO)).

### 11.9.2 GpioSetBits

Low Level Design Details about CSU GpioSetBits will follow in the sub sections.

#### 11.9.2.1 Brief Description

The function GpioSetBits sets the selected data port bits.

#### 11.9.2.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 11.9.2.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 11.9.2.4 Parameter list (Input/Output)

Inputs: T\_UINT16 gpio\_pin - Specifies the port bits to be set.

Outputs: T\_GPIO\* gpio\_x - GPIO peripheral, where x can be (A...I) to select the GPIO peripheral.

#### 11.9.2.5 Return Value

None

#### 11.9.2.6 Other CSUs called by this CSU

None

#### 11.9.2.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to GpioSetBits.

##### 11.9.2.7.1 daulibstm32f4xxgpio-GpioSetBits-LLR-001

Requirement ID: H398-LLD-GWY-FNC-5199

The function shall set the GPIO port bit set/reset low register i.e, set bsrr1 of gpio\_x to gpio\_pin.

### 11.9.3 GpioResetBits

Low Level Design Details about CSU GpioResetBits will follow in the sub sections.

#### 11.9.3.1 Brief Description

The function GpioResetBits clears the selected data port bits.

#### 11.9.3.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 11.9.3.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 11.9.3.4 Parameter list (Input/Output)

Inputs: T\_UINT16 gpio\_pin - specifies the port bits to be reset.

Outputs: T\_GPIO\* gpio\_x - GPIO peripheral, where x can be (A...I) to select the GPIO peripheral.

#### 11.9.3.5 Return Value

None

#### 11.9.3.6 Other CSUs called by this CSU

None

#### 11.9.3.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to GpioResetBits.

##### 11.9.3.7.1 daulibstm32f4xxgpio-GpioResetBits-LLR-001

Requirement ID: H398-LLD-GWY-FNC-5208

The function shall set the GPIO port bit set/reset high register i.e, bsrrh of gpio\_x to gpio\_pin.

### 11.9.4 GpioReadInputDataBit

Low Level Design Details about CSU GpioReadInputDataBit will follow in the sub sections.

#### 11.9.4.1 Brief Description

The function GpioReadInputDataBit reads the specified input port pin.

#### 11.9.4.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 11.9.4.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 11.9.4.4 Parameter list (Input/Output)

Inputs: T\_UINT16 gpio\_pin - specifies the port bits to be read.

T\_GPIO\* gpio\_x - GPIO peripheral, where x can be (A...I) to select the GPIO peripheral

Outputs: None

#### 11.9.4.5 Return Value

T\_UINT8 – Returns the input port pin value

#### 11.9.4.6 Other CSUs called by this CSU

None

#### 11.9.4.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to GpioReadInputDataBit.

##### 11.9.4.7.1 daulibstm32f4xxgpio-GpioReadInputDataBit-LLR-001

Requirement ID: H398-LLD-GWY-FNC-5217

The function shall Return BIT\_SET when gpio\_pin in gpio\_x input data register(idr) is SET i.e idr of gpio\_x Bitwise AND with gpio\_pin is not BIT\_RESET.

##### 11.9.4.7.2 daulibstm32f4xxgpio-GpioReadInputDataBit-LLR-002

Requirement ID: H398-LLD-GWY-FNC-5218

The function shall Return BIT\_RESET when gpio\_pin in gpio\_x input data register(idr) is RESET i.e idr of gpio\_x Bitwise AND with gpio\_pin is BIT\_RESET.

### 11.9.5 GpioToggleBits

Low Level Design Details about CSU GpioToggleBits will follow in the sub sections.

#### 11.9.5.1 Brief Description

The function GpioToggleBits toggles the specified GPIO pins.

#### 11.9.5.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 11.9.5.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 11.9.5.4 Parameter list (Input/Output)

Inputs: T\_GPIO\* gpio\_x - GPIO peripheral, where x can be (A...I) to select the GPIO peripheral.

T\_UINT16 gpio\_pin - Specifies the pins to be toggled.

Outputs: T\_GPIO\* gpio\_x - GPIO peripheral, where x can be (A...I) to select the GPIO peripheral.

#### 11.9.5.5 Return Value

None

#### 11.9.5.6 Other CSUs called by this CSU

None

#### 11.9.5.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to GpioToggleBits.

##### 11.9.5.7.1 daulibstm32f4xxgpio-GpioToggleBits-LLR-001

Requirement ID: H398-LLD-GWY-FNC-5227

The function shall toggle the received gpio pin gpio\_pin in Gpio output data register i.e, set odr of gpio\_x to (odr of gpio\_x bitwise XOR with gpio\_pin).

### 11.9.6 GpioPinAFConfig

Low Level Design Details about CSU GpioPinAFConfig will follow in the sub sections.

#### 11.9.6.1 Brief Description

The function GpioPinAFConfig changes the mapping of the specified pin.

#### 11.9.6.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 11.9.6.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 11.9.6.4 Parameter list (Input/Output)

Inputs: T\_GPIO\* gpio\_x - GPIO peripheral, where x can be (A...I) to select the GPIO peripheral.

T\_UINT16 gpio\_pinsource - specifies the pin for the Alternate function.

T\_UINT8 gpio\_af - selects the pin to be used as Alternate function.

Outputs: T\_GPIO\* gpio\_x - GPIO peripheral, where x can be (A...I) to select the GPIO peripheral.

#### 11.9.6.5 Return Value

None

#### 11.9.6.6 Other CSUs called by this CSU

None

#### 11.9.6.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to GpioPinAFConfig

##### 11.9.6.7.1 daulibstm32f4xxgpio-GpioPinAFConfig-LLR-001

Requirement ID: H398-LLD-GWY-FNC-5236

The function shall change the mapping of specified pin gpio\_pinsource to the alternate function by configuring the afr (Alternate function register) i.e,

1. set Temp afr register to (gpio\_af left shift by (gpio\_pinsource bitwise AND with (M\_SEVEN\_HEX multiplied with M\_FOUR))).
2. Set afr of gpio\_x with index (gpio\_ pinsource right shift by M\_THREE) to (afr of gpio\_x with index (gpio\_ pinsource right shift by M\_THREE) bitwise AND with negated value of (M\_FIFTEEN left shift by (gpio\_ pinsource bitwise AND with (M\_SEVEN\_HEX multiplied with M\_FOUR)))).
3. set Temp afr register 2 to (afr of gpio\_x with index (gpio\_ pinsource right shift by M\_THREE) bitwise OR with Temp afr register.
4. Set afr of gpio\_x with index (gpio\_ pinsource right shift by M\_THREE) to Temp afr register 2.

## 11.10 daulibstm32f4xxiwdg

This module provides firmware functions to manage the following functionality of the Independent watchdog (IWDG) peripheral:

* Prescaler and Counter configuration
* IWDG activation

### 11.10.1 IwdgWriteAccessCmd

Low Level Design Details about CSU IwdgWriteAccessCmd will follow in the sub sections.

#### 11.10.1.1 Brief Description

The function IwdgWriteAccessCmd enables or disables write access to IWDG\_PR (Prescaler register) and IWDG\_RLR (Reload register).

#### 11.10.1.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 11.10.1.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 11.10.1.4 Parameter list (Input/Output)

Inputs: T\_UINT16 iwdg\_write\_access – New state of write access to IWDG\_PR and IWDG\_RLR registers.

Outputs: None

#### 11.10.1.5 Return Value

None

#### 11.10.1.6 Other CSUs called by this CSU

None

#### 11.10.1.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to IwdgWriteAccessCmd

##### 11.10.1.7.1 daulibstm32f4xxiwdg-IwdgWriteAccessCmd-LLR-001

Requirement ID: H398-LLD-GWY-FNC-5246

The function shall configure IWDG Key register with iwdg\_write\_access to enable or disable access to the IWDG\_PR and IWDG\_RLR registers i.e, set kr of M\_IWDG to iwdg\_write\_access.

Note:

1. Write access to the IWDG\_PR and IWDG\_RLR registers is protected. To modify these registers, 0x5555 (IWDG\_PR), 0xAAAA (IWDG\_RLR) has to be written in the IWDG\_KR register.
2. Writing the key value CCCCh starts the watchdog.

### 11.10.2 IwdgSetPrescaler

Low Level Design Details about CSU IwdgSetPrescaler will follow in the sub sections.

#### 11.10.2.1 Brief Description

The function IwdgSetPrescaler sets IWDG prescaler value in Prescaler register.

#### 11.10.2.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 11.10.2.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 11.10.2.4 Parameter list (Input/Output)

Inputs: T\_UINT8 iwdg\_prescaler - Specifies the IWDG Prescaler value.

Outputs: None

#### 11.10.2.5 Return Value

None

#### 11.10.2.6 Other CSUs called by this CSU

None

#### 11.10.2.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to IwdgSetPrescaler

##### 11.10.2.7.1 daulibstm32f4xxiwdg-IwdgSetPrescaler-LLR-001

Requirement ID: H398-LLD-GWY-FNC-5255

The function shall configure IWDG Prescaler register (pr) with prescaler value iwdg\_prescaler i.e, set pr of M\_IWDG to iwdg\_prescaler.

### 11.10.3 IwdgSetReload

Low Level Design Details about CSU IwdgSetReloadwill follow in the sub sections.

#### 11.10.3.1 Brief Description

The function IwdgSetReload sets IWDG reload value in Reload register.

#### 11.10.3.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 11.10.3.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 11.10.3.4 Parameter list (Input/Output)

Inputs: T\_UINT16 reload - specifies the IWDG Reload value.

Outputs: None

#### 11.10.3.5 Return Value

None

#### 11.10.3.6 Other CSUs called by this CSU

None

#### 11.10.3.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to IwdgSetReload.

##### 11.10.3.7.1 daulibstm32f4xxiwdg-IwdgSetReload-LLR-001

Requirement ID: H398-LLD-GWY-FNC-5264

The function shall configure IWDG Reload register (rlr) with reload value i.e, set rlr of M\_IWDG to reload.

### 11.10.4 IwdgReloadCounter

Low Level Design Details about CSU IwdgReloadCounter will follow in the sub sections.

#### 11.10.4.1 Brief Description

The function IwdgReloadCounter reloads IWDG counter with value defined in the reload register.

#### 11.10.4.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 11.10.4.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 11.10.4.4 Parameter list (Input/Output)

Inputs: None

Outputs: None

#### 11.10.4.5 Return Value

None

#### 11.10.4.6 Other CSUs called by this CSU

None

#### 11.10.4.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to IwdgReloadCounter.

##### 11.10.4.7.1 daulibstm32f4xxiwdg-IwdgReloadCounter-LLR-001

Requirement ID: H398-LLD-GWY-FNC-5273

The function shall reload the watchdog counter with the reload value in reload register (IWDG\_RLR) i.e, set kr of M\_IWDG to M\_KR\_KEY\_RELOAD.

### 11.10.5 IwdgEnable

Low Level Design Details about CSU IwdgEnable will follow in the sub sections.

#### 11.10.5.1 Brief Description

The function IwdgEnable enables IWDG (write access to IWDG\_PR and IWDG\_RLR registers disabled).

#### 11.10.5.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 11.10.5.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 11.10.5.4 Parameter list (Input/Output)

Inputs: None

Outputs: None

#### 11.10.5.5 Return Value

None

#### 11.10.5.6 Other CSUs called by this CSU

None

#### 11.10.5.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to IwdgEnable

##### 11.10.5.7.1 daulibstm32f4xxiwdg-IwdgEnable-LLR-001

Requirement ID: H398-LLD-GWY-FNC-5282

The function shall configure IWDG Key register with the value M\_KR\_KEY\_ENABLE to start the watchdog i.e, set kr of M\_IWDG to M\_KR\_KEY\_ENABLE.

## 11.11 daulibstm32f4xxpwr

This module provides firmware functions to manage the following functionality of the Power Controller (PWR) peripheral:

* Main Regulator configuration

### 11.11.1 PwrMainRegulatorModeConfig

Low Level Design Details about CSU PwrMainRegulatorModeConfig will follow in the sub sections.

#### 11.11.1.1 Brief Description

The function PwrMainRegulatorModeConfig configures the main internal regulator output voltage.

#### 11.11.1.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 11.11.1.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 11.11.1.4 Parameter list (Input/Output)

Inputs: T\_UINT32 pwr\_regulator\_voltage - Specifies the regulator output voltage to achieve a tradeoff between performance and power consumption when the device does not operate at the maximum frequency.

Outputs: None

#### 11.11.1.5 Return Value

None

#### 11.11.1.6 Other CSUs called by this CSU

None

#### 11.11.1.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to PwrMainRegulatorModeConfig.

##### 11.11.1.7.1 daulibstm32f4xxpwr-PwrMainRegulatorModeConfig-LLR-001

Requirement ID: H398-LLD-GWY-FNC-5292

The function shall configure power control register (PWR\_CR) to Scale 2 mode (bit 14) when scale 2 is seleted i.e, set cr of M\_PWR to (cr of M\_PWR bitwise AND with negated value of M\_PWR\_REGULATOR\_VOLTAGE\_SCALE1) when M\_PWR\_REGULATOR\_VOLTAGE\_SCALE2 is equal to pwr\_regulator\_voltage.

##### 11.11.1.7.2 daulibstm32f4xxpwr-PwrMainRegulatorModeConfig-LLR-002

Requirement ID: H398-LLD-GWY-FNC-5293

The function shall configure power control register (PWR\_CR) to Scale 1 mode (bit 14) when scale 1 is seleted i.e, set cr of M\_PWR to (cr of M\_PWR bitwise OR with M\_PWR\_REGULATOR\_VOLTAGE\_SCALE1) when M\_PWR\_REGULATOR\_VOLTAGE\_SCALE2 is not equal to pwr\_regulator\_voltage.

## 11.12 daulibstm32f4xxrcc

The daulibstm32f4xxrcc CSC provides firmware functions to manage the following functionalities of the Reset and clock control (RCC) peripheral:

* Internal/external clocks, PLL, CSS and MCO configuration
* System, AHB and APB busses clocks configuration
* Peripheral clocks configuration
* Interrupts and flags management

### 11.12.1 RccDeInit

Low Level Design Details about CSU RccDeInit will follow in the sub sections.

#### 11.12.1.1 Brief Description

The function RccDeInit Resets the RCC clock configuration to the default reset state.

#### 11.12.1.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 11.12.1.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 11.12.1.4 Parameter list (Input/Output)

Inputs: None

Outputs: None

#### 11.12.1.5 Return Value

None

#### 11.12.1.6 Other CSUs called by this CSU

None

#### 11.12.1.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to RccDeInit

##### 11.12.1.7.1 daulibstm32f4xxrcc-RccDeInit-LLR-001

Requirement ID: H398-LLD-GWY-FNC-5303

The function shall set the HSION bit in RCC clock control register (cr).(i.e., set cr of M\_RCC to(cr of M\_RCC bitwise OR with M\_HEX\_ONE)).

##### 11.12.1.7.2 daulibstm32f4xxrcc-RccDeInit-LLR-002

Requirement ID: H398-LLD-GWY-FNC-5304

The function shall reset the cfgr register of RCC (i.e., set cfgr of M\_RCC to M\_HEX\_ZERO)

##### 11.12.1.7.3 daulibstm32f4xxrcc-RccDeInit-LLR-003

Requirement ID: H398-LLD-GWY-FNC-5305

The function shall reset HSEON, CSSON and PLLON bits of RCC clock control register (cr). (i.e., set cr of M\_RCC to (cr of M\_RCC Bitwise AND with M\_RESET\_HSEON\_CSSON\_PLLON )).

##### 11.12.1.7.4 daulibstm32f4xxrcc-RccDeInit-LLR-004

Requirement ID: H398-LLD-GWY-FNC-5306

The function shall reset pllcfgr register of RCC.(i.e., pllcfgr of M\_RCC to M\_RESET\_PLLCFGR)

##### 11.12.1.7.5 daulibstm32f4xxrcc-RccDeInit-LLR-005

Requirement ID: H398-LLD-GWY-FNC-5307

The function shall reset HSEBYP bit of RCC clock control register (cr).(i.e., cr of M\_RCC Bitwise AND with M\_RESET\_HSEBYP)

##### 11.12.1.7.6 daulibstm32f4xxrcc-RccDeInit-LLR-006

Requirement ID: H398-LLD-GWY-FNC-5308

The function shall disable all interrupts by resetting RCC clock interrupt register (cir).(i.e., cir of M\_RCC to M\_HEX\_ZERO)

### 11.12.2 RccHseConfig

Low Level Design Details about CSU RccHseConfig will follow in the sub sections.

#### 11.12.2.1 Brief Description

The function RccHseConfig configures the External High Speed oscillator (HSE).

#### 11.12.2.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 11.12.2.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 11.12.2.4 Parameter list (Input/Output)

Inputs: T\_UINT8 rcc\_hse - Specifies the new state of the HSE

Outputs: None

#### 11.12.2.5 Return Value

None

#### 11.12.2.6 Other CSUs called by this CSU

None

#### 11.12.2.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to RccHseConfig

##### 11.12.2.7.1 daulibstm32f4xxrcc-RccHseConfig-LLR-001

Requirement ID: H398-LLD-GWY-FNC-5317

The function shall reset HSEON and HSEBYP bits in 3rd byte of RCC clock control register (cr) and configure with new configuration rcc\_hse.(i.e., pointer to M\_CR\_BYTE3\_ADDRESS to M\_RCC\_HSE\_OFF and pointer to M\_CR\_BYTE3\_ADDRESS to rcc\_hse).

### 11.12.3 RccWaitForHseStartUp

Low Level Design Details about CSU RccWaitForHseStartUp will follow in the sub sections.

#### 11.12.3.1 Brief Description

This function waits for HSE start-up.

#### 11.12.3.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

11.12.3.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 11.12.3.4 Parameter list (Input/Output)

Inputs: None

Outputs: None

#### 11.12.3.5 Return Value

T\_ERROR\_STATUS - Returns error status.

#### 11.12.3.6 Other CSUs called by this CSU

None

#### 11.12.3.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to RccWaitForHseStartUp

##### 11.12.3.7.1 daulibstm32f4xxrcc-RccWaitForHseStartUp-LLR-001

Requirement ID: H398-LLD-GWY-FNC-5326

The function shall wait till HSE is ready and increments the counter. When Time out is reached as below:

* Loop till the loop count is not equal to HSE\_STARTUP\_TIMEOUT AND Return value of the function RccGetFlagStatus called with the parameter M\_RCC\_FLAG\_HSERDY is equal to RESET.

##### 11.12.3.7.2 daulibstm32f4xxrcc-RccWaitForHseStartUp-LLR-002

Requirement ID: H398-LLD-GWY-FNC-5327

The function shall return SUCCESS when RCC flag is SET i.e return value of the function RccGetFlagStatus with parameter M\_RCC\_FLAG\_HSERDY is not equal to RESET.

##### 11.12.3.7.3 daulibstm32f4xxrcc-RccWaitForHseStartUp-LLR-003

Requirement ID: H398-LLD-GWY-FNC-5328

The function shall return ERROR when RCC flag is reset i.e return value of the function RccGetFlagStatus with parameter M\_RCC\_FLAG\_HSERDY is equal to RESET and returns status.

### 11.12.4 RccPllConfig

Low Level Design Details about CSU RccPllConfig will follow in the sub sections.

#### 11.12.4.1 Brief Description

The function RccPllConfig configures the main PLL clock source, multiplication and division factors.

#### 11.12.4.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 11.12.4.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 11.12.4.4 Parameter list (Input/Output)

Inputs: T\_UINT32 rcc\_pll\_source - Specifies the PLL entry clock source.

T\_UINT32 pll\_m - Specifies the division factor for PLL VCO input clock

T\_UINT32 pll\_n - Specifies the multiplication factor for PLL VCO output clock

T\_UINT32 pll\_p - Specifies the division factor for main system clock (SYSCLK)

T\_UINT32 pll\_q - Specifies the division factor for OTG FS, SDIO and RNG clocks

Outputs: None

#### 11.12.4.5 Return Value

None

#### 11.12.4.6 Other CSUs called by this CSU

None

#### 11.12.4.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to RccPllConfig

##### 11.12.4.7.1 daulibstm32f4xxrcc-RccPllConfig-LLR-001

Requirement ID: H398-LLD-GWY-FNC-5779

The function shall configure RCC PLL configuration register with pll\_m, pll\_n, pll\_p, pll\_q and rcc\_pll\_source.(i.e., pllcfgr of M\_RCC to Bitwise OR with pll\_m, (pll\_n left shift of M\_SHIFT\_BY\_6), (pll\_p right shift of M\_SHIFT\_1) minus M\_ONE left shift of M\_SHIFT\_BY\_16, rcc\_pll\_source, (pll\_q left shift of M\_SHIFT\_24))

### 11.12.5 RccPllCmd

Low Level Design Details about CSU RccPllCmd will follow in the sub sections.

#### 11.12.5.1 Brief Description

The function RccPllCmd enables or disables the main PLL.

#### 11.12.5.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 11.12.5.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 11.12.5.4 Parameter list (Input/Output)

Inputs: T\_FUNCTIONAL\_STATE new\_state - New state of the main PLL.

Outputs: None

#### 11.12.5.5 Return Value

None

#### 11.12.5.6 Other CSUs called by this CSU

None

#### 11.12.5.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to RccPllCmd

##### 11.12.5.7.1 daulibstm32f4xxrcc-RccPllCmd-LLR-001

Requirement ID: H398-LLD-GWY-FNC-5345

The function shall configure PLLON bit of RCC clock control register with new\_state(i.e., pointer to M\_CR\_PLLON\_BB is set to new state).

### 11.12.6 RccSysClkConfig

Low Level Design Details about CSU RccSysClkConfig will follow in the sub sections.

#### 11.12.6.1 Brief Description

The function RccSysClkConfig configures the system clock (SYSCLK).

#### 11.12.6.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 11.12.6.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 11.12.6.4 Parameter list (Input/Output)

Inputs: T\_UINT32 rcc\_sysclk\_source - specifies the clock source used as system clock.

Outputs: None

#### 11.12.6.5 Return Value

None

#### 11.12.6.6 Other CSUs called by this CSU

None

#### 11.12.6.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to RccSysClkConfig

##### 11.12.6.7.1 daulibstm32f4xxrcc-RccSysClkConfig-LLR-001

Requirement ID: H398-LLD-GWY-FNC-5778

The function shall clear the System clock switch bits (bit 0 and 1) in RCC clock configuration register (cfgr) and configure with clock source used as system clock (rcc\_sysclk\_source)(i.e., sets the temporary register to cfgr of M\_RCC, bitwise AND with Negation of M\_RCC\_CFGR\_SW, bitwise OR with rcc\_sysclk\_source).

And store the new value(i.e., set the cfgr of M\_RCC to temporary register)

### 11.12.7 RccGetSysClkSource

Low Level Design Details about CSU RccGetSysClkSource will follow in the sub sections.

#### 11.12.7.1 Brief Description

The function RccGetSysClkSource returns the clock source used as system clock.

#### 11.12.7.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 11.12.7.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 11.12.7.4 Parameter list (Input/Output)

Inputs: None

Outputs: None

#### 11.12.7.5 Return Value

T\_UINT8 - Returns the clock source used as system clock.

#### 11.12.7.6 Other CSUs called by this CSU

None

#### 11.12.7.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to RccGetSysClkSource

##### 11.12.7.7.1 daulibstm32f4xxrcc-RccGetSysClkSource-LLR-001

Requirement ID: H398-LLD-GWY-FNC-5362

The function shall return the clock source used as system clock by extracting the SWS0 and SWS1 bits in RCC clock configuration register (cfgr).(i.e., returns the cfgr of M\_RCC bitwise AND with M\_RCC\_CFGR\_SWS)

### 11.12.8 RccHclkConfig

Low Level Design Details about CSU RccHclkConfig will follow in the sub sections.

#### 11.12.8.1 Brief Description

The function RccHclkConfig configures the AHB clock (HCLK).

#### 11.12.8.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 11.12.8.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 11.12.8.4 Parameter list (Input/Output)

Inputs: T\_UINT32 rcc\_sysclk - Defines the AHB clock divider. This clock is derived from the system clock (SYSCLK).

Outputs: None

#### 11.12.8.5 Return Value

None

#### 11.12.8.6 Other CSUs called by this CSU

None

#### 11.12.8.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to RccHclkConfig

##### 11.12.8.7.1 daulibstm32f4xxrcc-RccHclkConfig-LLR-001

Requirement ID: H398-LLD-GWY-FNC-5371

The function shall clear the AHB prescaler bits (bit 4 to 7) in RCC clock configuration register (cfgr) and configure with AHB clock divider rcc\_sysclk.(i.e., sets the temporary register to cfgr of M\_RCC, bitwise AND with not of M\_RCC\_CFGR\_HPRE, bitwise OR with rcc\_sysclk).

And stores the new value(i.e., cfgr of M\_RCC to temporary register)

### 11.12.9 RccPclk1Config

Low Level Design Details about CSU RccPclk1Config will follow in the sub sections.

#### 11.12.9.1 Brief Description

The function RccPclk1Config configures the Low Speed APB clock (PCLK1).

#### 11.12.9.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 11.12.9.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 11.12.9.4 Parameter list (Input/Output)

Inputs: T\_UINT32 rcc\_hclk - Defines the APB1 clock divider. This clock is derived from the AHB clock (HCLK).

Outputs: None

#### 11.12.9.5 Return Value

None

#### 11.12.9.6 Other CSUs called by this CSU

None

#### 11.12.9.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to RccPclk1Config

##### 11.12.9.7.1 daulibstm32f4xxrcc-RccPclk1Config-LLR-001

Requirement ID: H398-LLD-GWY-FNC-5380

The function shall clear the APB Low speed prescaler bits (bit 10 to 12) in RCC clock configuration register (cfgr) and configure with APB1 clock divider rcc\_hclk.(i.e., sets the temporary register to cfgr of M\_RCC, Bitwise AND with not of M\_RCC\_CFGR\_PPRE1, Bitwise OR with rcc\_hclk ).

And stores the new value(i.e., cfgr of M\_RCC to temporary register).

### 11.12.10 RccPclk2Config

Low Level Design Details about CSU RccPclk2Config will follow in the sub sections.

#### 11.12.10.1 Brief Description

The function RccPclk2Config configures the High Speed APB clock (PCLK2).

#### 11.12.10.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 11.12.10.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 11.12.10.4 Parameter list (Input/Output)

Inputs: T\_UINT32 rcc\_hclk - Defines the APB2 clock divider. This clock is derived from the AHB clock (HCLK).

Outputs: None

#### 11.12.10.5 Return Value

None

#### 11.12.10.6 Other CSUs called by this CSU

None

#### 11.12.10.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to RccPclk2Config

##### 11.12.10.7.1 daulibstm32f4xxrcc-RccPclk2Config-LLR-001

Requirement ID: H398-LLD-GWY-FNC-5389

The function shall clear the APB high-speed prescaler bits (bit 13 to 15) in RCC clock configuration register(cfgr) and write APB2 clock divider 'rcc\_hclk' value into APB high-speed prescaler bits of cfgr.(i.e., sets the temporary register to cfgr of M\_RCC, Bitwise AND with not of M\_RCC\_CFGR\_PPRE2, Bitwise OR with rcc\_hclk left shift of M\_SHIFT\_3)

And stores the new value(cfgr of M\_RCC to temporary register).

### 11.12.11 RccGetClocksFreq

Low Level Design Details about CSU RccGetClocksFreq will follow in the sub sections.

#### 11.12.11.1 Brief Description

The function RccGetClocksFreq returns the frequencies of different on chip clocks; SYSCLK, HCLK, PCLK1 and PCLK2.

#### 11.12.11.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 11.12.11.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 11.12.11.4 Parameter list (Input/Output)

Inputs : T\_RCC\_CLOCKS\* rcc\_clocks - Pointer to a T\_RCC\_CLOCKS structure which will hold the clocks frequencies.

Outputs: T\_RCC\_CLOCKS\* rcc\_clocks - Pointer to a T\_RCC\_CLOCKS structure which will hold the clocks frequencies.

#### 11.12.11.5 Return Value

None

#### 11.12.11.6 Other CSUs called by this CSU

None

#### 11.12.11.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to RccGetClocksFreq

##### 11.12.11.7.1 daulibstm32f4xxrcc-RccGetClocksFreq-LLR-001

Requirement ID: H398-LLD-GWY-FNC-5398

The function shall find the sysclk source by extracting 4 LSB bits of RCC clock configuration register (cfgr)

(i.e., cfgr of M\_RCC Bitwise AND with M\_RCC\_CFGR\_SWS)

##### 11.12.11.7.2 daulibstm32f4xxrcc-RccGetClocksFreq-LLR-002

Requirement ID: H398-LLD-GWY-FNC-5399

The function shall set SYSCLK frequency of Rcc clocks to HSI\_VALUE when sysclk source value is M\_HEX\_ZERO.

##### 11.12.11.7.3 daulibstm32f4xxrcc-RccGetClocksFreq-LLR-003

Requirement ID: H398-LLD-GWY-FNC-5400

The function shall set SYSCLK frequency of Rcc clocks to HSE\_VALUE when sysclk source value is M\_HEX\_FOUR.

##### 11.12.11.7.4 daulibstm32f4xxrcc-RccGetClocksFreq-LLR-004

Requirement ID: H398-LLD-GWY-FNC-5401

The function shall do the following when sysclk source value is M\_HEX\_EIGHT:

a) Extract pll source(PLLSRC bit) from RCC PLL configuration register(pllcfgr)(i.e., pllcfgr of M\_RCC Bitwise AND with M\_RCC\_PLLCFGR\_PLLSRC right shift of M\_SHIFT\_22).

b) Extract pll\_m (pll\_m bits 0 to 5) from RCC PLL configuration register(pllcfgr)(i.e., pllcfgr of M\_RCC Bitwise AND with M\_RCC\_PLLCFGR\_PLLM).

c) when pll source is not M\_ZERO, compute the VCO output frequency using HSE as PLL clock source(i.e., sets the pllvco to (HSE\_VALUE divide by pll\_m) multiply with (pllcfgr of M\_RCC Bitwise AND with M\_RCC\_PLLCFGR\_PLLN) right shift of M\_SHIFT\_BY\_6)

d) when pll source is M\_ZERO, compute the VCO output frequency using HSI as PLL clock source(i.e., set the pllvco to (HSI\_VALUE divide by pll\_m) multiply with (pllcfgr of M\_RCC Bitwise AND with M\_RCC\_PLLCFGR\_PLLN) right shift of M\_SHIFT\_BY\_6)

e) Compute PLL output clock frequency using VCO output frequency and set it to Sysclk\_frequency of rcc\_clocks(i.e., sets the pll\_p to (((pllcfgr of M\_RCC Bitwise AND with M\_RCC\_PLLCFGR\_PLLP) right shift of M\_SHIFT\_BY\_16) plus M\_ONE) multiply with M\_TWO and sets the Sysclk\_frequency of rcc\_clocks to pllvco divide by pll\_p).

##### 11.12.11.7.5 daulibstm32f4xxrcc-RccGetClocksFreq-LLR-005

Requirement ID: H398-LLD-GWY-FNC-5402

The function shall set SYSCLK frequency of Rcc clocks with HSI\_VALUE if sysclk source value is other than M\_HEX\_ZERO, M\_HEX\_FOUR and M\_HEX\_EIGHT.

##### 11.12.11.7.6 daulibstm32f4xxrcc-RccGetClocksFreq-LLR-006

Requirement ID: H398-LLD-GWY-FNC-5403

The function shall do the following:

a) Extract the HPRE bit from RCC clock configuration register (cfgr) i.e, sets the sysclk source to cfgr of M\_RCC Bitwise AND with M\_RCC\_CFGR\_HPRE

b) Get the corresponding prescaler value from Apbahb prescaler table (sysclk source is right shift of M\_SHIFT\_4).

c) Set the hclk\_frequency of rcc\_clocks to sysclk\_frequency of rcc\_clocks right shifted by prese (extracted using prescaler value obtained Apbahb\_presc\_table with index as sysclk\_src).

##### 11.12.11.7.7 daulibstm32f4xxrcc-RccGetClocksFreq-LLR-007

Requirement ID: H398-LLD-GWY-FNC-5404

The function shall do the following:

a) Extract the PPRE1 bit from RCC clock configuration register (cfgr) i.e, sets the sysclk source to cfgr of M\_RCC bitwise AND with M\_RCC\_CFGR\_PPRE1 and right shifted by M\_SHIFT\_10.

b) Get the corresponding prescaler value from Apbahb prescaler table with index as sysclk\_src

c) Set the pclk\_1\_frequency of rcc\_clocks to hclk\_frequency of rcc\_clocks extracted using prescaler value obtained.

##### 11.12.11.7.8 daulibstm32f4xxrcc-RccGetClocksFreq-LLR-008

Requirement ID: H398-LLD-GWY-FNC-5405

The function shall do the following:

a) Extract the PPRE2 bit from RCC clock configuration register(cfgr)(i.e., sets the sysclk src to cfgr of M\_RCC bitwise AND M\_RCC\_CFGR\_PPRE2 and right shift of M\_SHIFT\_10)

b) Get the corresponding prescaler value from Apbahb prescaler table with index as sysclk\_src

c) Set the pclk\_2\_frequency of rcc\_clocks with hclk\_frequency of rcc\_clocks extracted using prescaler value obtained.

### 11.12.12 RccAhb1PeriphClockCmd

Low Level Design Details about CSU RccAhb1PeriphClockCmd will follow in the sub sections.

#### 11.12.12.1 Brief Description

The function RccGetClocksFreq enables or disables the AHB1 peripheral clock.

#### 11.12.12.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 11.12.12.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 11.12.12.4 Parameter list (Input/Output)

Inputs: T\_UINT32 rcc\_ahb1\_periph - Specifies the AHB1 peripheral to gate its clock.

T\_FUNCTIONAL\_STATE new\_state - New state of the specified peripheral clock.

Outputs: None

#### 11.12.12.5 Return Value

None

#### 11.12.12.6 Other CSUs called by this CSU

None

#### 11.12.12.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to RccAhb1PeriphClockCmd

##### 11.12.12.7.1 daulibstm32f4xxrcc-RccAhb1PeriphClockCmd-LLR-001

Requirement ID: H398-LLD-GWY-FNC-5414

The function shall configure RCC AHB1 peripheral clock enable register to set the bit for received peripheral rcc\_ahb1\_periph i.e, ahb1enr of M\_RCC Bitwise OR with rcc\_ahb1\_periph when new\_state is not equal to DISABLE.

##### 11.12.12.7.2 daulibstm32f4xxrcc-RccAhb1PeriphClockCmd-LLR-002

Requirement ID: H398-LLD-GWY-FNC-5415

The function shall configure RCC AHB1 peripheral clock enable register to reset the bit for received peripheral rcc\_ahb1\_periph(i.e., ahb1enr of M\_RCC Bitwise AND with not of rcc\_ahb1\_periph) when new\_state is equal to DISABLE.

### 11.12.13 RccAhb3PeriphClockCmd

Low Level Design Details about CSU RccAhb3PeriphClockCmd will follow in the sub sections.

#### 11.12.13.1 Brief Description

The function RccAhb3PeriphClockCmd enables or disables the AHB3 peripheral clock.

#### 11.12.13.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 11.12.13.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 11.12.13.4 Parameter list (Input/Output)

Inputs: T\_UINT32 rcc\_ahb3\_periph - Specifies the AHB3 peripheral to gates its clock.

T\_FUNCTIONAL\_STATE new\_state - New state of the specified peripheral clock.

Outputs: None

#### 11.12.13.5 Return Value

None

#### 11.12.13.6 Other CSUs called by this CSU

None

#### 11.12.13.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to RccAhb3PeriphClockCmd

##### 11.12.13.7.1 daulibstm32f4xxrcc-RccAhb3PeriphClockCmd-LLR-001

Requirement ID: H398-LLD-GWY-FNC-5424

The function shall configure RCC AHB3 peripheral clock enable register to set the bit for received peripheral rcc\_ahb3\_periph (i.e., set ahb3enr of M\_RCC Bitwise OR with rcc\_ahb3\_periph) when new state is equal to DISABLE.

##### 11.12.13.7.2 daulibstm32f4xxrcc-RccAhb3PeriphClockCmd-LLR-002

Requirement ID: H398-LLD-GWY-FNC-5425

The function shall configure RCC AHB3 peripheral clock enable register to reset the bit for received peripheral rcc\_ahb3\_periph (ahb3enr of M\_RCC Bitwise AND with not of rcc\_ahb3\_periph) when new state is equal to DISABLE.

### 11.12.14 RccApb1PeriphClockCmd

Low Level Design Details about CSU RccApb1PeriphClockCmd will follow in the sub sections.

#### 11.12.14.1 Brief Description

The function RccApb1PeriphClockCmd enables or disables the Low Speed APB (APB1) peripheral clock.

#### 11.12.14.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 11.12.14.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 11.12.14.4 Parameter list (Input/Output)

Inputs: T\_UINT32 rcc\_apb1\_periph - Specifies the APB1 peripheral to gate its clock.

T\_FUNCTIONAL\_STATE new\_state - New state of the specified peripheral clock.

Outputs: None

#### 11.12.14.5 Return Value

None

#### 11.12.14.6 Other CSUs called by this CSU

None

#### 11.12.14.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to RccApb1PeriphClockCmd

##### 11.12.14.7.1 daulibstm32f4xxrcc-RccApb1PeriphClockCmd-LLR-001

Requirement ID: H398-LLD-GWY-FNC-5434

The function shall configure RCC APB1 peripheral clock enable register to set the bit for received peripheral rcc\_apb1\_periph (i.e., apb1enr of M\_RCC Bitwise OR with rcc\_apb1\_periph) when new state is not equal to DISABLE.

##### 11.12.14.7.2 daulibstm32f4xxrcc-RccApb1PeriphClockCmd-LLR-002

Requirement ID: H398-LLD-GWY-FNC-5435

The function shall configure RCC APB1 peripheral clock enable register to reset the bit for received peripheral rcc\_apb1\_periph (i.e., apb1enr of M\_RCC Bitwise AND with not of rcc\_apb1\_periph) when new state is DISABLE.

### 11.12.15 RccGetFlagStatus

Low Level Design Details about CSU RccGetFlagStatus will follow in the sub sections.

#### 11.12.15.1 Brief Description

The function RccGetFlagStatus checks whether the specified RCC flag is set or not.

#### 11.12.15.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 11.12.15.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 11.12.15.4 Parameter list (Input/Output)

Inputs: T\_UINT8 rcc\_flag - Specifies the flag to check.

Outputs: None

#### 11.12.15.5 Return Value

T\_FLAG\_STATUS - The new state of rcc\_flag (SET or RESET)

#### 11.12.15.6 Other CSUs called by this CSU

None

#### 11.12.15.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to RccGetFlagStatus

##### 11.12.15.7.1 daulibstm32f4xxrcc-RccGetFlagStatus-LLR-001

Requirement ID: H398-LLD-GWY-FNC-5444

The function shall identify the RCC register (cr, BDCR or CSR) in which the rcc\_flag resides and check whether the specified rcc\_flag is set or not in the respective register. If the flag is set return status as SET else RESET.(i.e., sets the status register to cr, bdcr, csr of M\_RCC when the register index(rcc\_flag right shift of M\_SHIFT\_5) is equal to M\_ONE, M\_TWO, and other than M\_ONE, M\_TWO and sets the bit status to SET or RESET(i.e., returns bit\_status) when RESET is not of (status register Bitwise AND M\_ONE left shift of register index(rcc\_flag Bitwise AND M\_FLAG\_MASK))

### 11.12.16 RccApb1PeriphResetCmd

Low Level Design Details about CSU RccApb1PeriphResetCmd will follow in the sub sections.

#### 11.12.16.1 Brief Description

The function RccApb1PeriphResetCmd forces or releases Low Speed APB (APB1) peripheral reset.

#### 11.12.16.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 11.12.16.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 11.12.16.4 Parameter list (Input/Output)

Inputs: T\_UINT32 rcc\_apb1\_periph - Specifies the APB1 peripheral to reset.

T\_FUNCTIONAL\_STATE new\_state - new state of the specified peripheral clock.

Outputs: None

#### 11.12.16.5 Return Value

None

#### 11.12.16.6 Other CSUs called by this CSU

None

#### 11.12.16.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to RccApb1PeriphResetCmd

##### 11.12.16.7.1 daulibstm32f4xxrcc-RccApb1PeriphResetCmd-LLR-001

Requirement ID: H398-LLD-GWY-FNC-5453

The function shall configure RCC APB1 peripheral reset register to set the bit for received peripheral rcc\_apb1\_periph(apb1rstr of M\_RCC Bitwise OR with rcc\_apb1\_periph) when new state is not equal to DISABLE.

##### 11.12.16.7.2 daulibstm32f4xxrcc-RccApb1PeriphResetCmd-LLR-002

Requirement ID: H398-LLD-GWY-FNC-5454

The function shall configure RCC APB1 peripheral reset register to reset the bit for received peripheral rcc\_apb1\_periph(apb1rstr of M\_RCC Bitwise AND with not of rcc\_apb1\_periph) when new state is equal to DISABLE.

### 11.12.17 RccApb2PeriphResetCmd

Low Level Design Details about CSU RccApb2PeriphResetCmd will follow in the sub sections.

#### 11.12.17.1 Brief Description

The function RccApb2PeriphResetCmd forces or releases High Speed APB (APB2) peripheral reset.

#### 11.12.17.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 11.12.17.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 11.12.17.4 Parameter list (Input/Output)

Inputs: T\_UINT32 rcc\_apb2\_periph - Specifies the APB2 peripheral to reset.

T\_FUNCTIONAL\_STATE new\_state - New state of the specified peripheral clock.

Outputs: None

#### 11.12.17.5 Return Value

None

#### 11.12.17.6 Other CSUs called by this CSU

None

#### 11.12.17.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to RccApb2PeriphResetCmd

##### 11.12.17.7.1 daulibstm32f4xxrcc-RccApb2PeriphResetCmd-LLR-001

Requirement ID: H398-LLD-GWY-FNC-5463

The function shall configure RCC APB2 peripheral reset register to set the bit for received peripheral rcc\_apb2\_periph(i.e., apb2rstr of M\_RCC Bitwise OR with rcc\_apb2\_periph) when new state is not equal to DISABLE.

##### 11.12.17.7.2 daulibstm32f4xxrcc-RccApb2PeriphResetCmd-LLR-002

Requirement ID: H398-LLD-GWY-FNC-5464

The function shall configure RCC APB2 peripheral reset register to reset the bit for received peripheral rcc\_apb2\_periph(apb2rstr of M\_RCC Bitwise AND with not of rcc\_apb2\_periph) when new state is equal to DISABLE.

### 11.12.18 RccApb2PeriphClockCmd

Low Level Design Details about CSU RccApb2PeriphClockCmd will follow in the sub sections.

#### 11.12.18.1 Brief Description

This function enables or disables the High Speed APB (APB2) peripheral clock.

#### 11.12.18.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 11.12.18.3 List of global variables accessed and modified

Accessed : None

Modified : None

#### 11.12.18.4 Parameter list (Input/Output)

Inputs: T\_UINT32 rcc\_apb2\_periph - specifies the APB2 peripheral to gate its clock.

T\_FUNCTIONAL\_STATE new\_state - new state of the specified peripheral clock.

Outputs: None

#### 11.12.18.5 Return Value

None

#### 11.12.18.6 Other CSUs called by this CSU

None

#### 11.12.18.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to RccApb2PeriphClockCmd

##### 11.12.18.7.1 daulibstm32f4xxrcc-RccApb2PeriphClockCmd-LLR-001

Requirement ID: H398-LLD-GWY-FNC-5473

The function shall configure RCC APB2 peripheral clock enable register to set the bit for received peripheral rcc\_apb2\_periph(i.e., apb2enr of M\_RCC Bitwise OR with rcc\_apb2\_periph) when new state is not equal to DISABLE.

##### 11.12.18.7.2 daulibstm32f4xxrcc-RccApb2PeriphClockCmd-LLR-002

Requirement ID: H398-LLD-GWY-FNC-5474

The function shall configure RCC APB2 peripheral reset register to reset the bit for received peripheral rcc\_apb2\_periph(apb2enr of M\_RCC Bitwise AND with not of rcc\_apb2\_periph) when new state is equal to DISABLE.

## 11.13 daulibstm32f4xxtim

This module provides firmware functions to manage the following functionalities of the TIM peripheral:

* TimeBase management
* Interrupts, DMA and flags management
* Synchronization management

### 11.13.1 TimPrescalerConfig

Low Level Design Details about CSU TimPrescalerConfig will follow in the sub sections.

#### 11.13.1.1 Brief Description

The function TimPrescalerConfig configures the Timer prescaler and event generation.

#### 11.13.1.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 11.13.1.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 11.13.1.4 Parameter list (Input/Output)

Inputs: T\_UINT16 prescaler - Specifies the prescaler Register value.

T\_UINT16 tim\_psc\_reloadmode - Specifies the TIM prescaler Reload mode.

Outputs:T\_TIM\* tim\_x - TIM peripheral,where x can be 1 to 14 to select the TIM peripheral.

#### 11.13.1.5 Return Value

None

#### 11.13.1.6 Other CSUs called by this CSU

None

#### 11.13.1.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TimPrescalerConfig.

##### 11.13.1.7.1 daulibstm32f4xxtim-TimPrescalerConfig-LLR-001

Requirement ID: H398-LLD-GWY-FNC-5498

The function shall configure prescaler register (psc) of tim\_x with the value prescaler i.e, set psc of tim\_x to prescaler.

##### 11.13.1.7.2 daulibstm32f4xxtim-TimPrescalerConfig-LLR-002

Requirement ID: H398-LLD-GWY-FNC-5499

The function shall configure event generation register (egr) of tim\_x i.e, set egr of tim\_x to tim\_psc\_reloadmode.

### 11.13.2 TimSetAutoReload

Low Level Design Details about CSU TimSetAutoReload will follow in the sub sections.

#### 11.13.2.1 Brief Description

The function TimSetAutoReload sets the timer auto reload register value.

#### 11.13.2.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 11.13.2.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 11.13.2.4 Parameter list (Input/Output)

Inputs: T\_UINT32 autoreload - Specifies the auto reload register new value.

Outputs: T\_TIM\* tim\_x - TIM peripheral, where x can be 1 to 14 to select the TIM peripheral.

#### 11.13.2.5 Return Value

None

#### 11.13.2.6 Other CSUs called by this CSU

None

#### 11.13.2.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TimSetAutoReload

##### 11.13.2.7.1 daulibstm32f4xxtim-TimSetAutoReload-LLR-001

Requirement ID: H398-LLD-GWY-FNC-5508

The function shall configure auto-reload register (arr) with auto reload value i.e, set arr of tim\_x to autoreload.

### 11.13.3 TimCmd

Low Level Design Details about CSU TimCmd will follow in the sub sections.

#### 11.13.3.1 Brief Description

The function TimCmd enables or disables the specified TIM peripheral.

#### 11.13.3.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 11.13.3.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 11.13.3.4 Parameter list (Input/Output)

Inputs: T\_FUNCTIONAL\_STATE new\_state - New state of the tim\_x peripheral.

T\_TIM\* tim\_x - TIM peripheral, where x can be 1 to 14 to select the TIM peripheral.

Outputs: T\_TIM\* tim\_x - TIM peripheral, where x can be 1 to 14 to select the TIM peripheral.

#### 11.13.3.5 Return Value

None

#### 11.13.3.6 Other CSUs called by this CSU

None

#### 11.13.3.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TimCmd

##### 11.13.3.7.1 daulibstm32f4xxtim-TimCmd-LLR-001

Requirement ID: H398-LLD-GWY-FNC-5517

The function shall configure control register 1 of timer to enable the counter (set bit 0 to 1) i.e, set cr1 of tim\_x to (cr1 of tim\_x bitwise OR with M\_TIM\_CR1\_CEN) when new\_state is ENABLE.

##### 11.13.3.7.2 daulibstm32f4xxtim-TimCmd-LLR-002

Requirement ID: H398-LLD-GWY-FNC-5518

The function shall configure control register 1 of timer to disable the counter (set bit 0 to 0) i.e, set cr1 of tim\_x to (cr1 of tim\_x bitwise AND with negated value of M\_TIM\_CR1\_CEN) when new\_state is DISABLE.

### 11.13.4 TimItConfig

Low Level Design Details about CSU TimItConfig will follow in the sub sections.

#### 11.13.4.1 Brief Description

The function TimItConfig enables or disables the specified TIM interrupts.

#### 11.13.4.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 11.13.4.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 11.13.4.4 Parameter list (Input/Output)

Inputs: T\_FUNCTIONAL\_STATE new\_state - New state of the tim\_x peripheral.

T\_TIM\* tim\_x - TIM peripheral, where x can be 1 to 14 to select the TIM peripheral.

T\_UINT16 tim\_it - specifies the TIM interrupts sources to be enabled or disabled.

Outputs: T\_TIM\* tim\_x - TIM peripheral, where x can be 1 to 14 to select the TIM peripheral.

#### 11.13.4.5 Return Value

None

#### 11.13.4.6 Other CSUs called by this CSU

None

#### 11.13.4.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TimItConfig.

##### 11.13.4.7.1 daulibstm32f4xxtim-TimItConfig-LLR-001

Requirement ID: H398-LLD-GWY-FNC-5527

The function shall configure DMA/Interrupt enable register to enable the received Interrupt source i.e, set dier of tim\_x to (dier of tim\_x bitwise OR with tim\_it) when new\_state is ENABLE.

##### 11.13.4.7.2 daulibstm32f4xxtim-TimItConfig-LLR-002

Requirement ID: H398-LLD-GWY-FNC-5528

The function shall configure DMA/Interrupt enable register to disable the received Interrupt source source i.e, set dier of tim\_x to (dier of tim\_x bitwise AND with negated value of tim\_it) when new\_state is DISABLE.

### 11.13.5 TimSetIC1Prescaler

Low Level Design Details about CSU TimSetIC1Prescaler will follow in the sub sections.

#### 11.13.5.1 Brief Description

The function TimSetIC1Prescaler sets the tim\_x Input Capture 1 prescaler.

#### 11.13.5.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 11.13.5.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 11.13.5.4 Parameter list (Input/Output)

Inputs: T\_UINT16 tim\_icpsc - specifies the Input Capture1 prescaler new value

T\_TIM\* tim\_x - where x can be 1 to 14 except 6 and 7, to select the TIM peripheral

Outputs: T\_TIM\* tim\_x - where x can be 1 to 14 except 6 and 7, to select the TIM peripheral

#### 11.13.5.5 Return Value

None

#### 11.13.5.6 Other CSUs called by this CSU

None

#### 11.13.5.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TimSetIC1Prescaler.

##### 11.13.5.7.1 daulibstm32f4xxtim-TimSetIC1Prescaler-LLR-001

Requirement ID: H398-LLD-GWY-FNC-5537

The function shall reset the IC1PSC Bits by setting ccmr1 of tim\_x with Bitwise AND of ccmr1 of tim\_x and negation of M\_TIM\_CCMR1\_IC1PSC

##### 11.13.5.7.2 daulibstm32f4xxtim-TimSetIC1Prescaler-LLR-002

Requirement ID: H398-LLD-GWY-FNC-5538

The function shall set the IC1PSC value by setting ccmr1 of tim\_x with Bitwise OR of ccmr1 of tim\_x and tim\_icpsc

### 11.13.6 TimSetIC2Prescaler

Low Level Design Details about CSU TimSetIC2Prescaler will follow in the sub sections.

#### 11.13.6.1 Brief Description

The function TimSetIC2Prescaler sets the tim\_x Input Capture 2 prescaler.

#### 11.13.6.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 11.13.6.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 11.13.6.4 Parameter list (Input/Output)

Inputs: T\_UINT16 tim\_icpsc - specifies the Input Capture2 prescaler new value

T\_TIM\* tim\_x - where x can be 1, 2, 3, 4, 5, 8, 9 or 12 to select the TIM peripheral

Outputs: T\_TIM\* tim\_x - where x can be 1, 2, 3, 4, 5, 8, 9 or 12 to select the TIM peripheral

#### 11.13.6.5 Return Value

None

#### 11.13.6.6 Other CSUs called by this CSU

None

#### 11.13.6.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TimSetIC2Prescaler.

##### 11.13.6.7.1 daulibstm32f4xxtim-TimSetIC2Prescaler-LLR-001

Requirement ID: H398-LLD-GWY-FNC-5547

The function shall reset the IC2PSC Bits by setting ccmr1 of tim\_x with Bitwise AND of ccmr1 of tim\_x and negation of M\_TIM\_CCMR1\_IC2PSC

##### 11.13.6.7.2 daulibstm32f4xxtim-TimSetIC2Prescaler-LLR-002

Requirement ID: H398-LLD-GWY-FNC-5548

The function shall set the IC2PSC value by setting ccmr1 of tim\_x with Bitwise OR of ccmr1 of tim\_x and (bit shift of tim\_icpsc to left by M\_GET\_IC2PSC\_POS\_IN\_CCMR1)

### 11.13.7 TimSetIC3Prescaler

Low Level Design Details about CSU TimSetIC3Prescaler will follow in the sub sections.

#### 11.13.7.1 Brief Description

The function TimSetIC3Prescaler sets the tim\_x Input Capture 3 prescaler.

#### 11.13.7.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 11.13.7.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 11.13.7.4 Parameter list (Input/Output)

Inputs: T\_UINT16 tim\_icpsc - specifies the Input Capture3 prescaler new value

T\_TIM\* tim\_x - where x can be 1, 2, 3, 4, 5, 8, 9 or 12 to select the TIM peripheral

Outputs: T\_TIM\* tim\_x - where x can be 1, 2, 3, 4, 5, 8, 9 or 12 to select the TIM peripheral

#### 11.13.7.5 Return Value

None

#### 11.13.7.6 Other CSUs called by this CSU

None

#### 11.13.7.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TimSetIC3Prescaler.

##### 11.13.7.7.1 daulibstm32f4xxtim-TimSetIC3Prescaler-LLR-001

Requirement ID: H398-LLD-GWY-FNC-5557

The function shall reset the IC3PSC Bits by setting ccmr2 of tim\_x with Bitwise AND of ccmr2 of tim\_x and negation of M\_TIM\_CCMR2\_IC3PSC

##### 11.13.7.7.2 daulibstm32f4xxtim-TimSetIC3Prescaler-LLR-002

Requirement ID: H398-LLD-GWY-FNC-5558

The function shall set the IC3PSC value by setting ccmr2 of tim\_x with Bitwise OR of ccmr2 of tim\_x and tim\_icpsc

### 11.13.8 TimSetIC4Prescaler

Low Level Design Details about CSU TimSetIC4Prescaler will follow in the sub sections.

#### 11.13.8.1 Brief Description

The function TimSetIC4Prescaler sets the tim\_x Input Capture 4 prescaler.

#### 11.13.8.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 11.13.8.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 11.13.8.4 Parameter list (Input/Output)

Inputs: T\_UINT16 tim\_icpsc - specifies the Input Capture4 prescaler new value

T\_TIM\* tim\_x - where x can be 1, 2, 3, 4, 5 or 8 to select the TIM peripheral

Outputs: T\_TIM\* tim\_x - where x can be 1, 2, 3, 4, 5 or 8 to select the TIM peripheral

#### 11.13.8.5 Return Value

None

#### 11.13.8.6 Other CSUs called by this CSU

None

#### 11.13.8.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TimSetIC4Prescaler.

##### 11.13.8.7.1 daulibstm32f4xxtim-TimSetIC4Prescaler-LLR-001

Requirement ID: H398-LLD-GWY-FNC-5567

The function shall reset the IC4PSC Bits by setting ccmr2 of tim\_x with Bitwise AND of ccmr2 of tim\_x and negation of M\_TIM\_CCMR2\_IC4PSC

##### 11.13.8.7.2 daulibstm32f4xxtim-TimSetIC4Prescaler-LLR-002

Requirement ID: H398-LLD-GWY-FNC-5568

The function shall set the IC4PSC value by setting ccmr2 of tim\_x with Bitwise OR of ccmr2 of tim\_x and (bit shift of tim\_icpsc to left by M\_GET\_IC4PSC\_POS\_IN\_CCMR2)

### 11.13.9 TimDmaCmd

Low Level Design Details about CSU TimDmaCmd will follow in the sub sections.

#### 11.13.9.1 Brief Description

The function TimDmaCmd enables or disables the tim\_x's DMA Requests.

#### 11.13.9.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 11.13.9.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 11.13.9.4 Parameter list (Input/Output)

Inputs: T\_UINT16 tim\_dmasource - specifies the DMA Request sources

T\_FUNCTIONAL\_STATE new\_state - state of DMA Request

T\_TIM\* tim\_x - where x can be 1, 2, 3, 4, 5, 6, 7 or 8 to select the TIM peripheral

Outputs: T\_TIM\* tim\_x - where x can be 1, 2, 3, 4, 5, 6, 7 or 8 to select the TIM peripheral

#### 11.13.9.5 Return Value

None

#### 11.13.9.6 Other CSUs called by this CSU

None

#### 11.13.9.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TimDmaCmd.

##### 11.13.9.7.1 daulibstm32f4xxtim-TimDmaCmd-LLR-001

Requirement ID: H398-LLD-GWY-FNC-5577

The function shall enable the DMA sources by setting dier of tim\_x with Bitwise OR of dier of tim\_x and tim\_dmasource when new\_state of DMA Request is not DISABLE

##### 11.13.9.7.2 daulibstm32f4xxtim-TimDmaCmd-LLR-002

Requirement ID: H398-LLD-GWY-FNC-5578

The function shall disable the DMA sources by setting dier of tim\_x with Bitwise AND of dier of tim\_x and negation of tim\_dmasource when new\_state of DMA Request is DISABLE

### 11.13.10 Ti1Config

Low Level Design Details about CSU Ti1Config will follow in the sub sections.

#### 11.13.10.1 Brief Description

The function Ti1Config configures the TI1 as Input.

#### 11.13.10.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 11.13.10.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 11.13.10.4 Parameter list (Input/Output)

Inputs: T\_TIM\* tim\_x - where x can be 1, 2, 3, 4, 5, 8, 9, 10, 11, 12, 13 or 14 to select the

TIM peripheral

T\_UINT16 tim\_ic\_polarity - The Input Polarity

T\_UINT16 tim\_ic\_selection - Specifies the input to be used

T\_UINT16 tim\_ic\_filter - Specifies the Input Capture Filter

Outputs: T\_TIM\* tim\_x - where x can be 1, 2, 3, 4, 5, 8, 9, 10, 11, 12, 13 or 14 to select the

TIM peripheral

#### 11.13.10.5 Return Value

None

#### 11.13.10.6 Other CSUs called by this CSU

None

#### 11.13.10.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to Ti1Config.

##### 11.13.10.7.1 daulibstm32f4xxtim-Ti1Config-LLR-001

Requirement ID: H398-LLD-GWY-FNC-5587

The function shall disable the Channel 1 i.e Reset the CC1E Bit by performing the following:

1. Set compare enable register i.e ccer of tim\_x with Bitwise AND of ccer of tim\_x and negation of M\_TIM\_CCER\_CC1E
2. Set temporary compare mode register 1 with ccmr1 of tim\_x (compare mode register 1)
3. Set temporary compare enable register with ccer of tim\_x (compare enable register)

##### 11.13.10.7.2 daulibstm32f4xxtim-Ti1Config-LLR-002

Requirement ID: H398-LLD-GWY-FNC-5588

The function shall set the filter by performing the following:

1. Set temporary compare mode register 1 with temporary compare mode register 1 Bitwise AND (negation of M\_TIM\_CCMR1\_CC1S) Bitwise AND (negation of M\_TIM\_CCMR1\_IC1F)
2. Set temporary compare mode register 1 with temporary compare mode register 1 Bitwise OR ( tim\_ic\_selection OR (bit shift of tim\_ic\_filter to left by M\_FOUR))

##### 11.13.10.7.3 daulibstm32f4xxtim-Ti1Config-LLR-003

Requirement ID: H398-LLD-GWY-FNC-5589

The function shall select the Polarity and set the CC1E Bit by performing the following:

1. Set temporary compare enable register with temporary compare enable register Bitwise AND (negation of (M\_TIM\_CCER\_CC1P Bitwise OR M\_TIM\_CCER\_CC1NP))
2. Set temporary compare enable register with temporary compare enable register Bitwise OR (tim\_ic\_polarity Bitwise OR M\_TIM\_CCER\_CC1E)

##### 11.13.10.7.4 daulibstm32f4xxtim-Ti1Config-LLR-004

Requirement ID: H398-LLD-GWY-FNC-5590

The function shall write to tim\_x CCMR1 and CCER registers by performing the following:

1. Set ccmr1 of tim\_x to temporary compare mode register 1
2. Set ccer of tim\_x to temporary compare enable register

### 11.13.11 Ti2Config

Low Level Design Details about CSU Ti2Config will follow in the sub sections.

#### 11.13.11.1 Brief Description

The function Ti2Config configures the TI2 as Input.

#### 11.13.11.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 11.13.11.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 11.13.11.4 Parameter list (Input/Output)

Inputs: T\_TIM\* tim\_x - where x can be 1, 2, 3, 4, 5, 8, 9 or 12 to select the

TIM peripheral

T\_UINT16 tim\_ic\_polarity - The Input Polarity

T\_UINT16 tim\_ic\_selection - Specifies the input to be used

T\_UINT16 tim\_ic\_filter - Specifies the Input Capture Filter

Outputs: T\_TIM\* tim\_x - where x can be 1, 2, 3, 4, 5, 8, 9 or 12 to select the

TIM peripheral

#### 11.13.11.5 Return Value

None

#### 11.13.11.6 Other CSUs called by this CSU

None

#### 11.13.11.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to Ti2Config.

##### 11.13.11.7.1 daulibstm32f4xxtim-Ti2Config-LLR-001

Requirement ID: H398-LLD-GWY-FNC-5599

The function shall disable the Channel 2 i.e reset the CC2E Bit by performing the following:

1. Set compare enable register i.e ccer of tim\_x with Bitwise AND of ccer of tim\_x and negation of M\_TIM\_CCER\_CC2E
2. Set temporary compare mode register 1 with ccmr1 of tim\_x (compare mode register 1)
3. Set temporary compare enable register with ccer of tim\_x (compare enable register)
4. Set select polarity to bit shift of tim\_ic\_polarity to left by M\_FOUR

##### 11.13.11.7.2 daulibstm32f4xxtim-Ti2Config-LLR-002

Requirement ID: H398-LLD-GWY-FNC-5600

The function shall set the filter by performing the following:

1. Set temporary compare mode register 1 with temporary compare mode register 1 Bitwise AND (negation of M\_TIM\_CCMR1\_CC2S) Bitwise AND (negation of M\_TIM\_CCMR1\_IC2F)
2. Set temporary compare mode register 1 with temporary compare mode register 1 Bitwise OR (bit shift of tim\_ic\_filter to left by M\_TWELVE)
3. Set temporary compare mode register 1 with temporary compare mode register 1 Bitwise OR (bit shift of tim\_ic\_selection to left by M\_GET\_IC\_SELECTION\_POS\_IN\_CCMR1)

##### 11.13.11.7.3 daulibstm32f4xxtim-Ti2Config-LLR-003

Requirement ID: H398-LLD-GWY-FNC-5601

The function shall select the Polarity and set the CC2E Bit by performing the following:

1. Set temporary compare enable register with temporary compare enable register Bitwise AND (negation of (M\_TIM\_CCER\_CC2P Bitwise OR M\_TIM\_CCER\_CC2NP))
2. Set temporary compare enable register with temporary compare enable register Bitwise OR (select polarity Bitwise OR M\_TIM\_CCER\_CC2E)

##### 11.13.11.7.4 daulibstm32f4xxtim-Ti2Config-LLR-004

Requirement ID: H398-LLD-GWY-FNC-5602

The function shall write to tim\_x CCMR1 and CCER registers by performing the following:

1. Set ccmr1 of tim\_x to temporary compare mode register 1
2. Set ccer of tim\_x to temporary compare enable register

### 11.13.12 Ti3Config

Low Level Design Details about CSU Ti3Config will follow in the sub sections.

#### 11.13.12.1 Brief Description

The function Ti3Config configures the TI3 as Input.

#### 11.13.12.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 11.13.12.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 11.13.12.4 Parameter list (Input/Output)

Inputs: T\_TIM\* tim\_x - where x can be 1, 2, 3, 4, 5 or 8 to select the

TIM peripheral

T\_UINT16 tim\_ic\_polarity - The Input Polarity

T\_UINT16 tim\_ic\_selection - Specifies the input to be used

T\_UINT16 tim\_ic\_filter - Specifies the Input Capture Filter

Outputs: T\_TIM\* tim\_x - where x can be 1, 2, 3, 4, 5 or 8 to select the

TIM peripheral

#### 11.13.12.5 Return Value

None

#### 11.13.12.6 Other CSUs called by this CSU

None

#### 11.13.12.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to Ti3Config.

##### 11.13.12.7.1 daulibstm32f4xxtim-Ti3Config-LLR-001

Requirement ID: H398-LLD-GWY-FNC-5611

The function shall disable the channel 3 i.e reset the CC3E Bit by performing the following:

1. Set compare enable register i.e ccer of tim\_x with Bitwise AND of ccer of tim\_x and negation of M\_TIM\_CCER\_CC3E
2. Set temporary compare mode register 2 with ccmr2 of tim\_x (compare mode register 2)
3. Set temporary compare enable register with ccer of tim\_x (compare enable register)
4. Set select polarity to bit shift of tim\_ic\_polarity to left by M\_GET\_IC\_POLARITY

##### 11.13.12.7.2 daulibstm32f4xxtim-Ti3Config-LLR-002

Requirement ID: H398-LLD-GWY-FNC-5612

The function shall set the filter by performing the following:

1. Set temporary compare mode register 2 with temporary compare mode register 2 Bitwise AND (negation of M\_TIM\_CCMR1\_CC1S) Bitwise AND (negation of M\_TIM\_CCMR2\_IC3F)
2. Set temporary compare mode register 2 with temporary compare mode register 2 Bitwise OR (tim\_ic\_selection Bitwise OR (bit shift of tim\_ic\_filter to left by M\_FOUR ))

##### 11.13.12.7.3 daulibstm32f4xxtim-Ti3Config-LLR-003

Requirement ID: H398-LLD-GWY-FNC-5613

The function shall select the Polarity and set the CC3E Bit by performing the following:

1. Set temporary compare enable register with temporary compare enable register Bitwise AND (negation of (M\_TIM\_CCER\_CC3P Bitwise OR M\_TIM\_CCER\_CC3NP))
2. Set temporary compare enable register with temporary compare enable register Bitwise OR (select polarity Bitwise OR M\_TIM\_CCER\_CC3E)

##### 11.13.12.7.4 daulibstm32f4xxtim-Ti3Config-LLR-004

Requirement ID: H398-LLD-GWY-FNC-5614

The function shall write to tim\_x CCMR2 and CCER registers by performing the following:

1. Set ccmr2 of tim\_x to temporary compare mode register 2
2. Set ccer of tim\_x to temporary compare enable register

### 11.13.13 Ti4Config

Low Level Design Details about CSU Ti4Config will follow in the sub sections.

#### 11.13.13.1 Brief Description

The function Ti4Config configures the TI4 as Input.

#### 11.13.13.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 11.13.13.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 11.13.13.4 Parameter list (Input/Output)

Inputs: T\_TIM\* tim\_x - where x can be 1, 2, 3, 4, 5 or 8 to select the

TIM peripheral

T\_UINT16 tim\_ic\_polarity - The Input Polarity

T\_UINT16 tim\_ic\_selection - Specifies the input to be used

T\_UINT16 tim\_ic\_filter - Specifies the Input Capture Filter

Outputs: T\_TIM\* tim\_x - where x can be 1, 2, 3, 4, 5 or 8 to select the

TIM peripheral

#### 11.13.13.5 Return Value

None

#### 11.13.13.6 Other CSUs called by this CSU

None

#### 11.13.13.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to Ti4Config.

##### 11.13.13.7.1 daulibstm32f4xxtim-Ti4Config-LLR-001

Requirement ID: H398-LLD-GWY-FNC-5623

The function shall disable the channel 4 i.e reset the CC4E Bit by performing the following:

1. Set compare enable register i.e ccer of tim\_x with Bitwise AND of ccer of tim\_x and negation of M\_TIM\_CCER\_CC4E
2. Set temporary compare mode register 2 with ccmr2 of tim\_x (compare mode register 2)
3. Set temporary compare enable register with ccer of tim\_x (compare enable register)
4. Set select polarity to bit shift of tim\_ic\_polarity to left by M\_TWELVE

##### 11.13.13.7.2 daulibstm32f4xxtim-Ti4Config-LLR-002

Requirement ID: H398-LLD-GWY-FNC-5624

The function shall set the filter by performing the following:

1. Set temporary compare mode register 2 with temporary compare mode register 2 Bitwise AND (negation of M\_TIM\_CCMR1\_CC2S) Bitwise AND (negation of M\_TIM\_CCMR1\_IC2F)
2. Set temporary compare mode register 2 with temporary compare mode register 2 Bitwise OR (bit shift of tim\_ic\_selection to left by M\_EIGHT)
3. Set temporary compare mode register 2 with temporary compare mode register 2 Bitwise OR (bit shift of tim\_ic\_filter to left by M\_TWELVE)

##### 11.13.13.7.3 daulibstm32f4xxtim-Ti4Config-LLR-003

Requirement ID: H398-LLD-GWY-FNC-5625

The function shall select the Polarity and set the CC4E Bit by performing the following:

1. Set temporary compare enable register with temporary compare enable register Bitwise AND (negation of (M\_TIM\_CCER\_CC4P Bitwise OR M\_TIM\_CCER\_CC4NP))
2. Set temporary compare enable register with temporary compare enable register Bitwise OR (select polarity Bitwise OR M\_TIM\_CCER\_CC4E)

##### 11.13.13.7.4 daulibstm32f4xxtim-Ti4Config-LLR-004

Requirement ID: H398-LLD-GWY-FNC-5626

The function shall write to tim\_x CCMR2 and CCER registers by performing the following:

1. Set ccmr2 of tim\_x to temporary compare mode register 2
2. Set ccer of tim\_x to temporary compare enable register

### 11.13.14 TimICInit

Low Level Design Details about CSU TimICInit will follow in the sub sections.

#### 11.13.14.1 Brief Description

The function TimICInit initializes the TIMER peripheral according to the specified tim\_ic\_init.

#### 11.13.14.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 11.13.14.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 11.13.14.4 Parameter list (Input/Output)

Inputs: T\_TIM\* tim\_x - where x can be 1 to 14 except 6 and 7 to select the TIM peripheral

T\_TIM\_ICINIT\* tim\_ic\_init - pointer to a T\_TIM\_ICINIT structure that contains the configuration

information for specified TIM peripheral

Outputs: None

#### 11.13.14.5 Return Value

None

#### 11.13.14.6 Other CSUs called by this CSU

None

#### 11.13.14.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TimICInit.

##### 11.13.14.7.1 daulibstm32f4xxtim-TimICInit-LLR-001

Requirement ID: H398-LLD-GWY-FNC-5635

The function shall perform the following when tim\_channel of tim\_ic\_init is M\_TIM\_CHANNEL\_1:

1. Configure TI1 by calling Ti1Config with parameters (tim\_x, tim\_ic\_polarity of tim\_ic\_init, tim\_ic\_selection of tim\_ic\_init, tim\_ic\_filter of tim\_ic\_init)
2. Set the Input Capture Prescaler value by calling TimSetIC1Prescaler with parameters (tim\_x, tim\_ic\_prescaler of tim\_ic\_init)

##### 11.13.14.7.2 daulibstm32f4xxtim-TimICInit-LLR-002

Requirement ID: H398-LLD-GWY-FNC-5636

The function shall perform the following when tim\_channel of tim\_ic\_init is M\_TIM\_CHANNEL\_2:

1. Configure TI2 by calling Ti2Config with parameters (tim\_x, tim\_ic\_polarity of tim\_ic\_init, tim\_ic\_selection of tim\_ic\_init, tim\_ic\_filter of tim\_ic\_init)
2. Set the Input Capture Prescaler value by calling TimSetIC2Prescaler with parameters (tim\_x, tim\_ic\_prescaler of tim\_ic\_init)

##### 11.13.14.7.3 daulibstm32f4xxtim-TimICInit-LLR-003

Requirement ID: H398-LLD-GWY-FNC-5637

The function shall perform the following when tim\_channel of tim\_ic\_init is M\_TIM\_CHANNEL\_3:

1. Configure TI3 by calling Ti3Config with parameters (tim\_x, tim\_ic\_polarity of tim\_ic\_init, tim\_ic\_selection of tim\_ic\_init, tim\_ic\_filter of tim\_ic\_init)
2. Set the Input Capture Prescaler value by calling TimSetIC3Prescaler with parameters (tim\_x, tim\_ic\_prescaler of tim\_ic\_init)

##### 11.13.14.7.4 daulibstm32f4xxtim-TimICInit-LLR-004

Requirement ID: H398-LLD-GWY-FNC-5638

The function shall perform the following when tim\_channel of tim\_ic\_init is other than M\_TIM\_CHANNEL\_1, M\_TIM\_CHANNEL\_2 and M\_TIM\_CHANNEL\_3:

1. Configure TI4 by calling Ti4Config with parameters (tim\_x, tim\_ic\_polarity of tim\_ic\_init, tim\_ic\_selection of tim\_ic\_init, tim\_ic\_filter of tim\_ic\_init)
2. Set the Input Capture Prescaler value by calling TimSetIC4Prescaler with parameters (tim\_x, tim\_ic\_prescaler of tim\_ic\_init)

### 11.13.15 TimClearITPendingBit

Low Level Design Details about CSU TimClearITPendingBit will follow in the sub sections.

#### 11.13.15.1 Brief Description

The function TimClearITPendingBit clears the timer’s interrupt pending bits.

#### 11.13.15.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 11.13.15.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 11.13.15.4 Parameter list (Input/Output)

Inputs: T\_UINT16 tim\_it - specifies the TIM interrupts sources to be enabled or disabled.

Outputs: T\_TIM\* tim\_x - TIM peripheral, where x can be 1 to 14 to select the TIM peripheral.

#### 11.13.15.5 Return Value

None

#### 11.13.15.6 Other CSUs called by this CSU

None

#### 11.13.15.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TimClearITPendingBit.

##### 11.13.15.7.1 daulibstm32f4xxtim-TimClearITPendingBit-LLR-001

Requirement ID: H398-LLD-GWY-FNC-5647

The function shall configure sr (status resister) of timer to clear the received interrupt pending bit i.e, set sr of tim\_x to negated value of tim\_it.

### 11.13.16 TimSelectOutputTrigger

Low Level Design Details about CSU TimSelectOutputTrigger will follow in the sub sections.

#### 11.13.16.1 Brief Description

The function TimSelectOutputTrigger selects the timer Trigger Output Mode.

#### 11.13.16.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 11.13.16.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 11.13.16.4 Parameter list (Input/Output)

Inputs: T\_TIM \* tim\_x - TIM peripheral, where x can be 1 to 14 to select the TIM peripheral.

T\_UINT16 tim\_trgo\_source - specifies the Trigger Output source.

Outputs: T\_TIM \* tim\_x - TIM peripheral, where x can be 1 to 14 to select the TIM peripheral.

#### 11.13.16.5 Return Value

None

#### 11.13.16.6 Other CSUs called by this CSU

None

#### 11.13.16.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to TimSelectOutputTrigger.

##### 11.13.16.7.1 daulibstm32f4xxtim-TimSelectOutputTrigger-LLR-001

Requirement ID: H398-LLD-GWY-FNC-5656

The function shall configure control register 2 of timer to reset the MMS bits (bit 4,5 and 6) and configure with timer trigger source i.e,

1. Set cr2 of tim\_x to (cr2 of tim\_x bitwise AND with negated value of M\_TIM\_CR2\_MMS).
2. Set cr2 of tim\_x to (cr2 of tim\_x bitwise OR with tim\_trgo\_source).

## 11.14 daulibstm32f4xxusart

The daulibstm32f4xxusart CSC provides firmware functions to manage the following functionalities of the Universal synchronous asynchronous receiver transmitter (USART):

* Initialization and Configuration
* data transfers
* Interrupts and flags management

### 11.14.1 UsartDeInit

Low Level Design Details about CSU UsartDeInit will follow in the sub sections.

#### 11.14.1.1 Brief Description

The function UsartDeInit deinitializes the USARTx peripheral registers to their default reset values.

#### 11.14.1.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 11.14.1.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 11.14.1.4 Parameter list (Input/Output)

Inputs: T\_USART \* usart\_x – pointer to USART or UART peripheral, where x can be 1, 2, 3, 4, 5 or 6 to select the USART or UART peripheral.

Outputs: None

#### 11.14.1.5 Return Value

None

#### 11.14.1.6 Other CSUs called by this CSU

RccApb2PeriphResetCmd,

RccApb1PeriphResetCmd

#### 11.14.1.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to UsartDeInit

##### 11.14.1.7.1 daulibstm32f4xxusart-UsartDeInit-LLR-001

Requirement ID: H398-LLD-GWY-FNC-5666

The function shall do the following when USARTx peripheral usart\_x is equal to M\_USART1:

a) Call the function RccApb2PeriphResetCmd with parameters M\_RCC\_APB2PERIPH\_USART1 and ENABLE.

b) Call the function RccApb2PeriphResetCmd with parameters M\_RCC\_APB2PERIPH\_USART1 and DISABLE.

##### 11.14.1.7.2 daulibstm32f4xxusart-UsartDeInit-LLR-002

Requirement ID: H398-LLD-GWY-FNC-5667

The function shall do the following when USARTx peripheral usart\_x is equal to M\_USART2:

a) Call the function RccApb1PeriphResetCmd with parameters M\_RCC\_APB1PERIPH\_USART2 and ENABLE.

b) Call the function RccApb1PeriphResetCmd with parameters M\_RCC\_APB1PERIPH\_USART2 and DISABLE.

##### 11.14.1.7.3 daulibstm32f4xxusart-UsartDeInit-LLR-003

Requirement ID: H398-LLD-GWY-FNC-5668

The function shall do the following when USARTx peripheral usart\_x is equal to M\_USART3:

a) Call the function RccApb1PeriphResetCmd with parameters M\_RCC\_APB1PERIPH\_USART3 and ENABLE.

b) Call the function RccApb1PeriphResetCmd with parameters M\_RCC\_APB1PERIPH\_USART3 and DISABLE.

##### 11.14.1.7.4 daulibstm32f4xxusart-UsartDeInit-LLR-004

Requirement ID: H398-LLD-GWY-FNC-5669

The function shall do the following when USARTx peripheral usart\_x is equal to M\_UART4:

a) Call the function RccApb1PeriphResetCmd with parameters M\_RCC\_APB1PERIPH\_UART4 and ENABLE.

b) Call the function RccApb1PeriphResetCmd with parameters M\_RCC\_APB1PERIPH\_UART4 and DISABLE.

##### 11.14.1.7.5 daulibstm32f4xxusart-UsartDeInit-LLR-005

Requirement ID: H398-LLD-GWY-FNC-5670

The function shall do the following when USARTx peripheral usart\_x is equal to M\_UART5:

a) Call the function RccApb1PeriphResetCmd with parameters M\_RCC\_APB1PERIPH\_UART5 and ENABLE.

b) Call the function RccApb1PeriphResetCmd with parameters M\_RCC\_APB1PERIPH\_UART5 and DISABLE.

##### 11.14.1.7.6 daulibstm32f4xxusart-UsartDeInit-LLR-006

Requirement ID: H398-LLD-GWY-FNC-5671

The function shall do the following when USARTx peripheral usart\_x is other than M\_USART1, M\_USART2, M\_USART3, M\_USART4 and M\_USART5. Perform the following

when USARTx peripheral usart\_x is equal to M\_USART6,

a) Call the function RccApb2PeriphResetCmd with parameters M\_RCC\_APB2PERIPH\_USART6 and ENABLE.

b) Call the function RccApb2PeriphResetCmd with parameters M\_RCC\_APB2PERIPH\_USART6 and DISABLE.

else does nothing.

### 11.14.2 UsartInit

Low Level Design Details about CSU UsartInit will follow in the sub sections.

#### 11.14.2.1 Brief Description

The function UsartInit initializes the USARTx peripheral according to the specified parameters in the usart\_init\_struct.

#### 11.14.2.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 11.14.2.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 11.14.2.4 Parameter list (Input/Output)

Inputs:T\_USART\* usart\_x – pointer to USART or UART peripheral, where x can be 1, 2, 3, 4, 5 or 6 to select the USART or UART peripheral.

T\_USART\_INIT\* usart\_init\_struct - pointer to a T\_USART\_INIT structure that contains the configuration information for the specified USART peripheral

Outputs: T\_USART\* usart\_x – pointer to USART or UART peripheral, where x can be 1, 2, 3, 4, 5 or 6 to select the USART or UART peripheral.

T\_USART\_INIT\* usart\_init\_struct - pointer to a T\_USART\_INIT structure that contains the configuration information for the specified USART peripheral

#### 11.14.2.5 Return Value

None

#### 11.14.2.6 Other CSUs called by this CSU

RCCGetClocksFreq

#### 11.14.2.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to UsartInit

##### 11.14.2.7.1 daulibstm32f4xxusart-UsartInit-LLR-001

Requirement ID: H398-LLD-GWY-FNC-5680

The function shall perform the following

* Configures the USART cr2(cr2 of usart\_x),
* Clears the STOP bits[13:12](register AND with not of M\_USART\_CR2\_STOP).
* Sets the stop [13:12] bits according to usart\_stopbits values.(register OR with usart\_stopbits of usart\_init\_struct).
* Write to USART cr2(register to cr2 of usart\_x)
* Sets the USART Cr1 configuration(register to cr1 of usart\_x)

##### 11.14.2.7.2 daulibstm32f4xxusart-UsartInit-LLR-002

Requirement ID: H398-LLD-GWY-FNC-5681

The function shall clear the M, PCE, PS, TE and RE bits of USART Control register 1(cr1) (register AND with not of M\_CR1\_CLEAR\_MASK) and configure with usart\_wordlength, usart\_parity, usart\_mode of usart\_init\_struct(register OR with usart\_wordlength, usart\_parity, usart\_mode of usart\_init\_struct).

Write to USART cr1(cr1 of usart\_x to register) and configuration of USART cr3(register to cr3 of usart\_x).

##### 11.14.2.7.3 daulibstm32f4xxusart-UsartInit-LLR-003

Requirement ID: H398-LLD-GWY-FNC-5682

The function shall clear the CTSE and RTSE bits of USART Control register 3(cr3)(register AND with not of M\_CR3\_CLEAR\_MASK) and configure with usart\_hardware\_flowcontrol of usart\_init\_struct(register OR with usart\_hardware\_flowcontrol of usart\_init\_struct) and write to USART cr3(cr3 of usart\_x to register).

##### 11.14.2.7.4 daulibstm32f4xxusart-UsartInit-LLR-004

Requirement ID: H398-LLD-GWY-FNC-5683

The function shall call the function RCCGetClocksFreq with parameter address of rcc clock status.

##### 11.14.2.7.5 daulibstm32f4xxusart-UsartInit-LLR-005

Requirement ID: H398-LLD-GWY-FNC-5684

The function shall set the apb clock to pclk\_2\_frequency of rcc clock status when usart\_x is equal to M\_USART1 OR M\_USART6.

##### 11.14.2.7.6 daulibstm32f4xxusart-UsartInit-LLR-006

Requirement ID: H398-LLD-GWY-FNC-5685

The function shall set the apb clock to pclk\_1\_frequency of rcc clock status when usart\_x is not equal to M\_USART1 OR M\_USART6 .

##### 11.14.2.7.7 daulibstm32f4xxusart-UsartInit-LLR-007

Requirement ID: H398-LLD-GWY-FNC-5686

The function shall configure USART Baud rate register with integer\_divider for Oversampling mode is 8 accordingly(set the integer divider to (M\_TWENTY\_FIVE multiply apb\_clock) / (M\_TWO multiply usart\_baudrate of usart\_init\_struct)) when M\_ZERO is not (cr1 of usart\_x AND M\_USART\_CR1\_OVER8).

##### 11.14.2.7.8 daulibstm32f4xxusart-UsartInit-LLR-008

Requirement ID: H398-LLD-GWY-FNC-5687

The function shall configure USART Baud rate register with integer\_divider for Oversampling mode is 16 accordingly(set the integer divider to (M\_TWENTY\_FIVE multiply apb\_clock) / (M\_FOUR multiply usart\_baudrate of usart\_init\_struct)) when M\_ZERO is equal to (cr1 of usart\_x AND M\_USART\_CR1\_OVER8).

##### 11.14.2.7.9 daulibstm32f4xxusart-UsartInit-LLR-009

Requirement ID: H398-LLD-GWY-FNC-5688

The function shall set the register to (integer divider divided by M\_HUNDRED) left shift of M\_FOUR and fractional part as fractional divider to integer divider subtracted (M\_HUNDRED multiply (register right shift of M\_FOUR)).

##### 11.14.2.7.10 daulibstm32f4xxusart-UsartInit-LLR-010

Requirement ID: H398-LLD-GWY-FNC-5689

The function shall implements the fractional part in the register. i.e., set the register to ((fractional divider multiply M\_EIGHT) added with M\_FIFTY and divided by M\_HUNDRED AND with M\_HEX\_SEVEN). Otherwise does the following as set the register to ((fractional divider multiply M\_SIXTEEN) added with M\_FIFTY and divided by M\_HUNDRED AND with M\_HEX\_FIFTEEN).

##### 11.14.2.7.11 daulibstm32f4xxusart-UsartInit-LLR-011

Requirement ID: H398-LLD-GWY-FNC-5690

The function shall write the USART BRR register i.e., sets the brr of usart\_x to register.

### 11.14.3 UsartCmd

Low Level Design Details about CSU UsartCmd will follow in the sub sections.

#### 11.14.3.1 Brief Description

The function UsartCmd enables or disables the specified USART peripheral.

#### 11.14.3.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 11.14.3.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 11.14.3.4 Parameter list (Input/Output)

Inputs: T\_USART\* usart\_x – pointer to USART or UART peripheral, where x can be 1, 2, 3, 4, 5 or 6 to select the USART or UART peripheral.

T\_FUNCTIONAL\_STATE new\_state - New state of the USARTx peripheral.

Outputs: T\_USART\* usart\_x – pointer to USART or UART peripheral, where x can be 1, 2, 3, 4, 5 or 6 to select the USART or UART peripheral.

#### 11.14.3.5 Return Value

None

#### 11.14.3.6 Other CSUs called by this CSU

None

#### 11.14.3.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to UsartCmd

##### 11.14.3.7.1 daulibstm32f4xxusart-UsartCmd-LLR-001

Requirement ID: H398-LLD-GWY-FNC-5699

The function shall enable the selected USART by setting the USART Enable (M\_USART\_CR1\_UE) bit in the USART cr1 register of usart\_x when new\_state is not of DISABLE.

##### 11.14.3.7.2 daulibstm32f4xxusart-UsartCmd-LLR-002

Requirement ID: H398-LLD-GWY-FNC-5700

The function shall disable the selected USART by clearing the USART (M\_USART\_CR1\_UE) bit in the USART cr1 register of usart\_x when new\_state is DISABLE.

### 11.14.4 UsartSendData

Low Level Design Details about CSU UsartSendData will follow in the sub sections.

#### 11.14.4.1 Brief Description

The function UsartSendData transmits single data through the usart\_x peripheral.

#### 11.14.4.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 11.14.4.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 11.14.4.4 Parameter list (Input/Output)

Inputs: T\_UINT16 data - Data to transmit.

Outputs: T\_USART\* usart\_x – pointer to USART or UART peripheral, where x can be 1, 2, 3, 4, 5 or 6 to select the USART or UART peripheral.

#### 11.14.4.5 Return Value

None

#### 11.14.4.6 Other CSUs called by this CSU

None

#### 11.14.4.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to UsartSendData

##### 11.14.4.7.1 daulibstm32f4xxusart-UsartSendData-LLR-001

Requirement ID: H398-LLD-GWY-FNC-5709

The function shall configure USART data register with data to be transmitted i.e., sets the dr of usart\_x to data Bitwise AND M\_USART\_DR.

### 11.14.5 UsartReceiveData

Low Level Design Details about CSU UsartReceiveData will follow in the sub sections.

#### 11.14.5.1 Brief Description

The function UsartReceiveData returns the most recent received data by the USARTx peripheral.

#### 11.14.5.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 11.14.5.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 11.14.5.4 Parameter list (Input/Output)

Inputs: T\_USART\* usart\_x – pointer to USART or UART peripheral, where x can be 1, 2, 3, 4, 5 or 6 to select the USART or UART peripheral.

Outputs: None

#### 11.14.5.5 Return Value

T\_UINT16 - returns the received data.

#### 11.14.5.6 Other CSUs called by this CSU

None

#### 11.14.5.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to UsartReceiveData

##### 11.14.5.7.1 daulibstm32f4xxusart-UsartReceiveData-LLR-001

Requirement ID: H398-LLD-GWY-FNC-5718

The function shall return the data received by usart peripheral i.e., return the dr of usart\_x Bitwise AND M\_USART\_DR.

### 11.14.6 UsartItConfig

Low Level Design Details about CSU UsartItConfig will follow in the sub sections.

#### 11.14.6.1 Brief Description

The function UsartItConfig enables or disables the specified USART interrupts.

#### 11.14.6.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 11.14.6.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 11.14.6.4 Parameter list (Input/Output)

Inputs: T\_USART\* usart\_x – pointer to USART or UART peripheral, where x can be 1, 2, 3, 4, 5 or 6 to select the USART or UART peripheral.

T\_UINT16 usart\_it - specifies the USART interrupt sources to be enabled or disabled.

T\_FUNCTIONAL\_STATE new\_state - new state of the specified USARTx interrupts.

Outputs: None

#### 11.14.6.5 Return Value

None

#### 11.14.6.6 Other CSUs called by this CSU

None

#### 11.14.6.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to UsartItConfig

##### 11.14.6.7.1 daulibstm32f4xxusart-UsartItConfig-LLR-001

Requirement ID: H398-LLD-GWY-FNC-5727

The function shall get the usart register index i.e., register to usart\_it right shift of M\_FIVE

And gets the interrupt position as sets the interrupt position to usart\_it AND M\_IT\_MASK, interrupt mask to M\_ONE left shift of interrupt position.

##### 11.14.6.7.2 daulibstm32f4xxusart-UsartItConfig-LLR-002

Requirement ID: H398-LLD-GWY-FNC-5728

The function shall set the usart base address as sum of usartx\_base and M\_USART\_CR1\_OFFSET when USART register index is equal to M\_ONE.

##### 11.14.6.7.3 daulibstm32f4xxusart-UsartItConfig-LLR-003

Requirement ID: H398-LLD-GWY-FNC-5729

The function shall set the usart base address as sum of usartx\_base and M\_USART\_CR2\_OFFSET when USART register index is M\_TWO.

##### 11.14.6.7.4 daulibstm32f4xxusart-UsartItConfig-LLR-004

Requirement ID: H398-LLD-GWY-FNC-5730

The function shall set the usart base address as sum of usartx\_base and M\_USART\_CR3\_OFFSET when USART register index is other than M\_ONE and M\_TWO.

##### 11.14.6.7.5 daulibstm32f4xxusart-UsartItConfig-LLR-005

Requirement ID: H398-LLD-GWY-FNC-5731

The function shall set the particular interrupt bit in usart control register (usart base address) when new\_state is not equal to DISABLE. ie., (usartx\_base OR with interrupt mask).

Note: bit to set the interrupt mask is calculated as below

- Interrupt position = usart\_it BITWISE AND with M\_IT\_MASK.

- Interrupt mask = interrupt position left shift by M\_ONE

##### 11.14.6.7.6 daulibstm32f4xxusart-UsartItConfig-LLR-006

Requirement ID: H398-LLD-GWY-FNC-5732

The function shall set the particular interrupt bit in usart control register (usart base address) when new\_state is equal to DISABLE. ie., (usartx\_base AND with not of interrupt mask).

Note: bit to reset the interrupt mask is calculated as below

- interrupt position = usart\_it BITWISE AND with M\_IT\_MASK.

- interrupt mask = interrupt position left shift by M\_ONE

### 11.14.7 UsartGetItStatus

Low Level Design Details about CSU UsartGetItStatus will follow in the sub sections.

#### 11.14.7.1 Brief Description

The function UsartGetItStatus checks whether the specified USART interrupt has occurred or not.

#### 11.14.7.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

#### 11.14.7.3 List of global variables accessed and modified

Accessed: None

Modified: None

#### 11.14.7.4 Parameter list (Input/Output)

Inputs: T\_USART\* usart\_x – pointer to USART or UART peripheral, where x can be 1, 2, 3, 4, 5 or 6 to select the USART or UART peripheral.

T\_UINT16 usart\_it - Specifies the USART interrupt source to check.

Outputs: None

#### 11.14.7.5 Return Value

T\_ITSTATUS: Returns the new state of usart\_it (SET or RESET).

#### 11.14.7.6 Other CSUs called by this CSU

None

#### 11.14.7.7 Description of list of LLRs allocated

The following section will list the LLRs allocated to UsartGetItStatus

##### 11.14.7.7.1 daulibstm32f4xxusart-UsartGetItStatus-LLR-001

Requirement ID: H398-LLD-GWY-FNC-5741

The function shall fetch the USART register index by extracting bit 5 to bit 7 of usart interrupt source usart\_it. i.e.,( usart\_it right shift of M\_FIVE ) and get the interrupt position as interrupt mask to M\_ONE left shifted by (usart\_it Bitwise AND M\_IT\_MASK)

##### 11.14.7.7.2 daulibstm32f4xxusart-UsartGetItStatus-LLR-002

Requirement ID: H398-LLD-GWY-FNC-5742

The function shall set the interrupt mask to value at particular interrupt bit position in usart control register 1(cr1 of usart\_x) when USART register index is equal to M\_ONE.

##### 11.14.7.7.3 daulibstm32f4xxusart-UsartGetItStatus-LLR-003

Requirement ID: H398-LLD-GWY-FNC-5743

The function shall set the interrupt mask to value at particular interrupt bit position in usart control register 2(cr2 of usart\_x) when USART register index is M\_TWO.

##### 11.14.7.7.4 daulibstm32f4xxusart-UsartGetItStatus-LLR-004

Requirement ID: H398-LLD-GWY-FNC-5744

The function shall set the interrupt mask to value at particular interrupt bit position in usart control register 3(cr3 of usart\_x) when USART register index is other than M\_ONE and M\_TWO.

##### 11.14.7.7.5 daulibstm32f4xxusart-UsartGetItStatus-LLR-005

Requirement ID: H398-LLD-GWY-FNC-5745

The function shall return the bit status as SET when interrupt mask is RESET OR bit value is RESET.

Bit value is calculated as below:

- bit position is M\_ONE left shifted by (usart\_it right shift by M\_SHIFT\_EIGHT)

- bit value equal to value of bit in usart Status register at bit position

Returns the bit status.

##### 11.14.7.7.6 daulibstm32f4xxusart-UsartGetItStatus-LLR-006

Requirement ID: H398-LLD-GWY-FNC-5746

The function shall return the bit status as RESET when interrupt mask is RESET AND bit value is RESET.

Bit value is calculated as below:

- bit position is M\_ONE left shifted by (usart\_it right shift by M\_SHIFT\_EIGHT)

- bit value equal to value of bit in usart Status register at bit position

Returns the bit status.

# 12 Software Low Level Requirements - MCD Software

The Software Low Level Requirements-MCD Software section specifies the Software Low Level Requirements for Gateway Module MCD.

Refer section 14 Appendix A: Data Structures of H398-003-001-GWY for Data Structures and Enumerations.

## 12.1 daugwyappdata.c

Implementation of all values of appdata configuration structure.

### 12.1.1 Brief Description

The Implementation of Gateway module configuration data.

### 12.1.2 List of HLRs allocated

The list of HLRs allocated to this CSU is available in the Bi-Directional Traceability Matrix from SLL to SRS/SAD (H398-003-012-GWY).

### 12.1.3 Description of list of LLRs allocated

The following section will list the LLRs allocated to daugwyappdata.

#### 12.1.3.1 daugwyappdata-LLR-001

Requirement ID: H398-LLD-GWY-FNC-5752

The CSC daugwyappdata shall set the Gateway Configuration Software Part Number to "H108E-672 -1.04".

#### 12.1.3.2 daugwyappdata-LLR-004

Requirement ID: H398-LLD-GWY-FNC-5755

The CSC daugwyappdata shall set the number of RS232 Configuration as follows:

Table : RS232 Configuration parameters

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Type | Baud rate | start | Data bits | parity | stop | State |
| COM1 | 57600 | 1 | 8 | 0 | 1 | DISABLED |
| COM2 | 38400 | 1 | 9 | 1 | 1 | DISABLED |
| COM3 | 57600 | 1 | 8 | 0 | 1 | DISABLED |
| COM4 | 57600 | 1 | 8 | 0 | 1 | ENABLED |

#### 12.1.3.3 daugwyappdata-LLR-011

Requirement ID: H398-LLD-GWY-FNC-5762

The CSC daugwyappdata shall set the Sensor lookup table configuration as follows:

Table : Analog Sensor lookup table configuration

|  |  |
| --- | --- |
| Table X | Table Y |
| 0 | 0 |
| t4\_input | t4\_output |
| Eop\_mv | Eop\_bar |
| Eot\_ohms | Eot\_temp |
| Trq\_input | Trq\_output |
| Fp\_signal | Fp\_bar |
| Ng\_hz | Ng\_speed |
| Mgbt\_signal | Mgbt\_psig |
| Lrhp\_signal | Lrhp\_bar |
| Mgbop\_signal | Mgbop\_bar |
| Tot\_ohm | Tot\_rsignal |

#### 12.1.3.4 daugwyappdata-LLR-012

Requirement ID: H398-LLD-GWY-FNC-5763

The CSC daugwyappdata shall set the ARINC 429 channel configuration as follows:

Table : ARINC 429 channel configuration parameters

|  |  |  |
| --- | --- | --- |
| Channel | Parameter | Configured Value |
| Arinc429\_1 RX\_1 | sys\_connected | CONNECTED |
| func\_state | ENABLED |
| Arinc429\_1 RX\_2 | sys\_connected | CONNECTED |
| func\_state | ENABLED |
| Arinc429\_2 RX\_1 | sys\_connected | CONNECTED |
| func\_state | ENABLED |
| Arinc429\_2 RX\_2 | sys\_connected | CONNECTED |
| func\_state | ENABLED |
| Arinc429\_3 RX\_1 | sys\_connected | CONNECTED |
| func\_state | ENABLED |
| Arinc429\_3 RX\_2 | sys\_connected | CONNECTED |
| func\_state | ENABLED |
| Arinc429\_4 RX\_1 | sys\_connected | NO\_CONNECTION |
| func\_state | DISABLED |
| Arinc429\_4 RX\_2 | sys\_connected | CONNECTED |
| func\_state | DISABLED |
| Arinc429\_5 RX\_1 | sys\_connected | NO\_CONNECTION |
| func\_state | DISABLED |
| Arinc429\_5 RX\_2 | sys\_connected | CONNECTED |
| func\_state | DISABLED |
| Arinc429\_6 RX\_1 | sys\_connected | NO\_CONNECTION |
| func\_state | DISABLED |
| Arinc429\_6 RX\_2 | sys\_connected | CONNECTED |
| func\_state | DISABLED |
| Arinc429\_7 RX\_1 | sys\_connected | NO\_CONNECTION |
| func\_state | DISABLED |
| Arinc429\_7 RX\_2 | sys\_connected | CONNECTED |
| func\_state | DISABLED |
| Arinc429\_8 RX\_1 | sys\_connected | NO\_CONNECTION |
| func\_state | DISABLED |
| Arinc429\_8 RX\_2 | sys\_connected | CONNECTED |
| func\_state | DISABLED |

#### 12.1.3.5 daugwyappdata-LLR-014

Requirement ID: H398-LLD-GWY-FNC-5765

The CSC daugwyappdata shall set the Number of Audio Data to 0.

#### 12.1.3.6 daugwyappdata-LLR-015

Requirement ID: H398-LLD-GWY-FNC-5766

The CSC daugwyappdata shall set the Audio Tones configuration as follows:

Table : Audio tones output Configuration format

|  |  |
| --- | --- |
| Size | Data |
| 0 | 0 |
| 4016 | masterCautionWarning |
| 3997 | autoPiloDecouple |
| 13973 | oneHrdFeet |
| 13944 | oneFiftyFeet |
| 10859 | checkAltitude |
| 9087 | checkHeight |
| 19929 | lowRotorRPM |
| 17680 | vneWarning |
| 13837 | Airbus412 |

#### 12.1.3.7 daugwyappdata-LLR-019

Requirement ID: H398-LLD-GWY-FNC-5770

The CSC daugwyappdata shall set the Number of Tach Data to 5.

#### 12.1.3.8 daugwyappdata-LLR-020

Requirement ID: H398-LLD-GWY-FNC-5771

The CSC daugwyappdata shall set the Number of Tach Data configuration as follows:

Table : Analog Parameter configuration format for TACH Channels

|  |  |  |
| --- | --- | --- |
| Channel | Parameter | Configured Value |
| Tach 1 | u8ChanState | ENABLED |
| u8DefaultState | ENABLED |
| u16DefaultVal | 0 |
| f32RefFreq | 70.0f |
| u16Scaled | 1000 |
| u16MinRange | 0 |
| u16MaxRange | 1250 |
| i16Resolution | 10 |
| Tach 2 | u8ChanState | ENABLED |
| u8DefaultState | ENABLED |
| u16DefaultVal | 0 |
| f32RefFreq | 70.0f |
| u16Scaled | 1000 |
| u16MinRange | 0 |
| u16MaxRange | 1250 |
| i16Resolution | 10 |
| Tach 3 | u8ChanState | ENABLED |
| u8DefaultState | ENABLED |
| u16DefaultVal | 0 |
| f32RefFreq | 70. 383f |
| u16Scaled | 1000 |
| u16MinRange | 0 |
| u16MaxRange | 1250 |
| i16Resolution | 10 |
| Tach 4 | u8ChanState | ENABLED |
| u8DefaultState | ENABLED |
| u16DefaultVal | 0 |
| f32RefFreq | 70. 383f |
| u16Scaled | 1000 |
| u16MinRange | 0 |
| u16MaxRange | 1250 |
| i16Resolution | 10 |
| Tach 5 | u8ChanState | ENABLED |
| u8DefaultState | ENABLED |
| u16DefaultVal | 0 |
| f32RefFreq | 71.76f |
| u16Scaled | 1000 |
| u16MinRange | 0 |
| u16MaxRange | 1250 |
| i16Resolution | 10 |

# 13 Appendix A : Data Dictionary

The Data Dictionary for the Gateway Module :



The constant data table for the Gateway Module MCD Software.



# 14 Appendix B: Data Constants

The Data Constants for the Gateway Module Application Software :

****

The Data Constants for the Module Configuration Data(MCD) Software of the Gateway Module :