MDA-QS-001-MAP-Rev A

29 October 2006

Missile Defense Agency Assurance Provisions (MAP)

Approved by: 11/30/06

RANDOLPH R. STONE **Date**
Director
Quality, Safety, and Mission Assurance

DEPARTMENT OF DEFENSE
MISSILE DEFENSE AGENCY
7100 DEFENSE PENTAGON
WASHINGTON, DC 20301-7100

NOV 2 8 2006

MEMORANDUM FOR: PROGRAM CHANGE BOARD

SUBJECT: MDA Assurance Provisions

The MDA Assurance Provisions (MAP) applies to the acquisition of mission and safety critical hardware, software or firmware for the Missile Defense Agency (MDA). Revision A of the MAP incorporates changes based upon lessons learned, DoD IG recommendations, and MDA's reengineering. MAP establishes a path to provide significant and measurable improvement in MDA's acquisition activities through the effective application of critical best practices for quality safety, and mission assurance.

Based upon the MDA Program Change Board (PCB) review of MDA Assurance Provisions (MAP) Rev A, I approve the subject document, MDA-QS-0001-Rev A, dated 29 October 2006, as the MDA requirements for acquiring safety and mission critical products.

HENRY A. OBERING III
Lieutenant General, USAF
Director

DISTRIBUTION:
D (LtGen Obering)
DD (BG McNamara)
DX (Dr. Sanders)
DE (Mr. Englander)
DO (Mr. Altwegg)
DT (Maj Gen Anzalone)
DA (Ms Wahl)
DV (Mr. Barnes)
DI (Ms Vinson)
AB (RADM Hicks)
GM (BG O'Reilly)
KI (Mr. Brewer)
MK (Mr. Matlock)
TH (COL Driessnack)
AW (Mr. Eddleman)
AL (Col Daniels)
SN (Ms. Finney)
BC (Brig Gen Dehnert)
SS (Col Pelc)
JFCC_IMD (CAPT Yancey)
QS (Mr. Stone)
IC (BG McNamara)

Revision Record

Revision	Date	Change	Affected Pages
Original	9 January 2004	Reflects CCB approval with comment.	
Rev A	29 October 2006	Address IG Audit Findings and MDA reengineering	All

This Page Intentionally Left Blank

– **Table of Contents** –

1.0 SCOPE 1

1.1 PURPOSE 1
1.2 APPLICABILITY AND ACCOUNTABILITY 1
1.2.1 Mission Assurance Implementation Plan 1
1.2.2 MAP on Contract: 1
1.3 APPLICABLE DOCUMENTS 2
1.4 ORDER OF PRECEDENCE 5
1.5 CONFLICTING REQUIREMENTS 5

2.0 DEFINITIONS 6

3.0 QUALITY, SAFETY, AND MISSION ASSURANCE PROVISIONS 12

3.1 MANAGEMENT 12
3.1.1 Contract Reviews 12
3.1.2 Management Reviews 12
3.1.3 Metrics 12
3.1.3.1 MDA Core Metrics 13
3.1.3.1.1 Failure Review Board Actions 13
3.1.3.1.2 Waivers and Deviations 13
3.1.3.1.3 Class I and Class II Change Requests 13
3.1.3.1.4 Design Escapes 14
3.1.3.1.5 Foreign Object Elimination 14
3.1.3.1.6 Cost of Quality 14
3.1.3.1.7 Program Quality Staffing 14
3.1.3.1.8 Software Metrics 14
3.1.4 Management Information Systems 15
3.1.5 Risk Management Program 15
3.1.5.1 Risk Management Plan 16
3.1.6 Pedigree Program 16
3.1.7 Internal Evaluation Program 16
3.1.8 Training and Certification Program 17
3.1.8.1 Training 17
3.1.8.2 Certification 17
3.1.9 Problem and Failure Reporting and Corrective Action System 17
3.1.9.1 Corrective Action 17
3.1.10 Data Exchange Programs Participation 18
3.1.11 Government Insight 18
3.1.12 Program Reviews 18
3.1.13 MDA Evaluations 18
3.1.14 MDA Mission Assurance Representatives 19
3.1.15 Government Furnished Material, Equipment, or Information 19
3.1.16 Repair, Refurbishment, and Modification 19
3.2 DESIGN AND DEVELOPMENT 22
3.2.1 Integrated Product and Process Development 22
3.2.2 Peer Reviews 22
3.2.3 Technical Performance Measurement 22
3.2.4 Systems Engineering for Design 23
3.2.4.1 Systems Engineering Management Plan 23
3.2.5 Design for Interoperability 23
3.2.6 Design for Producibility 23
3.2.7 Design for Testability 23
3.2.8 Design for Supportability 24
3.2.9 Design for Commercial and Non-Developmental Items 24
3.2.9.1 COTS/NDI Design Strategies 24

– **Table of Contents** –

3.2.10 Requirements Traceability and Verification Matrix *25*
3.2.11 System Design Verification and Validation *25*
3.2.12 Safety and Environmental Requirements *26*
3.2.13 Open Systems Design and Standards *26*
3.2.14 Modeling and Simulation *26*
3.2.15 Classification of Characteristics *26*
3.2.15.1 Classification of Characteristics Levels 26
3.2.16 Electromagnetic Environmental Effects Design & Verification *27*
3.2.17 Space Radiation, Nuclear Hardness and Survivability Program *27*
3.2.18 Transition to Operations or Production *28*
3.2.18.1 Transition to Operations or Production Plan 28
3.3 SOFTWARE AND FIRMWARE 30
3.3.1 Management Processes *30*
3.3.1.1 Intergroup Coordination 30
3.3.1.2 Software Development Planning 30
3.3.1.3 Estimation 32
3.3.1.4 Software Risk Management 32
3.3.1.5 Process Improvement 32
3.3.1.6 Technology Change Management 33
3.3.1.7 Software Supplier Management 33
3.3.1.7.1 Selection of Software Suppliers 33
3.3.1.7.2 Flowdown of Requirements 34
3.3.1.7.3 Software Supplier Monitoring 34
3.3.1.7.4 Acceptance of Supplier Software Products 34
3.3.1.8 Software Training 34
3.3.1.8.1 Software Personnel Training 35
3.3.2 Software Development, Maintenance, and Operational Processes *35*
3.3.2.1 Requirements 35
3.3.2.1.1 Software Reuse 36
3.3.2.2 Software Design 36
3.3.2.3 Software Code 37
3.3.2.4 Software Unit Testing 38
3.3.2.5 Software Integration 38
3.3.2.6 Software Qualification 39
3.3.2.6.1 Software Qualification Test Report 40
3.3.2.7 Regression Tests 40
3.3.2.8 Software Test Program Status Reports 40
3.3.2.9 System Integration 40
3.3.2.10 System Qualification 41
3.3.2.11 Software Installation 41
3.3.2.12 Software Acceptance 41
3.3.2.13 Operation 42
3.3.2.14 Software Maintenance 42
3.3.2.15 Software Retirement 42
3.3.3 Supporting Activities and Processes *43*
3.3.3.1 Software Quality Assurance 43
3.3.3.2 Software Verification 43
3.3.3.3 Software Validation 44
3.3.3.4 Support of Independent Verification and Validation 45
3.3.3.5 Independent Verification and Validation 45
3.3.3.6 Software Reviews 45
3.3.3.7 Software Audits 46
3.3.3.8 Software Problem Reporting 46
3.3.3.9 Software Dependability 47
3.3.3.10 Software Assurance Metrics 47

3.3.3.11 Software Safety 47
3.3.3.12 Software Configuration Management 47
3.3.3.12.1 Software Configuration Items 47
3.3.3.12.2 Software Change Control Process 47
3.3.3.12.3 Software Library 48
3.3.3.12.4 Software Configuration Audits 48
3.3.3.12.5 Software Status Accounting 48
3.3.3.12.6 Software and Firmware Media Generation 49
3.3.3.13 Software Documentation 49
3.3.3.13.1 Software Operation and Maintenance Documentation 49
3.4 TECHNICAL AND MISSION ASSURANCE REVIEWS 50
3.4.1 Technical Reviews 50
3.4.1.1 Systems Requirements Review 50
3.4.1.2 System Functional Review 51
3.4.1.3 Software Specification Review 52
3.4.1.4 Preliminary Design Assessments/Critical Design Assessments 52
3.4.1.5 Preliminary Design Review 52
3.4.1.6 Critical Design Review 53
3.4.1.7 Test Readiness Review 54
3.4.1.7.1 MDA Executive Level Flight Test Reviews 54
3.4.1.8 System Verification Review 54
3.4.2 Mission Assurance Reviews 55
3.4.2.1 Mission Readiness Review 55
3.4.2.2 Pre-Environmental Review 55
3.4.2.3 Pre-Shipment Review 55
3.4.2.4 Flight Operations Review 55
3.4.2.5 Pre-Flight Readiness Reviews 56
3.4.2.6 Launch Readiness Review 56
3.4.2.7 Mission Operations Review 56
3.5 RELIABILITY, MAINTAINABILITY, AND AVAILABILITY 58
3.5.1 Reliability, Maintainability, and Availability Program Plan 58
3.5.1.1 Reliability, Maintainability, and Availability Program Planning 58
3.5.2 Supplier Reliability, Maintainability, and Availability Requirements 58
3.5.3 Failure Reporting Analysis and Corrective Action System 58
3.5.4 Failure Review Board 58
3.5.5 Reliability Modeling, Allocation, and Prediction 59
3.5.5.1 Reliability Prediction Methodology 59
3.5.6 Reliability Analyses 59
3.5.6.1 Failure Modes, Effects, and Criticality Analysis 59
3.5.6.2 Fault Tree Analysis 60
3.5.6.3 Finite Element Analysis 61
3.5.6.4 Sneak Circuit Analysis 61
3.5.6.5 Worst Case Analysis 61
3.5.6.6 Electrical, Mechanical, and Thermal Stress Analyses 61
3.5.6.6.1 Thermal Stress Analysis 61
3.5.6.6.2 Mechanical Stress Analysis 62
3.5.6.6.3 Electrical/Electronic Stress Analysis 62
3.5.7 Mission Critical Items 62
3.5.8 Effects of Functional Testing, Storage, Handling, Packaging, Transportation, and Maintenance 62
3.5.9 Controlled and Limited Life Items 63
3.5.10 Reliability Growth Testing Program 63
3.5.11 Accelerated Life Testing 63
3.5.12 Process Failures Modes and Effects Analysis 63
3.5.13 Environmental Stress Screening 64

3.5.14 Reliability Qualification Test Program/Demonstration *64*
3.5.15 Maintainability Allocations and Predictions *65*
3.5.16 Maintainability Analysis *65*
3.5.17 Maintainability Demonstration *65*
3.5.18 Availability Allocations and Predictions *66*
3.5.19 Availability Assessment *66*
3.5.20 Reliability, Maintainability, and Availability Metrics *66*
3.5.21 Reliability, Maintainability, and Availability of Government Furnished Equipment/ Information *66*
3.6 PARTS AND MATERIALS CONTROL PROGRAM 67
3.6.1 Parts, Materials, and Processes Control Program Plan *67*
3.6.2 Parts and Materials Control Board *67*
3.6.3 Parts and Materials Selection List *67*
3.6.4 Diminishing Manufacturing Sources and Material Shortages *68*
3.6.5 Parts and Materials Process Monitoring Program *68*
3.6.6 Alternate and Substitute Part and Material Selection *68*
3.6.7 COTS Parts and Materials Management *68*
3.6.8 Plastic Encapsulated Microcircuit Qualification *68*
3.6.9 Environmental (Lead Free) *69*
3.6.10 Parts and Materials Qualification *69*
3.6.11 Reuse of Parts and Materials *70*
3.6.12 Electrical, Electronic, and Electromechanical Parts Program *70*
3.6.12.1 Electrical, Electronic, and Electromechanical Parts Selection 70
3.6.12.2 Electrical, Electronic, and Electromechanical Parts Derating 70
3.6.12.3 Electrical, Electronic, and Electromechanical Parts Screening 71
3.6.12.4 Electrical, Electronic, and Electromechanical Parts Qualification 71
3.6.12.5 Destructive Physical Analysis 71
3.6.12.6 Parts Age Control 72
3.6.12.7 Radiation Hardness and Survivability Assurance 72
3.6.12.8 Traceability and Lot Control 72
3.6.12.9 Custom Devices 72
3.6.13 Failure Analysis *72*
3.6.14 Material and Finishes Selection *73*
3.6.14.1 Material Outgassing 73
3.6.14.2 Thermal Vacuum Bakeout 73
3.6.15 Storage and Handling/Material Protection *73*
3.7 INTEGRATED TEST AND EVALUATION PROGRAM 75
3.7.1 Integrated Test and Evaluation Program Plan *75*
3.7.2 Engineering Evaluation Tests *76*
3.7.2.1 Integration Tests 76
3.7.2.2 Interoperability Tests 76
3.7.2.3 Test-Like-You-Fly Approach 76
3.7.3 Qualification/Re-Qualification Test Program *77*
3.7.3.1 Qualification Program Plan 77
3.7.3.2 Qualification Tests 77
3.7.3.2.1 Qualification by Similarity 78
3.7.4 Acceptance Tests *78*
3.7.5 Production Assessment Tests *78*
3.7.6 Surveillance and Service Life Evaluation Tests *79*
3.7.7 Ground and Flight Tests *79*
3.7.7.1 Critical Test Gate Process 79
3.7.7.2 Post-Test Performance Analysis 80
3.7.7.3 Failure Review Process 80
3.7.8 Modeling and Simulation *80*
3.7.9 Test Plans *81*

– **Table of Contents** –

3.7.10 Test Procedures *81*
3.7.11 Test Reports *81*
3.8 TEST, MEASURING, AND DIAGNOSTIC EQUIPMENT AND STANDARDS 83
3.8.1 Selection and Design *83*
3.8.1.1 Test, Measuring, and Diagnostic Equipment Configuration Documentation 83
3.8.1.2 Evaluation of Test, Measuring, and Diagnostic Equipment 83
3.8.1.3 Proofing, Qualification, and Correlation 84
3.8.2 Calibration and Maintenance *84*
3.8.2.1 Calibration and Maintenance Procedures 84
3.8.2.2 Records and Analysis 85
3.8.2.3 Out-of-Tolerance Conditions 85
3.8.2.4 Calibration Standards and Reference Materials 85
3.8.3 General Test, Measuring, and Diagnostic Equipment and Standards Requirements *85*
3.8.3.1 Intervals and Recall 85
3.8.3.2 Labeling 86
3.8.3.3 Sealing for Integrity 86
3.8.3.4 Removal of Test, Measuring, and Diagnostic Equipment and Standards 86
3.8.3.5 Test Station Logs 86
3.9 INTERFACE MANAGEMENT 87
3.9.1 Interface Control Plan *87*
3.9.1.1 Interface Control Plan Development 87
3.9.2 Interface Documentation *87*
3.9.3 Interface Control Working Groups *88*
3.9.4 Interface Change Notice *88*
3.10 CONFIGURATION MANAGEMENT 90
3.10.1 Configuration Management and Planning *90*
3.10.1.1 Identifying Context and Environment 90
3.10.1.2 Configuration Management Plan 90
3.10.1.3 Implementation Procedures 91
3.10.1.4 Training 91
3.10.1.5 Performance Measurement 91
3.10.1.6 Supplier Configuration Management 92
3.10.2 Configuration Identification *92*
3.10.2.1 Product Information 92
3.10.2.2 Product Structure 93
3.10.2.3 Product Identifiers 93
3.10.2.3.1 Identifying Individual Units of Product 94
3.10.2.3.2 Identifying Groups of Units of a Product 94
3.10.2.4 Document Identification 95
3.10.2.5 Baselines 95
3.10.2.5.1 Establishing Baselines 95
3.10.2.5.2 Types of Baselines 95
3.10.2.6 Product Identification Recovery 96
3.10.2.7 Interface Control 96
3.10.3 Configuration Change Management *96*
3.10.3.1 Change Identification 97
3.10.3.1.1 Requesting Changes 97
3.10.3.1.2 Classifying Changes 97
3.10.3.1.3 Documenting Requests for Changes 98
3.10.3.2 Change Evaluation and Coordination 99
3.10.3.2.1 Change Impact Assessment 100
3.10.3.2.2 Change Effectivity Determination 100
3.10.3.2.3 Change Cost/Price Determination 100
3.10.3.2.4 Change Approval Authority 101
3.10.3.3 Change Implementation and Verification 101

3.10.3.4 Change Management Process Applied to Variances 102
3.10.3.4.1 Request for Waiver 102
3.10.3.4.1.1 Restrictions on Waivers 102
3.10.3.4.1.2 Classification of Waivers 103
3.10.3.4.2 Requests for Deviations 103
3.10.3.4.2.1 Restrictions on Deviations 103
3.10.3.4.2.2 Classification of Deviations 104
3.10.3.4.3 Review and Approval of Variances 104
3.10.4 Configuration Status Accounting 104
3.10.4.1 Configuration Status Accounting Information 105
3.10.4.2 Configuration Status Accounting System 106
3.10.5 Configuration Verification and Audit 107
3.10.5.1 Design and Document Verification 107
3.10.5.2 Configuration Audit 107
3.10.5.3 Continuing Performance Audits and Surveillance 108
3.10.6 Configuration Management of Digital Data 108
3.10.6.1 Digital Data Identification 109
3.10.6.2 Data Status Level Management 109
3.10.6.3 Maintenance of Data and Product Configuration Relationships 109
3.10.6.4 Data Version Control and Management of Review, Comment, Annotation, and Disposition 110
3.10.6.5 Digital Data Transmittal 110
3.10.6.6 Data Access Control 110
3.11 CONTROL OF NONCONFORMING ITEMS AND MATERIALS 112
3.11.1 Preliminary Review 112
3.11.2 Material Review Board 112
3.11.2.1 Material Review Board Dispositions 113
3.11.3 Supplier Material Review Board 113
3.12 FABRICATION AND QUALITY 114
3.12.1 Process and Quality Control Planning 114
3.12.2 Process Selection and Development 114
3.12.2.1 Process Selection and Development Planning 114
3.12.2.2 Mission Critical Process Selection 115
3.12.2.3 Special Processes 115
3.12.3 Product Test and Inspection Plan 115
3.12.4 Fabrication and Quality Procedures 116
3.12.4.1 Fabrication and Process Procedures 116
3.12.4.2 Test and Inspection Procedures 116
3.12.4.3 Workmanship Standards 117
3.12.5 Product Control during Fabrication 117
3.12.5.1 Product Identification and Handling 117
3.12.5.2 Product Protection 117
3.12.5.2.1 Electrostatic Discharge Controls 118
3.12.5.2.2 Contamination Control Program 118
3.12.5.2.2.1 Clean Rooms 118
3.12.5.2.3 Foreign Object Elimination Program 118
3.12.5.3 Product Status Indication 118
3.12.6 Fabrication Process Control 119
3.12.6.1 Process Qualification Program 119
3.12.6.2 Fabrication and Quality Metrics 119
3.12.6.3 Fabrication Defects 120
3.12.6.4 Continuous Process Improvement 120
3.12.7 Fabrication Environmental Stress Screening 120
3.12.8 Fabrication Quality Verification 120
3.12.8.1 In-Process and Acceptance Test and Inspection 121

3.12.8.2 First Article Test and Inspection 121
3.12.8.3 Nondestructive Test and Inspection 121
3.12.8.4 Nonconforming Items Control 121
3.12.9 Fabrication and Quality Records 121
3.12.9.1 Fabrication Records 121
3.12.9.2 Quality Control Records 121
3.12.10 Packaging, Handling, Storage, and Transportation 122
3.12.10.1 Packaging 122
3.12.10.2 Handling and Storage 122
3.12.10.3 Preparation for Shipment and Transportation 123
3.13 SUPPLIER MANAGEMENT 124
3.13.1 Supplier Selection 124
3.13.1.1 Conditional Source Approval 125
3.13.2 Supplier Ratings 125
3.13.3 Supplier Evaluations 125
3.13.4 Supplier Program Requirements 126
3.13.5 Procurement Process 126
3.13.5.1 Technical Requirements 126
3.13.5.2 Detailed Provisions 126
3.13.5.3 Procurement Document Review 127
3.13.5.4 Procurement Document Change Control 127
3.13.6 Control of Customer/Government Furnished Material 128
3.13.7 Government Source Inspection 128
3.13.8 Developer Source Inspection 128
3.13.9 Receiving Inspection and Test 129
3.13.10 Intra-Corporate Work Transfers 129
3.14 SAFETY 130
3.14.1 Safety Program Requirements 130
3.14.2 Safety Program Documentation, Reports, and Working Groups 131
3.14.2.1 System Safety Program Plan 131
3.14.2.2 Safety Analyses 133
3.14.2.3 Safety Variance Reporting 134
3.14.2.4 Safety Assessment Reporting 134
3.14.2.5 Sustained Engineering 134
3.14.2.6 System Safety Working Groups 134
3.14.2.7 Hazard Identification and Tracking 134
3.14.2.8 Safety Verification 134
3.14.2.9 Safety Assessment 134
3.14.2.9.1 Safety Defect / Deficiency Assessment 135
3.14.2.10 System Safety Program Reviews/Audits 135
3.14.3 System Safety Requirements 136
3.14.3.1 System Safety Engineering Approach 137
3.14.3.2 System Safety Hazard Identification and Analysis Methodology 137
3.14.3.3 Assessment of Mishap Risk 138
3.14.3.4 Risk Acceptance Authority 138
3.14.3.5 Mishap Investigations 139
3.14.4 Safety Design Criteria 139
3.14.4.1 Unacceptable Conditions 139
3.14.4.2 Acceptable Conditions 139
3.14.4.3 Ignition System Safety Requirements 140
3.14.4.4 Fuze System Safety Requirements 140
3.14.4.5 Hazardous Materials Transportation 140
3.14.4.6 Insensitive Munitions Design and Safety Tests 140
3.14.4.7 Ordnance Systems 141
3.14.4.8 Missile and Space Vehicle Pressure Systems 141

3.14.4.9 Orbital Debris 141
3.14.4.10 Human Engineering 141
3.14.5 Occupational Safety and Health 141
3.14.5.1 Safety and Health 141
3.14.5.2 Hazardous Materials Management 142
3.14.5.3 Test Safety 142
3.14.6 Range Safety 143
3.14.6.1 Flight Termination Systems 143
3.14.6.1.1 Flight Termination System and Range Safety Tracking System Standards 143
3.14.6.2 Flight Safety Analysis 144
3.14.7 Safety Critical Computing System Functions 144
3.14.8 Software Safety 144
3.14.8.1 Modular Code 145
3.14.8.2 Number of Modules 145
3.14.8.3 Execution Path 145
3.14.8.4 Halt Instructions 145
3.14.8.5 Single Purpose Files 145
3.14.8.6 Unnecessary Features 145
3.14.8.7 Indirect Addressing Methods 145
3.14.8.8 Uninterruptible Code 145
3.14.8.9 Safety Critical Files 146
3.14.8.10 Unused Memory 146
3.14.8.11 Overlays 146
3.14.8.12 Operating System Functions 146
3.14.8.13 Compilers 146
3.14.8.14 Flags and Variables 146
3.14.8.15 Loop Entry Point 146
3.14.8.16 Software Maintenance Design 146
3.14.8.17 Variable Declaration 146
3.14.8.18 Unused Executable Code 147
3.14.8.19 Unreferenced Variables 147
3.14.8.20 Assignment Statements 147
3.14.8.21 Conditional Statements 147
3.14.8.22 Strong Data Typing 147
3.14.8.23 Timer Values Annotated 147
3.14.8.24 Critical Variable Identification 147
3.14.8.25 Global Variables 147
3.14.8.26 Software Formal Test Coverage 147
3.14.9 Software Maintenance Requirements 148
3.14.10 Design and Development of Computer Systems 148
3.14.10.1 General Design Requirements 149
3.14.10.2 Two Person Rule 149
3.14.10.3 Program Patch Prohibition 149
3.14.10.4 Design Verification and Validation 149
3.14.11 System Design Requirements for Computer Systems 149
3.14.11.1 Designed Safe States 149
3.14.11.2 Standalone Computer 149
3.14.11.3 Ease of Maintenance 149
3.14.11.4 Safe State Return 150
3.14.11.5 Restoration of Interlocks 150
3.14.11.6 Input/Output Registers 150
3.14.11.7 External Hardware Failures 150
3.14.11.8 Safety Kernel Failure 150
3.14.11.9 Circumvent Unsafe Conditions 150
3.14.11.10 Fallback and Recovery 150

3.14.11.11 Simulators 150
3.14.11.12 System Errors Log 150
3.14.11.13 Positive Feedback Mechanisms 150
3.14.11.14 Peak Load Conditions 151
3.14.11.15 Endurance 151
3.14.11.16 Corruption of Computing Environment 151
3.14.12 Power-Up System Initialization Requirements 151
3.14.12.1 Power-up Initialization 151
3.14.12.2 Power Faults 151
3.14.12.3 Primary Computer Failure 151
3.14.12.4 Maintenance Interlocks 152
3.14.12.5 System Level Check 152
3.14.13 Munitions/Directed Energy Systems Hardware Requirements 152
3.14.13.1 Central Processing Units Selection 152
3.14.13.2 Minimum Clock Cycles 152
3.14.13.3 Read-Only-Memories 152
3.14.14 Self-Checking Design Requirements 152
3.14.14.1 Watchdog Timers 152
3.14.14.2 Memory Checks 153
3.14.14.3 Fault Detection 153
3.14.14.4 Operational Checks 153
3.14.15 Safety Critical Computing System Functions and Data 153
3.14.15.1 Safety Degradation 153
3.14.15.2 Unauthorized Interaction 153
3.14.15.3 Unauthorized Access 153
3.14.15.4 Safety Kernel 153
3.14.15.5 Inadvertent Jumps 153
3.14.15.6 Load Data Integrity 154
3.14.15.7 Operational Reconfiguration Integrity 154
3.14.16 Interface Design Requirements 154
3.14.16.1 Feedback Loops 154
3.14.16.2 Interface Control 154
3.14.16.3 Decision Statements 154
3.14.16.4 Inter-CPU Communications 154
3.14.16.5 Data Transfer Messages 154
3.14.16.6 External Functions 154
3.14.16.7 Input Reasonableness Checks 154
3.14.16.8 Full Scale Representations 155
3.14.17 User Interface to Safety Critical Computing Systems 155
3.14.17.1 Processing Cancellation 155
3.14.17.2 Hazardous Function Initiation 155
3.14.17.3 Safety Critical Displays 155
3.14.17.4 Operator Entry Errors 155
3.14.17.5 Safety Critical Alerts 155
3.14.17.6 Unsafe Situation Alerts 155
3.14.17.7 Unsafe State Alerts 156
3.14.18 Critical Timing and Interrupt Functions 156
3.14.19 MDA Safety Integration 156
3.14.19.1 Developer and Supplier Integrator Responsibility 156
3.14.19.2 Non-Integration Developers and Suppliers 157
3.14.20 Flowdown of Requirements from Developer to Supplier 157
3.14.21 Safety Metrics 158

4.0 NOTES 160

4.1 CUSTODIAN 160

– **Table of Contents** –

APPENDIX (A).....1

This Page Intentionally Left Blank

1.0 SCOPE

The mission assurance program encompasses development, engineering, testing, production, procurement, and implementation of missile defense elements under the cognizance of MDA. The program uses quality assurance as a tool in mission assurance. The outcome is an effective, reliable, and safe missile defense capability. The MDA Assurance Provisions (MAP) provides a measurable, standardized set of Quality, Safety, and Mission Assurance requirements to be applied to developers for mission and safety critical items in support of evolutionary acquisition and deployment of MDA systems. ***Mission and safety critical items are those items where failure would directly affect system or personnel safety, mission success, or operational readiness.***

Mission Assurance is defined as: (1) An engineering level assurance process performed over the lifecycle of a program to identify and mitigate design, production and test deficiencies that could effect mission success. (2) The disciplined application of general system engineering, risk management, quality, and management principles to achieve mission success. A disciplined mission assurance process has independent technical assessment as a cornerstone throughout the entire design, development, testing, deployment, and operations process. Mission assurance is a cradle-to-grave process.

1.1 Purpose

The MAP establishes Quality, Safety, and Mission Assurance processes and actions through the disciplined application of general system engineering; interface, configuration, and risk management; quality, safety, and management principles needed to achieve mission success throughout the evolutionary acquisition process. It provides MDA with methods to measure, verify, and validate mission success through the collection of metrics; risk assessment; technical evaluations, independent assessments, and reviews; and conduct of test.

The implementation of MAP disciplines promotes continuous process improvement to reduce costs; improve productivity; and enhance Quality, Safety, and Mission Assurance.

1.2 Applicability and Accountability

The provisions of the MAP apply to developers and suppliers involved in designing, developing, fabricating, testing, deploying, and supporting systems under the cognizance of MDA. Developers and suppliers include Deputates and Elements, prime contractors, subcontractors, sub-tier suppliers, integrators, and Government laboratories. DoD Services are encouraged to use the MAP upon transition to maintain Quality, Safety, and Mission Assurance disciplines throughout the acquisition process.

All developers of MDA products and services shall establish and maintain accountability for fulfilling the Quality, Safety, and Mission Assurance requirements herein. Accountability shall be documented through the assignment of specific roles, responsibilities, and authorities.

1.2.1 Mission Assurance Implementation Plan

The Deputates responsible for an overall MDA function (e.g., DE-Systems Engineering, Models and Simulations; DT-Responsible Test Organization) shall develop a Mission Assurance Implementation Plan (MAIP) to describe how the MAP is implemented on their programs.

1.2.2 MAP on Contract:

Deputates and Elements responsible for the design, development, fabrication, production, test, support or repair of MDA systems by prime contractors shall place the MAP on contract with a MDA/QS approved letter for any alternate approaches, modifications or exceptions of provision requirements.

1.3 Applicable Documents

The following documents form a part of this specification to the extent specified herein. Request for use of revisions to these specifications shall be submitted to MDA/QS as part of an update to the MAIP.

Aerospace Report No. TR-99 (1413)-1	Natural and Triggered Lightning Launch Commit Criteria	Jan 99
AIAA S-080	Standard for Space Systems - Metallic Pressure Vessels, Pressurized Structures, and Pressure Components	1998
AIAA S-081	Standard for Space Systems - Composite Overwrapped Pressure Vessels	2000
ANSI/EIA-632-1998	Processes for Engineering a System	Jan 99
ANSI/EIA-649-1998	National Consensus Standard for Configuration Management	Jul 98
ANSI/NCSL Z540-1-1994	American National Standard for Calibration – Calibration Laboratories and Measuring and Test Equipment-General Requirements	Jul 94
IPC/EIA J-STD-001C (Class III)	Requirements For Soldered Electrical and Electronic Assemblies	Mar 00
IPC/EIA J-STD-001CS	Space Applications Electronic Hardware Addendum to Requirements for Soldered Electrical and Electronic Assemblies	Jul 03
ASME Y14.24M-1999	Types and Applications of Engineering Drawings	1999
ASME Y14.34M-1998	Associated Lists	1996
CJCS Instruction 3170.01C	Joint Capabilities Integration and Development System	Jun 03
CJCS Manual 3170.01	Operations of Joint Capabilities Integration and Development System	Jun 03
DOD 4145.26-M	DOD Contractor's Safety Manual for Ammunition and Explosives	Sep 97
DODD 5000.1	The Defense Acquisition System	May 03
DODI 5000.2	Operation of the Defense Acquisition System	May 03
DODD 3200.11	Major Range and Test Facility Base	May 02
EWR 127-1	Eastern and Western Range Safety Requirements	Mar 95
IEEE Standard 1012-1998	Software Verification and Validation	1998
IPC-2220 Series (Class III)	Series of Printed Board Design Documents	Feb 00
IPC-6011 Series (Class III)	Series of Generic Performance Specification for Printed	Feb 00

	Boards	
MDA Policy Memorandum No. 33	Director's Safety Policy	Dec 02
MIL-STD-810F (including notices 1-3)	Environmental Engineering Considerations and Laboratory Tests	May 03
MIL-STD-464A	Electromagnetic Environmental Effects Requirements for Systems	Dec 02
MIL-STD-882D	System Safety	Feb 00
MIL-STD-1316E(1)	Fuze Design, Safety Criteria For	Jan 99
MIL-STD-1472F	Human Engineering	Aug 99
MIL-STD-1522A	General Requirements for Safe Design and Operation of Pressurized Missile and Space Systems	Jun 84
MIL-STD-1576	Electroexplosive Subsytem Safety Requirements and Test Methods For Space Systems	Sep 92
MIL-STD-1686C	Electrostatic Discharge Control Program for Protection of Electrical and Electronic Parts, Assemblies and Equipment (Excluding Electrically Initiated Explosive Devices)	Oct 95
MIL-STD-1901A	Munitions Rocket and Missile Motor Ignition System Design, Safety Criteria For	Jun 02
MIL-STD-2105C, Section 5.2	Hazard Assessment Test for Non-Nuclear Munitions	Jul 03
NSS 1740.14	NASA Safety Standard, Guidelines and Assessment Procedures for Limiting Orbital Debris	Aug 95
NASA-STD-8739.4	Crimping, Interconnecting Cables, Harnesses, and Wiring	Feb 98
NASA-STD-8739.5	Fiber Optics Terminations, Cable Assemblies, and Installation	Feb 98
NAVSEAINST 9310.1B	Naval Lithium Battery Safety Program	Jun 91
OSHA Form 174	Material Safety Data Sheet	
RCC-106-01	Telemetry Standards	May 01
RCC-254-94	Non-Coherent Transponder Standards	Apr 94
RCC-319	Flight Termination Commonality Standard	
RCC-321-02	Common Risk Criteria for National Test Ranges: Inert Debris	Jun 02
RCC-324-01	Global Positioning and Inertial Measurements Range Safety Tracking Systems Commonality Standard	Jun 01

SAE AS9100 Rev A	Quality Systems – Aerospace – Model for Quality Assurance in Design, Development, Production, Installation and Servicing	Aug 01
SD-18	Part Requirement and Application Guide	Oct 02
TB-700-2/ NAVSEAINST 8020.8/ DLAR 8220.1	Explosives Hazard Classification Procedures	Jan 98
Executive Order 12196	Occupational Safety and Health Programs For Federal Employees, 3 CFR (1980 Compilation), as amended, 5 U.S.C. 7902 (note)	Feb 80
Public Law 91-596, 29 USC 651-678	Occupational Health and Safety Act	
Public Law 10 USC 141 Section 2389	Armed Forces Miscellaneous Procurement Provisions: Ensuring Safety Regarding Insensitive Munitions	
29 CFR 1960	Basic Program Elements for Federal Employee Occupational safety and Health Programs and Related Matters	
49 CFR 100-199	Transportation	

1.4 Order of Precedence

In the event of conflict between the text of the MAP and the applicable documents cited herein, with the exception of public law, the text of the MAP takes precedence.

1.5 Conflicting Requirements

Conflicts between the provisions of this document and other documents shall be referred, in writing, to the cognizant MDA Deputates or Elements and MDA/QS for interpretation, clarification, resolution, or correction.

2.0 DEFINITIONS

For the purposes of this document the following definitions apply. Other definitions used in this document may be found in the MDA glossary at http://www.mda.mil/mdalink/pdf/glossary.pdf.

Acceptance: The act of an authorized representative of the government by which the government, for itself, or as agent of another, assumes ownership of existing identified supplies tendered, or approves specific services rendered, as partial or complete performance of the contract or work authorization.

Alternate Part: A part that is equal to or better than the part specified on a parts list.

Autonomous Software Control: Software control that does not require human intervention to process data or issue commands. In this sense, a fault, failure, or defect in the software will lead to a hazard or a mishap over which the operator has no control.

Baseline: (1) An agreed-to description of the attributes of a product, at a point in time, which serves as a basis for defining change. (2) An approved and released document, or a set of documents, each of a specific revision; the purpose of which is to provide a defined basis for managing change. (3) The currently approved and released configuration documentation. (4) A released set of files comprising a software version and associated configuration documentation.

Battleshort (Safety Arc): The capability to bypass certain safety features in a system to ensure completion of a mission without interruption due to the safety feature. Bypassed safety features include such items as circuit current overload protection, and thermal protection.

Commercial-Off-The-Shelf Items: Products or equipment developed by industry for sale in the general commercial market place. Commercial items may include modifications, provided the modifications are either:

a. Of a type customarily available in the commercial marketplace.

b. Of a type not customarily available in the commercial marketplace, which do not alter the non-governmental function, essential physical characteristics of an item or component, or change the purpose of a process.

Configuration: (1) The performance, functional, and physical attributes of an existing or planned product, or a combination of products. (2) One of a series of sequentially created variations of a product.

Configuration Audit: Product configuration verification accomplished by inspecting documents, products and records; and reviewing procedures, processes, and systems of operation to verify that the product has achieved its required attributes (performance requirements and functional constraints) and the product's design is accurately documented. Sometimes divided into separate functional and physical configuration audits.

Configuration Change Management; Configuration Control: (1) A systematic process which ensures that changes to released configuration documentation are properly identified, documented, evaluated for impact, approved by an appropriate level of authority, incorporated, and verified. (2) The configuration management activity concerning: the systematic proposal, justification, evaluation, coordination, and disposition of proposed change; and the implementation of all approved and released changes into (a) the applicable configurations of a product, (b) associated product information, and (c) supporting and interfacing products and their associated product information.

Configuration Control Board (CCB): A board composed of technical and administrative representatives who recommend approval or disapproval of proposed engineering changes to a CI's current approved

configuration documentation. The board also recommends approval or disapproval of proposed waivers and deviations from a CI's current approved configuration documentation.

Configuration Documentation: Technical information, the purpose of which is to identify and define a product's performance, functional, and physical attributes (e.g., specifications, drawings).

Configuration Identification; Product Definition: (1) The systematic process of selecting the product attributes, organizing associated information about the attributes, and stating the attributes. (2) Unique identifiers for a product and its configuration documents. (3) The configuration management activity that encompasses selecting configuration documents; assigning and applying unique identifiers to a product, its components, and associated documents; and maintaining document revision relationships to product configurations.

Configuration Management (CM): A management process for establishing and maintaining consistency of a product's performance, functional, and physical attributes with its requirements, design and operational information throughout its life.

Configuration Status Accounting (CSA); Product Configuration Information: The configuration management activity concerning capture and storage of, and access to, configuration information needed to manage products and product information effectively.

Configuration Verification: The action of verifying that the product has achieved its required attributes (performance requirements and functional constraints) and the product's design is accurately documented.

Controlled Items: An assembly or subassembly, known or suspected to be subject to a rate of deterioration with operation sufficient to cause reliability degradation during service life. Controlled items require the recording of operating data for reliability evaluation or preventative maintenance purposes. The operating data to be tracked includes operating time, operating cycle, or power turn-on, as applicable to the item. The term "controlled items" excludes items that deteriorate solely based on calendar age (see "limited life items").

Derating: The reduction of the applied load (or rating) of a device to improve reliability or to permit operation at high ambient temperatures.

Designed Safe State: A system state that provides the maximum degree of safety within the constraint of the current operational or logistic phase.

Destructive Physical Analysis: An internal destructive examination of a finished part or device to assess design, workmanship, assembly, and any other processing associated with fabrication of the part.

Developer: Any entity that manages or performs design, development, fabrication, production, test, support, or repair of MDA cognizant product.

Devices: (1) A hardware item or assembly that can be further disassembled. (2) A piece of equipment or a mechanism designed to serve a special purpose or perform a special function.

EEE Parts: EEE parts include capacitors, connectors, crystal oscillators, diodes and transistors, fiber optics, filters, fuses, hybrid microcircuits, monolithic microcircuits, magnetics, relays, resistors, thermistors, wire, and cable.

Energetics: A system that uses explosives, propellants, directed energy, pyrotechnics, initiating composition, or nuclear, biological, or chemical material for use in military operations.

Engineering Change: Any alteration to a product or its released configuration documentation. Effecting an engineering change may involve modification of the product, product information and associated interfacing products.

Establish and Maintain: Establish and maintain includes planning, developing, preparing, implementing, documenting, assessing, updating, and performing.

Fabrication: The process of converting raw materials into required material. It includes the functions of scheduling, inspection, quality control, and related processes.

Facility Location: The location where the purchased item is actually built or produced. This does not include the location of a vendor or a distributor who simply sells supplies that they do not produce.

Failure: An event in which an item does not perform one or more of its required functions within the specified limits under specified conditions. A failure can either be catastrophic (total loss of function) or out-of-tolerance (degraded function beyond specified limits due to such occurrences as part failure, detuning, misalignment, and maladjustment, which are often classified as faults).

Failure Modes and Effects Analysis: A procedure by which each credible failure mode of each item from a low indenture level to the highest is analyzed using inductive logic to determine the effects on the system and to classify each potential failure mode in accordance with the severity of its effect.

Fault Tree Analysis: A process of reviewing and analytically examining a system or equipment in such a way as to emphasize the lower-level fault occurrences, which directly or indirectly contribute to the major fault or undesired event. Fault tree analysis emphasizes a pictorial presentation and deductive logic.

Firmware: The combination of a hardware device and computer instructions or computer data that reside as read-only software on the hardware device. The software cannot be readily modified under program control.

Foreign Object Damage: Damage to product caused by a foreign object.

Foreign Object Debris: Foreign material which could potentially cause product damage or degraded performance.

Hardware: Products made of material and their components (mechanical, electrical, electronic, hydraulic, pneumatic). Computer software and technical documentation are excluded.

Hazard: Any real or potential condition that can cause injury, illness, or death to personnel; damage to or loss of a system, equipment or property; or damage to the environment. A hazard is a prerequisite to a mishap.

Interface: The performance, functional, and physical attributes required to exist at a common boundary.

Interface Control: The process of identifying, documenting, and controlling all performance, functional, and physical attributes relevant to the interfacing of two or more products provided by one or more organizations.

Life Cycle: A generic term relating to the entire period of concept refinement and technology development; system development and demonstration; production and deployment; operations and support; and disposal of a product.

Limited Life Items: A component, part, or material known or suspected to be subject to a rate of deterioration with calendar time sufficient to cause reliability degradation prior to installation or during service life. Limited life items required calendar age controls prior to acceptance to prevent the use of

over-age components, parts, or materials and to provide a baseline for reliability evaluation or preventative maintenance purposes.

Maintainability: A measure of the ease and rapidity with which a system or equipment can be restored to operational status following a failure. It is characteristic of equipment design and installation, personnel availability in the required skill levels, adequacy of maintenance procedures and test equipment, and the physical environment under which maintenance is performed.

Material: A substance required for production or fabrication of a product. For example the term material includes epoxies, adhesives, propellant, core material, alloys, mixtures, and compounds.

Mishap: An unplanned event or series of events resulting in death, injury, occupational illness, damage to or loss of equipment or property, or damage to the environment.

Mission Assurance: (1) An engineering level assurance process performed over the lifecycle of a program to identify and mitigate design, production and test deficiencies that could effect mission success. (2) The disciplined application of general system engineering, risk management, quality, and management principles to achieve mission success. A disciplined mission assurance process has independent technical assessment as a cornerstone throughout the entire design, development, testing, deployment, and operations process. Mission assurance is a cradle-to-grave process.

Mission Critical Item: A mission critical item, if defective will prevent command and control, sensors, weapons, combat, or flight systems from achieving mission primary objectives. A failure of the mission critical item would affect system or personnel safety, mission success, or operational readiness. Examples of mission critical items include, but are not limited to: items having limited operating life (controlled items), one shot devices, items causing single points of failure, or items that can not be tested before flight or use.

Nonconformance: A condition of any hardware, software, material, or service in which one or more characteristics do not conform to requirements.

Outgassing: The emanation of volatile materials under vacuum conditions resulting in a mass loss and/or material condensation on nearby surfaces.

Open System: A system that implements sufficient open specifications for interfaces, services, and supporting formats to enable properly engineered components to be utilized across a wide range of systems with minimal changes, to interoperate with other components on local and remote systems, and to interact with users in a style that facilitates portability.

Part: A hardware element that is not normally subject to further subdivision or disassembly without destruction of design use. Examples include resistor, integrated circuit, relay, connector, bolt, and gaskets.

Performance: A quantitative measure characterizing a physical or functional attribute relating to the execution of an operation or function. Performance attributes include quantity (how many and how much), quality (how well), coverage (how much area, how far), timeliness (how responsive, how frequent), and readiness (availability, mission/operational readiness). Performance is an attribute for all systems, people, products and processes including those for development, production, verification, deployment, operations, support, training, and disposal. Thus supportability parameters, manufacturing process variability, reliability and so forth, are all performance measures.

Positive Control (During Flight): The ability to reduce to an acceptable level, the hazards associated with the errant flight of a launch vehicle during test and operations; normally achieved with a flight and/or thrust termination system.

Product: Anything that is used or produced to satisfy a need, for example, facilities, systems, hardware, software, firmware, data, processes, materials, or services.

Qualification Test: These tests simulate defined environmental conditions with a predetermined safety factor (margin), the results indicating whether a given design can perform its function within the expected mission environment for the system. These tests are preformed on items that are representative of their expected fielded configuration.

Reliability: The probability that an item will perform its intended function for a specified interval under stated conditions.

Repair: A corrective maintenance action performed as a result of a failure so as to restore an item to operate within specified limits.

Responsible Test Organization: The lead government entity that is responsible for Developmental Test and Evaluation.

Residual Safety Risk: The remaining mishap risk that exists after all mitigation techniques have been implemented or exhausted, in accordance with the system safety design order of precedence.

Rework: Return for completion of operations (complete to drawing). The article is to be reprocessed to conform to the original specifications or drawings.

Risk: A measure of the inability to achieve program objectives within defined cost and schedule constraints. Risk is associated with all aspects of the program, e.g., threat, technology, design processes, or work breakdown structure elements. It has two components, the probability of failing to achieve a particular outcome, and the consequences of failing to achieve that outcome.

Risk Analysis: A detailed examination of each identified program risk, which refines the description of the risk, isolates the cause, and determines the impact of the program risk in terms of its probability of occurrence, its consequences, and its relationship to other risk areas or processes.

Risk Identification: The process of examining the program areas and each critical technical process to identify and document the associated risk.

Risk Management: The act or practice of dealing with risk. It includes planning for risk, assessing (identifying and analyzing) risk areas, developing risk-handling options, monitoring risks to determine how risks have changed, and documenting the overall risk management program. It includes plans and actions taken to identify, assess, mitigate, continuously track, control, and document program risks.

Risk Mitigation: Risk mitigation is the process of avoiding, reducing and controlling, or deliberately accepting risk on the program.

Risk Monitoring: A process that systematically tracks and evaluates the performance of risk items against established metrics throughout the acquisition process and develops further risk reduction handling options, as appropriate.

Safety Critical: A term applied to a condition, event, operation, process, or item whose proper recognition, control, sequencing, performance or tolerance is essential for safe system operation or use.

Safety Kernel: An independent computer program that monitors the state of the system to determine when potentially unsafe system states may occur, or when transitions to potentially unsafe system states may occur. The Safety Kernel is designed to prevent the system from entering the unsafe state and to return it to a known safe state.

Software Quality: The ability of software to satisfy its specified requirements.

Software Reuse: The process of implementing or updating software systems using existing software assets.

Software Safety Critical: A condition, event, operation, process, or item of whose proper recognition, control, performance or tolerance is essential to safe system operation or use. Safety Critical Software includes firmware and software programs or routines whose incorrect, inadvertent or improper functioning, functioning in an improper sequence, or failure to function when required, can result in a hazard, or loss of predictability or control of the system.

Strong Data Typing: A fault tolerance technique wherein a discrete or variable data is represented by a bit pattern that is unique for each valid value and cannot be confused with any other valid value even as a result of a one or two bit error.

Substitute Part: A part whose performance may be less capable than the part specified on a parts list for one or more reasons (e.g., quality or reliability level, tolerance, parametric, temperature range).

Supplier: An entity that provides a product or service. The term supplier also encompasses subcontractors and vendors.

System Loss: Within the context of this document, system loss does not refer to unavailability of a system; rather, it implies significant rework required or replacement is required to return the system to its undamaged state.

Unsafe State: A system state that may result in a hazard/mishap.

Variance (Deviation or Waiver): A specific written authorization to depart from a particular requirement of a product's current approved configuration documentation for a specific number of units or a specified time period. (A variance differs from an engineering change in that an approved engineering change requires corresponding revision of the product's current approved configuration documentation, whereas a variance does not).

Version: An identified and documented body of software. Modifications to a version of software (resulting in a new version) require configuration management actions by the contractor, government, or both.

Watchdog Timer: An independent, external timer that ensures that the computer cannot enter an infinite loop. Watchdog timers are normally reset by the computer program. Expiration of the timer results in generation of an interrupt, program restart, or other function that terminates current program execution.

3.0 QUALITY, SAFETY, AND MISSION ASSURANCE PROVISIONS

3.1 Management

The developer shall establish and maintain fundamental management disciplines necessary to plan, establish, and monitor a Quality, Safety, and Mission Assurance program. It shall include requirements for internal and external communication, sharing information, mitigating risk, and encouraging continuous improvement. This is accomplished by establishing management programs, including policy, planning, training, documentation, and review processes necessary to execute the Quality, Safety, and Mission Assurance Program. The developer shall have a Quality Management System that is compliant with requirements of AS9100, Aerospace Model for Quality Assurance in Design, Development, Production, Installation and Servicing. Assurance related activities not covered by AS9100 requirements are identified in the following sections and supplement AS9100 requirements.

3.1.1 Contract Reviews

The developer shall establish and maintain a process for contract reviews as an element of contract administration. All phases of the contracting process shall be managed from the proposal stage through definitization, negotiation of requirements, and finalization of contract requirements. Results of the review process shall be documented and used to communicate requirements to various supporting organizations and disciplines within the program. Program management shall conduct contract reviews with participation from affected disciplines (e.g., contracts, quality, manufacturing, engineering, configuration, and materiel). The developer shall verify contracts using the criteria listed below, as a minimum:

a. The supplier has the capability to satisfy the requirements.

b. The requirements are consistent and cover user needs.

c. Adequate procedures for handling changes to requirements and escalating problems are stipulated.

d. Procedures and their extent for interface and cooperation among the parties are stipulated, including ownership, warranty, copyright, licenses, and confidentiality.

e. Acceptance criteria and procedures are stipulated in accordance with requirements.

Amendments or modifications to a contract shall result in a review of requirement changes with affected disciplines as deemed necessary by the developer's program manager. Records of contract reviews shall be retained in accordance with record retention requirements.

3.1.2 Management Reviews

Management reviews discussed in AS9100 shall also include information on: results of metrics monitoring, external audits, analysis of product data, and risk assessments. Output from management reviews shall include assessments related to the effectiveness of systems used to implement Quality, Safety, and Mission Assurance requirements, including program/technical performance results. The developer shall use output of management reviews to continually improve effectiveness of the Quality, Safety, and Mission Assurance program. Records from management reviews, including any actions assigned, shall be maintained and made available to MDA Deputates and Elements or designated representatives upon request.

3.1.3 Metrics

The developer shall establish and maintain metrics to monitor program and process effectiveness. The developer shall have an approach and methodology to identify and select metrics to monitor and control critical program and process requirements throughout the acquisition process. When analyzing and

reporting metrics, the developer shall assess the validity and performance of each metric. Metrics shall include parameters used for measuring continuous process improvement and for assessing effectiveness of Safety, Quality, and Mission Assurance requirements implementation. Metrics shall be included as part of the risk management program. MDA core metrics shall be reported monthly to MDA QS or designated representative. All other metrics shall be made available to the cognizant MDA Deputate and Element or designated representative upon request.

3.1.3.1 MDA Core Metrics

The developer shall establish and maintain a system for collection and analysis of core MDA metrics for hardware and software products and processes. Core metrics shall be reported to MDA/QS on a monthly basis. The following program and software metrics are designated as the MDA core metrics.

3.1.3.1.1 Failure Review Board Actions

The developer shall establish and maintain a system for collection and monthly reporting of failure review board actions (3.5.4). Failure review board metrics shall be established to capture test failures. Metrics to be collected, analyzed, and reported include:

a. Total number of failure review board actions (open vs. closed).

b. Cycle time of active failure review boards actions (aging).

3.1.3.1.2 Waivers and Deviations

The developer shall establish and maintain a system for collection and monthly reporting of waivers and deviations (3.10.3.4). Waiver and deviation metrics to be reported include:

a. Total number of waivers (submitted and approved):

   1) By classification.

   2) Number recurring.

b. Total number of deviations (submitted and approved):

   1) By classification.

   2) Number recurring.

3.1.3.1.3 Class I and Class II Change Requests

The developer shall establish and maintain a system for collection and monthly reporting of Class I and Class II change requests that are processed by the configuration control board managed by the developer at their level of authority. Class I and Class II change request metrics to be collected, analyzed, and reported include:

a. Total number of Class I change requests (submitted and approved).

b. Total number of Class II change requests (submitted and approved).

c. Cycle time required for approval.

3.1.3.1.4 Design Escapes

The developer shall establish and maintain a system for collection and monthly reporting of design escapes. Design escapes include failures and defects found through analysis attributed to design. Metrics to be collected, analyzed, and reported include:

a. Number detected.

b. Where escape was detected.

c. Disposition of escape.

3.1.3.1.5 Foreign Object Elimination

The developer shall establish and maintain a system for collection and monthly reporting of Foreign Object Damage and Debris (FOD) incidents (occurrence). Metrics to be collected, analyzed, and reported include:

a. Total number of FOD related incidents (damage and debris).

b. Total number of FOD related incidents resulting in scrap, rework, or repair.

3.1.3.1.6 Cost of Quality

The developer shall establish and maintain a system for collection and monthly reporting of cost of quality. The developer shall use the following formula to determine and report the cost of quality metric ratio:

$$\frac{\text{Cost of Prevention \& Detection}}{\text{Revenue}} : \frac{\text{Cost of Defects \& Failures}}{\text{Revenue}}$$

Costs of prevention and detection include costs associated with any process or product, which is designed to eliminate current or potential non-compliances to specifications.

Costs of defects and failures include costs associated with resolving any defect in a non-compliant process or product after the defect has been identified.

Both costs are compared to contract value (revenue) for the project.

3.1.3.1.7 Program Quality Staffing

The developer shall establish and maintain a system for collection and monthly reporting of the number of on–board versus planned quality personnel.

3.1.3.1.8 Software Metrics

The developer shall establish and maintain a system for collection and monthly reporting of software metrics. Metrics to be collected, analyzed, and reported include:

a. *Earned Value Work Performed (Schedule Performance Index (SPI))*. Values to be collected and reported by the developer are budgeted cost of work performed (BCWP) and budgeted cost of work scheduled (BCWS) through the end of the reporting period.

b. *Earned Value Costs (Cost Performance Index (CPI))*. Values to be collected and reported by the developer are budgeted cost of work performed (BCWP) and actual cost of work performed (ACWP) through the end of the reporting period.

c. *Staffing Adequacy (SA)*. Values to be collected and reported by the developer are the number of staff, categorized by seniority, who have left the team during the reporting period, and actual and planned staffing levels through the end of the reporting period.

d. *Functionality Volatility Index (FVI)*. Values to be collected and reported by the developer are the total number of functional capability requirements, and the changes to the functionality. A change in functionality is defined as a new function, a change to an existing function, or the slippage of a function's implementation. The FVI is the percentage of functions changed during the reporting period.

e. *Software Requirements Index (SRI)*. Values to be collected and reported by the developer are total number of requirements allocated to software and the changes to these requirements. A change in requirements is defined as requirements added, modified, deleted or slippage of a requirements implementation. The SRI is the percentage of requirements changed during the reporting period.

f. *Software Test Coverage*. Values to be collected and reported by the developer are tests scheduled through the end of the reporting period, tests successfully completed through the end of the reporting period, total number of requirements, and number of requirements covered by tests.

g. *Source Lines of Code (SLOC)*. Values to be collected and reported by the developer are the estimated total lines of the code to be delivered at completion, and the actual number of lines of code developed through the end of the reporting period.

h. *Defect History*. Values to be collected and reported by the developer are the number of defects opened and closed in the developed software during the reporting period, categorized by criticality.

i. *Defect Density*. Values to be collected and reported by the developer are the cumulative number of defects discovered per estimated KSLOC of developed software, categorized by criticality.

3.1.4 Management Information Systems

The developer shall establish and maintain a Management Information System (MIS) for the effective collection, control, processing, use, storage, retention of data, expert knowledge, success reports, and lessons learned from current and previous programs. The MIS shall support design, quality of fabricated items, RM&A program, test programs, safety program, and corrective action system throughout the evolutionary development process. The developer shall ensure that data, expert knowledge, and lessons learned during development, fabrication, and operational support is distributed to appropriate personnel and can be readily identified, accessed, and retrieved by all internal organizations and designated government representatives that require use of the data.

The developer is encouraged to participate in the MDA Lessons Learned (MDALL) program used to collect, organize, track, and disseminate lessons learned throughout MDA and the BMDS community.

3.1.5 Risk Management Program

The developer shall establish and maintain a risk management program to continuously identify, analyze, mitigate, monitor, and report systems engineering process, product, technology, cost, schedule and other program risks. Metrics shall be developed to document the effectiveness of the risk management program. Results of the risk management process shall be used for continuous improvement and risk reduction. The developer shall make risk management documentation and data available to MDA Systems Engineering & Integration, cognizant MDA Deputates and Elements, and designated representative. The developer shall report status of high and moderate risk areas and corresponding mitigation plans and activities at management and program reviews. In addition, when identified, any risk item impacting a critical path shall be immediately brought to the attention of MDA Systems Engineering & Integration and the cognizant MDA Deputates and Elements.

The developer shall support MDA incremental risk assessments at their facilities and at mission critical suppliers in support of design and mission assurance reviews (3.4).

3.1.5.1 Risk Management Plan

The developer shall develop a Risk Management Plan that describes the risk management approach to be used on the program, including appropriate tools and techniques used to identify and mitigate risk. In the implementation of the plan, the following aspects shall be considered, as a minimum:

a. Likelihood and severity of risks or their uncertainties expected in demonstration of design performance, and with items having small design margins.

b. Risks identified by reliability and safety analyses.

c. Likelihood and severity of risks or their uncertainties expected in development of new products, components, parts, materials, processes, and critical technologies.

d. Likelihood and severity of risks or their uncertainties expected in procurement, manufacturing, assembly, inspection, test, handling, storage and transportation, which may lead to unacceptable degradations in product quality.

e. Likelihood and severity of risks anticipated in product utilization or service implementation.

f. Risk identified by suppliers at lower levels.

g. Risk of product quality degradation as the result of cost and schedule constraints imposed on the program.

h. Effectiveness of risk reduction and control measures.

i. Acceptability of residual risks.

MDA risk management strategy includes a stakeholder collaborative effort in the overall risk management process. The risk management plan shall reflect this interaction. Qualitative and quantitative risk criteria shall be mutually agreeable between affected stakeholders. The plan shall include a process for flowing risks up through the required levels (e.g., from supplier to the cognizant MDA Deputates and Elements and from Deputates and Elements to MDA Systems Engineering & Integration). The plan shall be made available to MDA Systems Engineering & Integration, the cognizant MDA Deputates and Elements, and designated representative upon request.

3.1.6 Pedigree Program

The developer, in conjunction with their suppliers shall have a formal pedigree program, which supports technical and mission assurance reviews, investigations, and critical events. Pedigree data that shall be retained and made readily available to the cognizant MDA Deputates and Elements or designated representative includes: configuration status accounting data, manufacturing and process data, test data, quality data, interface control data, and safety data.

3.1.7 Internal Evaluation Program

In addition to AS9100 requirements, the developer shall evaluate their safety, quality and mission assurance program to determine compliance with Quality, Safety, and Mission Assurance (QSMA) requirements. Planning and periodicity for evaluations shall consider program phase, critical events and milestones, known problems, and level of activity in the functional area. Evaluations shall consist of reviews of QSMA disciplines and product conformance.

Internal evaluations of QSMA disciplines shall be performed to determine adequacy and implementation of policies and procedures used to satisfy QSMA requirements. Product conformance evaluations shall be performed to review fabrication, software media generation, test, and inspection operations. Evaluations shall assess effectiveness of processes used for assuring product conformance to applicable drawings, specifications, and procedures and shall include random assessment of product conformance to applicable drawings, specifications, and procedures.

Summaries of evaluations and corrective actions taken shall be prepared and distributed to internal top level management and made available to MDA/QS, the cognizant MDA Deputates and Elements, and designated representative upon request.

3.1.8 Training and Certification Program

The developer shall establish, implement, and maintain a training and certification program to ensure sufficient program knowledge and personnel skills are developed and sustained. Developer personnel shall have necessary skills and knowledge to perform their assigned training activities.

3.1.8.1 Training

The developer shall establish and maintain a training program for personnel whose work relates to, influences, or has an effect on quality or reliability of the product as required by AS9100. Particular emphasis shall be given to new products, upgrades, and sensitive, or hazardous manufacturing processes or materials. Training needs shall be periodically assessed to determine requirements for additional training. The training program shall be evaluated on a periodic basis for consistency with, and relevance to, the organization's needs.

3.1.8.2 Certification

The developer shall establish and maintain a program for certification of personnel responsible for operation, test, inspection, or control of special processes (3.12.2.3) and equipment that require certified skills. Criteria for determination of which processes require personnel certification shall be documented. The developer shall develop and maintain a list of skills and personnel requiring certification. Certification shall include a training program and a testing procedure to ensure proficiency. Documented evidence of individual certifications shall be readily available to, and used by, the immediate supervisor in assigning personnel for specific tasks. Results of tests on which certification were granted shall be maintained. A period of certification effectivity shall be specified for each skill. Personnel not exhibiting required proficiency shall be excluded from operations involved until properly recertified. The impact to any end items produced by personnel with expired certification shall be assessed. Test, inspection, and evaluation results shall be used as indicators for recertification regardless of the established period.

3.1.9 Problem and Failure Reporting and Corrective Action System

The developer shall establish and maintain a closed loop problem and failure reporting and corrective action system. The system shall include reporting of problems and failures, investigations, analyses, and performance of actions to correct problems and failures and preclude recurrence. Developer procedures shall define the level and detail of documentation, dependent on the nature and criticality of the problem and failure. Problem and failure reporting shall include identification of items and conditions experienced. The developer shall investigate problems and failures to determine trends and the need for analysis and corrective action. As a result of the investigation, the developer shall conduct problem and failure analysis to determine root cause of the problem or failure. All problems and failures identified and resulting corrective action shall be recorded and made available to the cognizant MDA Deputates and Elements and designated representative upon request.

3.1.9.1 Corrective Action

The following actions shall be taken by the developer and documented:

a. Corrective action shall be recommended and planned, an expected completion date established, and the organization responsible for performing the corrective action identified.

b. The corrective action shall be accomplished in a timely manner.

c. Follow-up verification shall be performed to ensure completion and effectiveness of corrective action.

3.1.10 Data Exchange Programs Participation

The developer and their suppliers shall participate in both Government Industry Data Exchange Program (GIDEP) and MDA Assurance Advisory Reporting System. GIDEP alerts and MDA Assurance Advisories are received by each participant's coordinator, screened, and forwarded to the appropriate program or functional group for action. All developers shall prepare a Quality, Safety, and Mission Assurance Problem Impact Statement for each MDA Assurance Advisory containing the following:

a. MDA Assurance Advisory Report Number Reference.

b. Points of Contact for Information.

c. Element and Program affected.

d. Impact on Program.

e. Action Taken.

Impact statements shall be submitted to the cognizant MDA Deputates and Elements and MDA/QS. Developers and their suppliers shall generate new GIDEP alerts and MDA Assurance Advisories. The developer shall provide technical assistance to their suppliers who are not GIDEP participants.

3.1.11 Government Insight

Government insight is a method used to gain an understanding of developer's progress in meeting contractual requirements through observation, evaluation, and participation. Insight provides the government with real-time information on problems and issues, and provides an independent source of technical expertise. To facilitate government insight, the developer shall provide the government open access to all matters and data relating to the contract. The access shall include, but not be limited to: facilities; meetings such as program reviews, technical interchange meetings, failure review boards, and change control boards; program activities such as test events; training programs; information and analyses for any anomalies or issues occurring during fabrication, assembly, test, handling, or transportation which affect system integrity; and all data directly related to the program. The government may offer feedback to the developer for consideration. Government insight shall be extended to MDA personnel and their designated representatives. Personnel with government insight shall protect developer activities and information received or accessed from unauthorized disclosure.

3.1.12 Program Reviews

The developer shall support periodic MDA or cognizant MDA Deputates and Elements reviews to report program progress, risks, and status. Developer's support shall include hosting, participating, preparing meeting minutes, and responding to review action items.

3.1.13 MDA Evaluations

Management and work activities, operations, documentation, software, and metrics of the developer and suppliers are subject to onsite evaluation, review, survey, and inspection by MDA/QS, cognizant MDA **Deputates and Elements** representatives, and/or their designated independent assurance agency. The developer shall grant access to MDA/QS, cognizant MDA Deputates and Elements and/or their representatives to conduct planned evaluations. Resources and an acceptable work area shall be

provided to assist with the evaluation with minimal disruption to work activities. The developer shall provide documents, records, and equipment required to perform Quality, Safety, and Mission Assurance activities. The developer shall support MDA evaluations and provide timely corrective actions, as required.

3.1.14 MDA Mission Assurance Representatives

The developer shall provide MDA Mission Assurance Representatives (MAR) with documents, records, equipment, and working areas within his facilities that are required by designated government representatives to perform delegated activities. The developer shall make support services and office space available for resident MDA MARs.

3.1.15 Government Furnished Material, Equipment, or Information

The developer shall comply with AS9100 and the following when the government furnishes materials, equipment, or information:

a. Perform examination upon receipt, consistent with practicability, to detect damage resulting from transit.

b. Inspect to verify quantity, completeness, and proper identification.

c. Handle and store the material or item in a manner to guard against damage, deterioration, and disclosure.

d. Periodically inspect the stored material, item, or information to ensure adequate storage conditions and to guard against damage, deterioration, and disclosure during storage.

e. Perform required maintenance and calibration.

f. Establish controls for proper use or disposition.

The developer shall report to the government representative any government furnished material, equipment, or information that is lost, found damaged, malfunctioning, exposed to conditions which could lead to degradation, or that is otherwise unsuitable for use. In the event of damage or malfunction during or after installation, the developer shall determine and record probable cause. Individual decisions regarding particular government furnished material, equipment, or information shall be documented in the contract file.

3.1.16 Repair, Refurbishment, and Modification

The developer shall develop and document methods, procedures, and standards for performance of repair, refurbishment, and modification of returned government owned products. Standards for acceptable and unacceptable conditions shall be prepared and shall define any allowances for wear during the acquisition process of the product. Standards and procedures shall be of sufficient detail for use by repair activities and submitted to the cognizant MDA Deputates and Elements or designated representative for review and approval. Products shall be:

a. Verified as to condition and configuration.

b. Controlled to prevent commingling of serviceable and unserviceable items.

c. Assessed to determine actions required for restoring the product to an acceptable condition.

The developer shall establish processes to be applied when a reported failure cannot be confirmed upon receipt. The process shall define additional actions to be taken, including number and nature of tests necessary, retest criteria and approval, test facilities, and personnel required. After following approved

processes, non-repeatable or unverifiable failures shall be referred to the cognizant MDA Deputates and Elements or designated representative for disposition.

This Page Intentionally Left Blank

3.2 Design and Development

The developer's design and development program shall ensure required system capabilities are translated into a documented, integrated design solution; and verification is performed to ensure the solution meets requirements. The program shall ensure functional and performance requirements, and internal and external interfaces are identified, classified, achieved, and controlled. The program shall use an integrated product and process development and iterative systems engineering approach to ensure all aspects of the product's life cycle are considered during the design and development process and desired outputs are achieved to ensure mission success. The developer shall perform system engineering in accordance with EIA 632, Process For Engineering a System. Verification and validation activities associated with the system engineering processes shall be performed in accordance with documented plans and procedures. Verification and validation results shall be documented and retained. Drawings and specifications shall be generated to document the design solution and be controlled in accordance with 3.10. The design and development program shall be executed in accordance with policy supported by controlled engineering manuals, procedures, and guidelines that implement fundamental design principles, practices, and processes. The developer shall ensure systematic implementation of design principles and practices through the use of proven industry design standards and best practices. The developer shall establish and maintain plans to manage and control design and development project activities. Requirements of this section apply to all new designs, redesigns, block changes, and modifications.

3.2.1 Integrated Product and Process Development

The developer shall establish and maintain a process that integrates all design and development activities, through the use of multidisciplinary teams, to concurrently balance the product design and its associated fabrication, manufacturing, and supportability processes to achieve life cycle system cost and performance objectives. The multi-disciplined Integrated Product Teams (IPT) shall represent all necessary specialties, functions, disciplines, and allow for participation of cognizant MDA Deputates and Elements, including MDA designated technical representatives. Cognizant MDA Deputates and Elements, and designated technical representatives, operate as an integral part of the selected IPT to independently assess performance and identify and resolve potential design problems.

3.2.2 Peer Reviews

The developer team shall conduct engineering working-level reviews (peer reviews) throughout design and development to identify and resolve technical issues and concerns prior to formal system level reviews (3.4). Engineering peer reviews for hardware and software are required during all phases of the program life cycle as a key component of the developer's mission assurance program. These reviews are expected to present more detail than system level formal reviews. Peer review shall be conducted informally at the subsystem and lower levels by independent technical experts having current detailed knowledge of the design specialties associated with the item under review. The purpose of peer reviews is to substantiate a detailed understanding of the design's ability to meet all of its performance and interface requirements, to surface correctable problems early, identify risks, and to ensure best known practices are used to enhance design robustness by avoiding known or predictable problems.

3.2.3 Technical Performance Measurement

The developer shall establish and maintain a process that provides a method of measuring Technical Performance Measurements (TPM). TPMs are derived from system requirements to provide a cross section or representative sample of the measures that define key system performance or high program risk. Each TPM shall be reviewed quarterly to determine their continued relevance and/or effectiveness. Changes (additions/deletions) in TPMs shall be coordinated with MDA Systems Engineering & Integration and the cognizant MDA Deputates and Elements. Metrics shall be established to provide visibility into actual versus planned performance for TPMs. TPMs trend data shall be evaluated and the results shall be included in the risk management program (3.1.5).

3.2.4 Systems Engineering for Design

The system engineering process shall be used to translate mission and operational requirements and objectives, functional and performance requirements, design constraints, interface and interoperability requirements, statutory and regulatory requirements, and other applicable input requirements into an integrated design solution through concurrent consideration of all life cycle needs. When designing product, the developer shall comply with the systems engineering process as defined in ANSI/EIA-632.

3.2.4.1 Systems Engineering Management Plan

The developer shall document design, engineering, and technical management disciplines and processes that support all phases of the life cycle in a Systems Engineering Management Plan (SEMP). SEMP shall be submitted to MDA Systems Engineering & Integration for information and to the cognizant MDA Deputates and Elements for approval.

3.2.5 Design for Interoperability

The developer shall establish and maintain a design engineering process that ensures interoperability with other MDA systems. The developer shall coordinate with MDA Systems Engineering & Integration and the cognizant MDA Deputates and Elements to establish, document, and control the interface requirements necessary to ensure interoperability with other affected MDA systems. Interoperability requirements identified during the systems engineering process shall be incorporated into the design's interface control documentation (3.9.2) and evolve consistent with the evolutionary acquisition approach. Interoperability requirements for a design shall be specified at a level of detail that allows for verification and test.

3.2.6 Design for Producibility

The developer shall establish and maintain a design engineering process that makes producibility of the design a priority early in the design and development effort. The developer shall concurrently develop product designs and the required manufacturing processes to be used during fabrication and production. The manufacturing processes selected shall be statistically capable and have adequate capacity to meet expected production rate. The product shall be designed in such a manner that fabrication and manufacturing methods and processes have flexibility in producing the product at a reasonable cost while maintaining required functionality, performance, quality, and reliability. As part of the design for producibility, developer shall identify and document key characteristics as defined in AS9100. Trade studies shall be performed to balance cost, time, simplicity, standardization, and other producibility aspects and still satisfy performance requirements of the product. Problem areas shall be identified early in the design process to ensure product designs, which:

a. Minimize variability in the manufacturing process.

b. Achieve higher quality within cost and schedule.

c. Allow for insertion of new technologies to achieve increased producibility.

d. Increase systems reliability.

3.2.7 Design for Testability

The developer shall design for testability. The testability program shall define the functional test parameters and the most efficient method and point at which the item will be tested. The testability program shall accomplish the following:

a. Establish diagnostic concept and testability requirements for built-in (Built-In Test/Built-In Test Equipment (BIT/BITE)) and off-line test performance.

b. Integrate testability into requirements and systems during the design process in coordination with the Integrated Test and Evaluation Program (3.7) and maintainability design process.

c. Evaluate the extent to which the design meets testability requirements.

d. Include testability in the design review process.

3.2.8 Design for Supportability

Supportability analyses shall be an integral part of the systems engineering process to ensure the product designed and developed meets the government's planned logistics support approach. During the initial stages of the design and development process the developer shall coordinate with the cognizant MDA Deputates and Elements to identify and document a logistics support strategy for the product.

3.2.9 Design for Commercial and Non-Developmental Items

The developer shall establish and maintain a system to control the design selection, evaluation, acceptance, and support of Commercial-Off-The-Shelf (COTS) and Non-Developmental Items (NDI), including both hardware and software. If COTS/NDI products are used in MDA systems, subsystems, or assemblies, the developer shall ensure COTS/NDI items meet all functional and interface requirements. COTS/NDI shall be selected, and qualified (3.6 and 3.7.3) to operate in the intended application. The developer shall verify the COTS/NDI meet or exceed performance, quality, reliability, environmental, and survivability requirements, and develop a strategy for supporting or upgrading the products throughout the system life cycle.

3.2.9.1 COTS/NDI Design Strategies

If COTS/NDI are used, as a minimum, the design strategies shall:

a. Use form, fit, and function requirements to query the market.

b. Begin market analysis early in program planning. Market analysis shall consider the stability of the market for each item and projected technology advances, including the quality, stability, and quantity of suppliers who provide products for each commercial and non-developmental item.

c. Assess the availability, relevance, and adequacy of: design documentation; reliability data, performance data, and quality data; or, if needed, develop a mitigation plan to account for the lack of data.

d. Design systems to withstand insertion of new technology. When selecting COTS/NDI for inclusion in the design, the developer shall include consideration of hardware and software support as follows:

   1) *Hardware*: Refresh cycle; availability and capability of vendor-supported repair or alternate repair support; warranty cost and coverage; Total Ownership Cost (TOC); vendor technical and design support; sole source or multi-vendor availability; remaining program life; and availability of technical data package purchase rights.

   2) *Software:* Vendor technical and design support or alternate support; revision schedule; remaining program life; cost of licenses and upgrades; TOC; stability of product; projected revisions; problem history; any known software issues or defects; and availability of technical data package purchase rights.

e. Use an open-system architecture with strict adherence to commercial standard interfaces for hardware and software.

f. Assure the strategy considers mission and environmental requirements and margin.

g. Develop a procurement strategy for determining COTS/NDI viability for specific systems.

h. Require extensive compatibility testing of the product at both subassembly and system levels.

i. Functionally test at the system or subsystem level COTS/NDI spares using operational software.

j. Produce vendor item control drawings, controlled in accordance with 3.10, documenting the engineering description and acceptance criteria for COTS/NDI. The vendor item control drawing shall provide a suggested source(s) of supply, the vendor's item identification, and sufficient engineering definition for acceptance of interchangeable items within specified limits.

k. Define a COTS spare policy for times when licenses and warranties expire before product spares are used.

3.2.10 Requirements Traceability and Verification Matrix

The developer shall establish and maintain a system for providing traceability to ensure that hardware and software specification and interface requirements are implemented in the design, including any COTS/NDI used, and verified. Each requirement contained in system specifications, subsystem specifications, equipment specifications, software requirements specifications, interface control documents, coordination drawings, and any other documents containing technical requirements shall be traceable to the demonstration, analysis, test, or inspection document in which requirements are verified. Bi-directional traceability shall be established from the source requirement down to its implementation level requirements and from the implementation level requirements back to the source.

The developer shall create and maintain a requirements traceability and verification matrix. For each requirement, the requirement traceability and verification matrix identifies the method of verification (analysis, inspection, demonstration or test). The requirements traceability and verification matrix shall be controlled to ensure that emerging requirements are documented and have performance verification methods assigned.

3.2.11 System Design Verification and Validation

The developer shall perform verification and validation in compliance with ANSI/EIA-632 and AS9100 as supplemented by the following requirements.

The developer shall perform system design verification at appropriate stages of development to assure that the design output meets the design input requirements. Design verification shall verify that:

a. Design element descriptions are traceable to requirements of the functional architecture.

b. Requirements of the functional architecture are allocated and traceable to the design architecture.

All internal and external design interfaces shall be upward and downward traceable to their source requirement. The requirements traceability and verification matrix (3.2.10) shall be used to trace verification methods to requirements of the functional architectural and requirements baseline.

The developer shall perform system design validation to demonstrate that mission capabilities are met. It may be necessary to perform validation incrementally prior to completion of the final product. Validation activities include, as appropriate: test, simulation, demonstration, or other applicable methods. The developer shall perform system design validation of product against its requirements baseline established during requirements analysis. System design validation shall consist of the following activities:

a. Evaluation of product against its requirements baseline to ensure that it represents identified MDA expectations and project, developer, and external constraints.

b. Technical assessment of product against its requirements baseline to determine whether the full spectrum of possible system operations and system life-cycle support concepts has been adequately addressed.

Design verification and validation documentation shall consist of data, results, and reports from: tests, inspections, demonstrations, calculations, analyses, and other relevant verification and validation activities. Design verification and validation testing activities shall be planned, controlled, reviewed, and documented to assure tests are performed in accordance with specifications and the requirements pertaining to test plans (3.7.9) and test procedures (3.7.10).

3.2.12 Safety and Environmental Requirements

The developer shall design a system that complies with safety and environmental statutes, regulations, policies, agreements, and provision 3.14. The selection and use of energetic materials and the design of munitions and other explosive components, materials, or systems shall comply with Department of Defense explosives safety requirements.

3.2.13 Open Systems Design and Standards

The developer shall have an open systems design strategy that maximizes opportunities for reuse of existing technologies, previously designed product, and to facilitate product upgrades. Open systems architectures and design standards shall ensure interoperability and compatibility in the system and product designs. Open system designs and standards shall be selected and controlled through the systems engineering process.

3.2.14 Modeling and Simulation

The developer shall establish and maintain a system for selection, control, verification, validation, and accreditation of modeling and simulation techniques, and tools that are used for design and development activities. Modeling and simulation may be applied to support design and development activities and provide capabilities such as: evaluating requirements; performing sensitivity and trade-off studies; performing reliability predictions; supporting design decisions; understanding and demonstrating system capabilities and performance; and exercising systems under test.

Verification is the process of determining that a model or simulation implementation accurately represents the conceptual description and specifications. Validation is the process of determining the degree to which a model or simulation is an accurate representation of the real world from the perspective of the intended uses. Accreditation is the formal certification that a model or simulation is acceptable for use for a specific purpose.

3.2.15 Classification of Characteristics

The developer shall establish and maintain a system to analyze the design to identify and classify characteristics of the product, which could affect Coordination, Life, Interchangeability, Function, and Safety. Characteristics that must be controlled, maintained, and appraised to ensure design integrity, and are deemed essential for government acceptance requirements, shall be identified and classified in design disclosure documentation and applicable technical documentation.

3.2.15.1 Classification of Characteristics Levels

As a minimum, the following classification levels (critical, major, or minor) shall be used:

a. *Critical*: A critical characteristic is one that analysis indicates is likely, if defective, to create or increase a hazard to human safety, or to result in failure of a weapon system or major system to perform a required mission.

b. *Major*: A major characteristic is one that analysis indicates is not critical but is likely, if defective, to result in failure of an end item to perform a required mission.

c. *Minor*: A minor characteristic is one that analysis indicates is significant to product quality but is not likely, if defective, to impair the mission performance of the item.

The developer shall:

a. Develop a policy that delineates criteria for determining the level of test, inspection, or control to be applied to each classification level (critical, major, or minor).

b. Base classification solely on the impact to the product if the characteristic is not within specified limits and not on the magnitude of the characteristic's tolerance.

c. Identify critical and major characteristics on the drawing(s) or specification(s) or via an alternate method approved by the cognizant MDA Deputates and Elements.

d. Complete prior to establishing each successive baseline for the system or product.

3.2.16 Electromagnetic Environmental Effects Design & Verification

The developer shall ensure the system is designed to be electromagnetically compatible among all subsystems, ordnance, equipment, and parts within the system and to be survivable and compatible with environments caused by electromagnetic effects external to the system. Verification of compliance with electromagnetic environmental effects operational and design requirements shall be accomplished in accordance with MIL-STD-464 requirements through the application of test, analysis, or a combination thereof. Verification shall also address all acquisition process aspects of the system, including normal in-service operation, maintenance, aging checkout, storage, transportation, handling, packaging, loading/unloading, launch, and other applicable operations. Electromagnetic environmental effects shall encompass all applicable electromagnetic disciplines, such as, electromagnetic compatibility; electromagnetic interference; electromagnetic pulse; hazards of electromagnetic radiation to ordnance, fuel, and personnel; electrostatic discharge; and DC magnetics.

3.2.17 Space Radiation, Nuclear Hardness and Survivability Program

When vulnerability requirements require the weapon system to survive radiation environment, the developer shall establish and maintain a hardness assurance, maintenance, and surveillance program that addresses all phases of the program. The program shall enable the system to operate in hostile environments, natural space radiation environments, and other nuclear radiation environments expected to be encountered during performance of a mission.

a. Hardness maintenance shall ensure that Hardness Critical Items (HCI) are identified and tracked throughout the acquisition process and any changes in fabrication and production processes and materials are evaluated to ensure no radiation hardness degradation has occurred.

b. Hardness assurance shall ensure that developers, suppliers, and Department of Energy laboratories implement the hardness critical test, inspection, and processes during the assembly, production, maintenance, storage, and shipping of HCIs (3.6.12.7).

c. Hardness surveillance shall identify and perform periodic test and inspection of HCIs.

The program shall be coordinated with systems engineering and the Parts and Materials Control Board (3.6.2).

3.2.18 Transition to Operations or Production

The developer shall plan and design for transition to operations or production throughout the development process. Planning for transition shall begin early in the system development and demonstration phase. Planning shall ensure production disciplines and processes required for operations or production is developed concurrently with the product's design.

The developer shall utilize the risk management process (3.1.5) to minimize risks associated with transitioning designs to operations or production. The developer shall perform incremental risk assessments to identify and mitigate transition risk to support fielding of new or upgraded designs and production milestone decisions. The transition risk assessment shall include transition risks identified at mission critical suppliers. The developer shall identify and report any remaining transition risks during Preliminary Design Reviews (3.4.1.5), Critical Design Reviews (3.4.1.6), and System Verification Reviews (3.4.1.8). The risks associated with transitioning shall be effectively communicated to the cognizant MDA Deputates and Elements, and MDA Systems Engineering & Integration.

3.2.18.1 Transition to Operations or Production Plan

The developer shall establish and maintain a Transition to Operations or Production Plan, which defines the approach for supporting of operations or production decisions. The plan shall address transition activities using the critical path templates defined in the Technical Risk Identification and Mitigation System (TRIMS). The Transition to Operations or Production Plan shall be developed concurrently with product design and be made available to the cognizant MDA Deputates and Elements, MDA Systems Engineering & Integration, and designated representative upon request.

This Page Intentionally Left Blank

3.3 Software and Firmware

The developer shall establish and maintain a system to implement software and firmware requirements as specified herein.

These requirements are mandatory for all deliverable safety and mission critical software and firmware, all software or firmware used to accept safety and mission critical deliverable items, and software used for modeling, simulating, and predicting safety critical and mission critical deliverable items.

Additionally, these requirements apply to programmable logic controllers, firmware such as erasable programmable read only memory, Commercial-Off-The-Shelf (COTS) and Government-Off-The-Shelf (GOTS) products, auto generated code, and reused code that is safety critical, mission critical or that has a direct impact on, or association with safety critical hardware or function.

3.3.1 Management Processes

The developer shall implement software management and infrastructure processes that are in accordance with AS9100, provision 3.1, and the following requirements. Developer's program manager shall ensure that a software project manager is designated for each software project. The developer's software project manager shall ensure that assigned projects are planned, tracked, and controlled. Additionally, project managers shall be responsible for negotiating commitments; developing, documenting and implementing the software development plan, software projects activities, products, and results.

3.3.1.1 Intergroup Coordination

Software engineering shall participate with other engineering groups, safety, end users, and MDA in establishing system requirements, performing requirements allocation, and making trade offs. This includes defining critical characteristics of the product, negotiating dependencies, and documenting acceptance criteria. Software engineering shall work with other engineering groups to:

a. Monitor and coordinate technical activities and resolve technical issues.

b. Identify, negotiate, and track critical dependencies.

The developer shall develop and maintain a process to handle issues that cannot be resolved by group participants. In addition, the developer's processes shall set forth the rules for ensuring that support tools used by different engineering groups are compatible.

3.3.1.2 Software Development Planning

The developer shall use software allocated requirements as the basis for planning software design, development, test, and activities. The planning shall be documented in a Software Development Plan (SDP).

The plan shall define the project's model, tasks, activities, and products. For each activity the plan shall identify entry and exit criteria for each development activity. In addition, the plan shall identify the software products that are required and the controls that are to be implemented over the products. Methods for tracking activities, tasks, and progress shall be defined in the SDP.

As a minimum, the SDP shall include the following:

a. Software estimations for cost, schedule, effort, and resources.

b. Project organizational structure, authority and responsibility of each organizational unit, including external interfaces.

c. Work breakdown structure.

d. Software products that undergo review or inspection.

e. Methods and tools to be used for design and development, requirements analysis; software safety; coding; verification; validation; testing; configuration management; and software assurance.

f. Engineering environment (for development, operation, or maintenance, as applicable), including test environment, library, equipment, facilities, standards, procedures, and tools.

g. Identify all proposed subcontracts, GFE, GFI, COTS, third party and NDI software; and address how the software is to be used and its applicable controls.

The SDP shall define the verification method used to ensure that development environment is available to software developers and other users prior to start of each development phase; development team experience or training in applying the tools and methods; and configuration control over tools.

The SDP shall include the proposed verification activities for all safety, quality, functional, and performance requirements; including the approach for interfacing with the independent verification and validation agent, if specified. Verification activities include various techniques such as reviews, inspections, tests, walk-throughs, desk checking, and many types of analysis such as traceability analysis, formal proof, or fault tree analysis.

The developer's SDP shall include or reference process documentation addressing:

a. Software models approved for use.

b. Guidelines and criteria for tailoring the organization's standard software processes.

c. Requirements and guidelines for establishing and maintaining software process databases.

d. A library and guidelines for software process related documentation (e.g., procedures, manuals, project plans, standards).

The developer shall verify that:

a. Project planning requirements are adequate and timely.

b. Processes selected for the project are adequate, implemented, being executed as planned, and compliant with the contract.

c. The standards, procedures, and environments for the project's processes are adequate.

d. The project is staffed and personnel trained.

The SDP and all updates shall be submitted to the cognizant MDA Deputates and Elements for approval. The plan shall be maintained consistent with current project requirements. The software developer shall maintain records of all software planning and re-planning efforts and data.

3.3.1.3 Estimation

The developer shall establish and maintain a system for estimating software projects and products. The system shall address:

a. Software scheduling efforts, including critical dependencies and critical paths.

b. Software product size and complexity.

c. Software effort and costs.

d. Critical computer resources (e.g., input/output, buffer, memory).

The developer's estimations shall be used as inputs to the planning process. Records of estimations shall be maintained for future use and reference.

3.3.1.4 Software Risk Management

The developer's software risk management process shall be in accordance with 3.1.5 and include the following:

a. Probabilities associated with all medium, serious, and high risks.

b. Identification of consequences associated with all medium, serious, and high risks.

c. Mitigation plan for all medium, serious, and high risks.

d. Tracking of risks throughout the development, test, operations, and maintenance efforts.

e. Corrective actions for deviations of more than 10 percent from the mitigation plan.

f. Watch list for all low risks, which shall be periodically re-assessed.

3.3.1.5 Process Improvement

The developer's top level management shall define and oversee the organization's program for implementing process development, assessment, and continuous process improvement. Management shall monitor and control effectiveness of processes used during development, maintenance and operation of software, including relevant processes corresponding to services supporting other organizational entities. The program shall address, as a minimum:

a. Increasing software quality and productivity.

b. Decreasing software development cycle time.

c. Adopting new technologies and processes.

d. Developing software processes and related process assets.

e. Coordinating software process development, assessment, and improvement across the organization.

f. Ensuring software safety (3.14).

As part of the process improvement program the developer shall establish and maintain a system for performing quantitative process management activities. Measurement data shall be collected, analyzed and the software processes brought under quantitative control. Results of

the activities shall be documented and distributed to affected organizations. Quantitative measurement activities shall be used to establish process capability baselines.

The developer shall establish and maintain a software improvement process for developing, assessing, measuring, controlling, and improving software processes. It shall address, as a minimum, the following:

a. A suite of organizational processes for all software processes as they apply to its development and maintenance activities. The processes and their application to specific cases shall be documented in organization's documentation. A process control mechanism shall be established to develop, monitor, control, and improve processes.

b. Process assessments and required records.

c. Improvements to its processes as a result of process assessment and review. Process documentation shall be updated to reflect improvement in the organizational processes.

d. Standards for documenting software processes and improvements.

e. Coordination of software process databases.

f. Monitoring, evaluating, introducing, or transferring new processes, methods, and tools into the organization.

g. Collecting, analyzing, maintaining, and using software quality data to support process assessments and improvements.

h. Communication of software process development, maintenance, and improvement activities within the organization is preformed.

3.3.1.6 Technology Change Management

The developer's top level management shall define the organization's policy for improving the software quality, productivity, and cycle development time. This includes establishing the responsibilities and authority for implementing the policy, allocating resources for technology change management activities, and coordinating requirements and issues associated with technology change management at the appropriate management levels within the organization. The developer shall establish a program, which identifies, selects, evaluates new technologies, and incorporates technologies that improve software quality, increase productivity, and decrease development cycle time.

3.3.1.7 Software Supplier Management

The developer shall use the supplier management program for the selection of software suppliers and management of software subcontracts. The requirements of this section supplement those of AS9100 and 3.13, Supplier Management. The developer shall inform the cognizant MDA Deputates and Elements of all mission or safety critical software supplier selection decisions prior to award of the contract.

3.3.1.7.1 Selection of Software Suppliers

Selection of software suppliers shall be based on an evaluation of the supplier's capability to develop and maintain software. Selection of software suppliers shall also comply with requirements of 3.13. Evaluation criteria shall include prior performance, software engineering and management capabilities, available personnel, available resources, and a documented and implemented quality system. The developer shall assess the supplier's software engineering,

software quality, software configuration management and software management capabilities and maturity.

3.3.1.7.2 Flowdown of Requirements

The developer shall flow down requirements in accordance with 3.13.4, Supplier Program Requirements. The developer shall require the supplier to obtain developer approval prior to subcontracting any of its software development work. Supplier's planning documentation shall be made available to the cognizant MDA Deputates and Elements and designated representative upon request.

3.3.1.7.3 Software Supplier Monitoring

The developer shall monitor and control the suppliers in accordance with their flow down requirements, including metrics, procedures, and software planning. Monitoring results shall be collected, analyzed, documented, and distributed to program management, project management, and mission assurance organizations. Monitoring results shall be used as a factor to determine supplier ratings (3.13.2). Departures from plans, procedures, or flow down requirements shall be reviewed with the developer and corrective action taken as directed.

3.3.1.7.4 Acceptance of Supplier Software Products

Supplier product acceptance shall be accomplished in accordance with approved plans and procedures addressing quality, safety, functional, performance, load, interoperability, stress, and testing requirements. Acceptance shall also include review and approval of all operational and maintenance documentation.

3.3.1.8 Software Training

The developer's training program (3.1.8) shall include training for software personnel. The program shall include a method or procedure by which software training needs are identified, provided, and assessed. Training for the organization and projects software processes shall be coordinated across the organization. The software training program shall include the following, as a minimum:

a. A review of the project requirements to establish and make timely provision for acquiring or developing the resources and skills required by the management and technical staff. The results of this review shall be documented in software training plans. The types and levels of training and categories of personnel needing training shall be determined. A training plan addressing implementation schedules, resource requirements, and training needs, shall be developed and documented. Training records shall be maintained.

b. Development of training manuals, including presentation materials used in providing training.

c. Training plans that identify the group or organization responsible for fulfilling training needs. In addition, the developer shall develop training standards and procedures defining how software training courses are to be selected, developed, and maintained.

d. Identification of the training subject, which includes specific tools, techniques, methodologies, and computer resources to be used in the development, maintenance and management of the software product.

The developer's management shall ensure the right composition and categories of appropriately trained personnel are available for planned software activities and tasks.

3.3.1.8.1 Software Personnel Training

The developer's management shall ensure software managers, software engineers, software quality assurance personnel, configuration management personnel, and other related groups are adequately trained to implement the tasks and activities required for their work.

Personnel performing Software Quality Assurance (SQA) and Software Configuration Management (SCM) functions shall be trained to perform their activities, while other members of the software project shall receive orientation on their roles, responsibilities, and value of SQA.

Software managers, software engineers, and other individuals participating in project planning shall be trained in the software estimating and planning procedures applicable to their area of responsibility. Software managers shall be trained in managing the technical, administrative, and personnel aspects of a software project.

First-line managers shall receive orientation in the technical aspects of the software project. Individuals responsible for developing the project's software processes shall receive required training in how to tailor the organization's standard software processes.

3.3.2 Software Development, Maintenance, and Operational Processes

The software developer shall establish and maintain plans for performing development, maintenance, and operational processes and activities. The plans shall include specific standards, methods, tools, actions, and responsibility associated with the development, qualification, and maintenance of all requirements including safety and security. If necessary, separate plans may be developed.

Non-deliverable items may be employed in the development or maintenance of the software product. However, the software developer shall ensure that the operation and maintenance of the deliverable software product after its delivery are independent of such items, otherwise those items shall be considered as deliverable.

3.3.2.1 Requirements

The developer's system requirements analysis, system architectural design, and software requirements analysis activities shall be performed in accordance with the following:

a. The developer shall establish and maintain the requirements allocated to software. The requirements shall be reviewed to ensure that they are complete, feasible, clearly and properly stated, consistent with each other, and verifiable. Problems noted with the allocated requirements shall be documented and resolved with the responsible parties. Bi-directional traceability shall be established from the source requirement down to its implementation level requirements and from the implementation level requirements back to the source. Changes to allocated requirements shall be reviewed, approved, and incorporated into the project in accordance with documented processes and procedures.

b. The developer shall flag or uniquely identify software mission or safety critical item requirements and characteristics affecting coordination, life, interface, function, and safety requirements (3.2.15).

c. The developer shall select and use operating systems, standard languages, architectures, and tools, which provide for open systems to the maximum extent practical.

The system and software requirements shall be evaluated to assess the following, as a minimum:

a. System requirements are consistent, feasible, verifiable, and testable.

b. System requirements have been appropriately allocated to hardware items, software items, and manual operations according to design criteria.

c. Software requirements are traceable, consistent, feasible, testable, verifiable, and accurately reflect system requirements.

d. Software requirements related to security, and safety and mission criticality are correct.

e. Appropriateness of programming language features, constructs and limitations, design standards, and methods used.

f. Feasibility of the software items fulfilling their allocated requirements.

The results of the evaluation shall be documented and maintained as a quality record.

3.3.2.1.1 Software Reuse

The developer shall evaluate reusable software products that meet specific MDA program requirements and are cost-effective over the life of the system. Reused software includes previously developed software, which is used for project development as is or with adaptation. This includes COTS software, and software supplied by the government (i.e., GFI and NDI).

The developer's analyses of existing software shall be carried out and finalized at the architectural design stage. The developer shall provide evidence of the product's suitability. This includes an assessment of the relationship between the software's original intended environment and proposed environment and the impact of any differences on the software, and analyzing known defects for their impact on proposed operational environment and mission requirements. When analysis of available data indicates risks, the developer shall propose and obtain agreement from cognizant MDA **Deputates and Elements** on the additional verification tasks to be performed. The basis for reuse decisions shall be documented and maintained.

The developer shall make software reuse documentation available to the cognizant MDA **Deputates and Elements** for review prior to its incorporation into the product baseline. Reused software shall be subject to the same requirements as newly developed software.

3.3.2.2 Software Design

The developer shall ensure software design is developed, maintained, documented, and verified. The software design shall be traceable to the software requirements and form the framework for coding. Software design products shall be maintained consistent with software requirements, software code, and project requirements. The software developer's design activities shall include the following, as a minimum:

a. Definition and documentation of test requirements and schedule for testing software units. The test requirements shall include stressing the software unit.

b. Development and delivery of operation and maintenance documentation. Preliminary and/or draft versions of the documentation shall be made available at the critical design review for review and comment by end users, the cognizant MDA Deputates and Elements, maintainers, and software quality. Documentation shall be updated based upon feedback from reviewers. Software developer's final operation and maintenance documentation presented for government acceptance shall:

1) Define delivered capabilities and limitations.

2) Be proofed or qualified during software qualification testing.

The software developer shall evaluate software design and test requirements to assess the following, as a minimum:

a. Design implements proper sequence of events, inputs, outputs, interfaces, logic flow, allocation of timing and sizing budgets, and error definition, isolation, and recovery.

b. Selected design can be derived from requirements.

c. Design implements security, and safety and mission critical requirements correctly.

d. External consistency with architectural design.

e. Internal consistency between software components and software units.

f. Appropriateness of design methods and standards used.

g. Traceability to the requirements of the software item.

h. Feasibility of testing.

i. Feasibility of operation and maintenance.

The evaluation results shall be documented and maintained as a quality record.

3.3.2.3 Software Code

The software developer shall ensure software code is developed, maintained, documented, and verified. The code shall be traceable to the software design. The software code shall be documented and include comments in accordance with coding standards. Use of language features shall be in accordance with specific guidelines and safety limitations. There shall be no undocumented features in the code. Software code shall be maintained consistent with the software design and project requirements.

The software developer shall evaluate software code to assess the following, as a minimum:

a. Code is traceable to design and requirements, testable/verifiable, correct, and compliant with requirements and coding standards.

b. Code implements proper event sequence; consistent interfaces; correct data and control flow; appropriate allocation timing and sizing budgets; and error definition, isolation, and recovery.

c. Selected code can be derived from design or requirements.

d. Code implements safety, security, and other critical requirements correctly.

e. External consistency with the requirements and design of the software item.

f. Internal consistency between unit requirements.

g. Test coverage of units.

h. Appropriateness of programming language features, constructs and limitations, coding methods, and standards used.

i. Feasibility of software integration and testing.

j. Feasibility of operation and maintenance.

The evaluation results shall be documented and maintained as a quality record.

3.3.2.4 Software Unit Testing

The software developer shall perform unit testing to demonstrate software design has been successfully implemented in the software code. Unit test criteria (inputs, expected results, evaluation and acceptance criteria) shall be developed to specify the types of test cases that are to be executed. Test cases shall cover the unit's design.

Unit testing for mission critical or safety critical software shall be performed to:

a. Detect errors in translation of design requirements into code prior to integration with other computer software units.

b. Detect errors in algorithms and logic used to implement software requirements and design specifications.

c. Verify that each computer software unit fully satisfies applicable software requirements specifications and design specifications.

d. Detect and eliminate all unused, unreachable, or unexecutable code.

The results of unit testing shall be documented and maintained. Defects shall be recorded and tracked to closure or certify that software code has been compiled error free and successfully passed all unit tests.

The software developer shall evaluate software unit test results to assess the following, as a minimum:

a. Traceability to the requirements and design of the software item.

b. External consistency with the requirements and design of the software item.

c. Internal consistency between unit requirements.

d. Test coverage of units.

e. Appropriateness of coding methods and standards used.

f. Feasibility of software integration and testing.

g. Feasibility of operation and maintenance.

The evaluation results shall be documented and maintained as a quality record.

3.3.2.5 Software Integration

The software developer shall plan and perform integration testing. Integration testing shall verify the software code implements design and interface requirements specified in design documentation at unit,

component, and Software Configuration Item (SCI) level. Test cases shall cover SCI architectural design. When applicable, software integration testing shall assure adequacy of human-machine interfaces. All problems or issues identified during integration testing shall be documented and tracked to resolution. Problems or issues associated with software safety shall be reported in accordance with provision 3.14. Results of software integration testing shall be collected, reported, and maintained.

The software developer shall evaluate the integration plan, design, code, tests, test results, and user documentation to assess the following, as a minimum:

a. Software components and units of each software item have been completely and correctly integrated into the software item.

b. Traceability to the system requirements.

c. External consistency with the system requirements.

d. Internal consistency between code and documentation.

e. Test coverage of the requirements of the software item.

f. Appropriateness of test standards and methods used.

g. Conformance to expected results.

h. Feasibility of software qualification testing.

i. Feasibility of operation and maintenance.

The evaluation results shall be documented and maintained as a quality record.

3.3.2.6 Software Qualification

The developer shall plan and perform software qualification. Software qualification shall validate the Software Configuration Item (SCI) and integration of SCIs to ensure they meet the allocated software requirements. Software qualification cases and procedures shall be planned, prepared, and executed by personnel independent from those responsible for the item's design and implementation (code). Test cases shall be traceable to individual software requirements. Software qualification shall be performed against baselined software and baselined documentation for the allocated software requirements. Results of software qualification shall be collected, analyzed, reported, and maintained. All problems or issues identified during software qualification shall be documented and tracked to resolution. Safety problems, issues, or deficiencies shall be clearly identified and reported in accordance with provision 3.14.

Software qualifications shall:

a. Ensure implementation of each software requirement is tested for compliance.

b. Be performed at the highest level of integration practicable, utilizing intended system hardware, and actual operating conditions to the highest degree practicable. Qualification testing shall demonstrate all interfaces, and verify user documentation is complete and correct.

c. Consist of functional, performance, load, stress, and fault testing.

d. Demonstrate any associated human-machine interfaces are complete and correct.

The software developer shall make available its testing facilities, application software, and support tools for independent government testing once its software qualification is complete. The software developer and the cognizant MDA Deputates and Elements shall concur on mutually agreeable resource schedule.

3.3.2.6.1 Software Qualification Test Report

The software developer shall prepare a software test report upon completion of the qualification testing. Representatives from the software developer, software quality, and maintenance organization shall sign the report. The report shall certify conformance to the procedures and state the conclusion concerning the test result for the software product under test (accepted, conditionally accepted, rejected). All safety problems, issues or deficiencies shall be clearly identified and reported in accordance with safety provision requirements (3.14). The software developer shall make software qualification test report available to the cognizant MDA Deputates and Elements and MDA Responsible Test Organization (RTO) upon request.

3.3.2.7 Regression Tests

The software developer shall develop a method for verifying changes to software once the software has successfully completed unit test. This method shall ensure that regression tests are performed on software to verify that changes have been successfully implemented, errors have not been introduced, and software complies with specified requirements. Criteria for determining the extent of regression testing required shall be developed and made available to the cognizant MDA Deputates and Elements and designated representative upon request. Regression test suites or tests shall be made available to the cognizant MDA **Deputates and Elements** upon request. The method or process shall also ensure that only approved changes have been implemented into the code.

3.3.2.8 Software Test Program Status Reports

The software developer shall prepare and distribute software test program status reports to project and program management on a monthly basis for information and action. Information from these reports shall be used as inputs to the metrics (3.1.3) and risk management (3.1.5) programs. These reports shall include, as a minimum:

a. A description of significant problems, corrective actions taken, and schedules for accomplishment of planned actions.

b. Updates of test schedules.

c. A list of all tests planned and completed during the report period with an indication as to whether the test objective was met.

d. A description and status of all defects, problems and failures that occurred during the reporting period. Defects, problems and failures associated with safety and mission critical items shall be highlighted.

e. Status of previous failures, which remain open, and a description of corrective actions taken on failures closed during the reporting period.

These test status reports shall be made available to the cognizant MDA Deputates and Elements and designated representative upon request.

3.3.2.9 System Integration

The developer shall integrate the software configuration items, with hardware configuration items, manual operations, and other systems as necessary, into the system. The aggregates

shall be tested, as they are developed, against their requirements. The integration test results shall be documented, collected, analyzed, reported, and maintained. The developer shall ensure that the integrated system is ready for system qualification testing.

The developer shall evaluate the integrated system to assure the following, as a minimum:

a. Hardware items, software items, and manual operations of the system have been completely and correctly integrated into the system.

b. Integration tasks have been performed in accordance with an integration plan.

c. Test coverage of system requirements.

d. Appropriateness of test methods and standards used.

e. Conformance to expected results.

f. Feasibility of system qualification testing.

g. Feasibility of operation and maintenance.

The evaluation results shall be documented and maintained as a quality record.

3.3.2.10 System Qualification

The software developer shall support qualification of computing systems as part of the overall system qualification (3.7.3).

3.3.2.11 Software Installation

The developer shall establish and maintain a plan for the installation of software product and upgrades in the target environment. The resources and information necessary to install the software product shall be determined and made available. The software developer shall develop the procedure for installing the software product in the target environment. The software developer shall assist the government with installation activities as required. Where the installed software product is replacing an existing system, the software developer shall support any parallel running activities that are required.

When required the software developer shall perform installation in accordance with the installation procedure. All deficiencies, problems or issues associated with performance, capabilities, limitations, or the installation process shall be documented, reported, tracked, and resolved per 3.1.9.

3.3.2.12 Software Acceptance

At the time of software acceptance the developer shall provide objective evidence that:

a. Deliverable software complies with the contractual requirements (including any specified content of the software acceptance data package).

b. Deliverable products are complete and contain the proper versions.

c. Executable code was regenerated from configuration managed source code components and installed in accordance with predefined procedures on the target environment.

d. Approved changes have been implemented and verified.

e. All discrepancies, nonconformances, open work, and variances (waivers or deviations) are properly documented, and resolved.

f. All acceptance documentation is present, including any necessary signed certifications.

g. All tools and the development and build environments are available to the government.

3.3.2.13 Operation

The developer shall establish and maintain a plan and set of operational procedures for performing the activities and tasks associated with operational testing, systems operation, and user support. These tasks and activities shall include the following, as a minimum:

a. Procedures for testing the software product in its operational environment.

b. Procedure for receiving, recording, tracking problems, providing feedback on problems, and resolving problems. Problem reports and their resolutions shall be made available to the cognizant MDA Deputates and Elements or designated representative upon request.

c. Procedures for providing assistance and consultation to users, as required.

d. Procedures for forwarding user request, as necessary, to maintenance process for resolution.

e. Procedures for systems operations.

3.3.2.14 Software Maintenance

The developer shall establish and maintain a software maintenance plan. The plan shall be verified against specified requirements for maintenance of the software product. Requirements for the submission of maintenance reports shall be established and agreed upon as part of the maintenance plan. The organization responsible for software maintenance shall be identified early in the development process to allow a smooth transition into the maintenance phase.

The software maintainer shall establish procedures for receiving, recording, and tracking problem reports and modification requests from the users and providing feedback to the users. The developer shall analyze problem reports or modification requests for their impact on the organization, the existing system, and the interfacing system for the following:

a. Type (e.g., corrective, improvement, preventative, or adaptive to new environment).

b. Scope (e.g., size of modification, cost involved, time to modify).

c. Criticality (e.g., impact on performance, safety, quality, or security).

For modifications to the software the developer shall conduct an analysis and determine which documentation, software units, and versions of software need to be modified.

3.3.2.15 Software Retirement

In the event software is to be retired, a retirement plan to remove active support by the operation and maintenance organizations shall be developed and documented. The plan shall address the following, as a minimum:

a. Cessation of full or partial support.

b. Archiving of the software product and its associated documentation.

c. Responsibility for any future residual support issues.

d. Transition to new product, if applicable.

e. Accessibility of archive copies of data.

All associated development documentation, logs, and code should be placed in archives when the software is retired. Data used or associated with the retired software product shall be accessible in accordance with specified requirements.

3.3.3 Supporting Activities and Processes

The developer shall establish and maintain a planned and systematic set of activities and tasks, which ensure software processes and products conform to the requirements. This includes defining and implementing a software assurance program, software configuration management disciplines, and software documentation requirements. The software assurance program shall include Software Quality Assurance (SQA), Software Safety, Software Dependability, and Software Verification and Validation disciplines, as a minimum.

3.3.3.1 Software Quality Assurance

The software developer shall establish and maintain a Software Quality Assurance (SQA) Plan that addresses resources, schedule, and responsibilities for performing quality assurance activities. The plan shall identify procedures for assuring the implementation of SQA, verification, validation, reviews, audits and assessments, software problem reporting, software dependability, software assurance metrics, and other supporting processes, which can affect mission assurance, program performance, cost and schedule. The plan shall identify:

a. Standards, methodologies, procedures, and tools for performing the quality assurance activities.

b. Methods for collecting, measuring, analyzing, and comparing software products and data.

c. Procedures for identification, collection, filing, maintenance, and disposition of required software quality assurance records.

The SQA plan and subsequent updates shall be submitted to the cognizant MDA Deputates and Elements for review and approval and MDA/QS for information.

3.3.3.2 Software Verification

The developer shall plan and accomplish verification of the safety, quality, functional, and performance requirements allocated to software and firmware. Developer shall select a qualified organization to be responsible for conducting the verification effort. This organization shall be vested with the authority and independence necessary to perform this task. The developer at each stage shall ensure:

a. Planned verification activities are adequate to determine the products are compliant with inputs requirements.

b. Verification activities are performed using methods, procedures, and tools defined in the Software Development Plan and Software Quality Assurance Plan.

c. Planned verification activities include full verification of critical software.

d. Traceability matrices are verified at the completion.

The developer shall verify the outputs of each development stage for conformance against the inputs to that phase and demonstrate:

a. Conformance to appropriate development standards.

b. Acceptance criteria for forwarding to subsequent stages are contained or referenced.

c. Identification of product characteristics that are crucial to its safe and proper functioning.

The developer shall ensure that:

a. Verification results, including any problem reports and any corrective actions against specified requirements are met, recorded, and verified.

b. Traceability matrices are updated and verified at the completion of each task/stage.

The developer shall document the verification activities undertaken for each development task and their results. The documentation shall be made available to cognizant MDA Deputates and Elements upon request.

3.3.3.3 Software Validation

The developer validation organization shall be vested with the authority and independence necessary to perform this task. The developer shall ensure the organization responsible for validation tasks establishes and maintains validation procedures and criteria. The developer shall provide adequate resources for performing validation processes, developing work products, and providing support services. The developer shall provide trained personnel for performing or supporting the validation process.

The developer shall validate the software to ensure that it is suitable for use in its intended operating environment. Results from validation activity shall be analyzed and issues identified. Problems and nonconformances detected by the validation effort shall be documented and tracked to resolution (3.1.9).

The developer shall establish and maintain a validation environment. The validation environment shall include, as appropriate:

a. Test tools interfacing with the software being validated.

b. Temporary embedded software.

c. Recording tools for analysis or replay.

d. Simulated subsystems, components, or interface systems.

e. Real interface systems.

f. Facilities.

g. Test management tools, test case generators, test coverage analyzers, and emulators.

h. Loads, stress, and performance tools.

The developer shall establish and maintain test requirements, test cases, and test specifications, which reflect their intended use.

The developer shall establish and maintain a validation plan. The plan shall address, as a minimum:

a. Resources, responsibilities, and schedule for validation task.

b. Identification of items subject to validation or criteria for selecting items subject to validation.

c. Validation tasks to be performed, including associated methods, techniques, and tools.

d. Procedures for forwarding validation report/results to cognizant MDA Deputates and Elements.

e. Identification of validation work product and their appropriate configuration control.

f. Method used to monitor and control the validation process against the plan.

g. Review of activities, status, and results of the validation process with top level management.

h. Objective evaluation of the validation processes against its process description, standards, and plans. This discussion shall address how noncompliances will be resolved.

3.3.3.4 Support of Independent Verification and Validation

The software developers shall support MDA's Independent Verification and Validation (IV&V) efforts. This support includes:

a. Providing work products and associated documentation.

b. Participating in IV&V reviews of developers work products.

c. Providing work areas for IV&V personnel.

3.3.3.5 Independent Verification and Validation

The organization performing Independent Verification and Validation (IV&V) tasks and activities on MDA software or firmware products shall develop and implement an IV&V program in accordance with IEEE Standard 1012, Software Verification and Validation.

3.3.3.6 Software Reviews

The developer shall hold periodic reviews to assess technical, performance, and schedule progress. In addition, the reviews shall assess the projects success in implementing the software requirements of this provision. The developer software organization shall participate in or support all technical and mission assurance reviews (3.4). A documented procedure shall be established and maintained to address all software technical, management, and mission assurance reviews. Action items resulting from reviews shall be documented and tracked to resolution.

Program management shall review the activities for managing the software provisions in accordance with provision 3.1.2. This includes, but is not limited to, software and firmware development plans, project plans, test plans, process descriptions (e.g., standards, guides, procedures), allocated requirements, software and firmware requirements, software design, software code, test plans, test procedures, and test cases.

Developer's software engineering shall participate in the review of products produced or acquired for/by other engineering groups (within the developer's organization) to ensure that the products meet the receiving groups requirements and needs.

3.3.3.7 Software Audits

The developer shall perform software audits in accordance with 3.1.7. Developer software personnel performing the reviews or audits shall be independent of those responsible for producing the products or performing software activities. Software audits shall be held at predetermined milestones as specified in the SQA plan. Results of the audits shall be reported to software engineering management, software project management, and developer's top level management.

Software audits shall ensure that:

a. Coded software products reflect the design documentation.

b. Acceptance and testing requirements are adequate for acceptance of software products.

c. Software products are successfully tested and meet their specifications.

d. Test reports are correct and discrepancies between actual and expected results have been resolved.

e. User documentation is complete, accurate, and meets standards as specified.

f. Activities have been conducted according to applicable requirements, processes, procedures, and plans.

g. Cost and schedules adhere to establish plans.

The developer's software assurance program shall require software quality participation in the preparation, review and approval of software plans, standards, and procedures. Software quality shall review or audit the software engineering activities associated with software and firmware products to verify compliance with the Software Development Plan, procedures, and standards. Findings from reviews and audits shall be tracked to resolution (3.1.7).

3.3.3.8 Software Problem Reporting

The developer shall utilize the problem failure reporting and corrective action system (3.1.9) to report, investigate, analyze and correct software nonconformances and problems. When problems (including nonconformances) have been detected in a software product or activity, a problem report shall be prepared to describe each problem. The requirements of this section supplement those of 3.1.9. Software nonconformance and corrective active reporting shall commence with baselining of a work product.

Problem resolutions and dispositions shall be evaluated to verify that problems have been resolved, adverse trends have been reversed, and changes have been correctly implemented in the appropriate software, products, and activities. In addition, resolutions and dispositions shall be reviewed to determine whether additional problems have been introduced.

The developer's software projects shall use defect prevention techniques to identify defect causes and provide assessment for potential process improvement opportunities. The responsibilities and authorities for implementing the defect prevention activities shall be documented in software project plans.

The developer's defect prevention program shall include: causal analysis; periodic review and coordinated implementation of actions; documentation and tracking of defect prevention data; incorporation of revisions to organization's and project's software processes resulting from

defect prevention actions; and feedback to software engineering and related groups on defect prevention activities.

3.3.3.9 Software Dependability

The developer's software dependability program shall address:

a. Identification and mitigation of risks associated with software failures.

b. Emphasis on building in software error prevention, fault detection, isolation, recovery, and operating at reduced functional capability/states.

c. Measuring and analyzing defects in the software product during development to find and address problem areas within the software.

3.3.3.10 Software Assurance Metrics

The developer shall establish and use software metrics to determine the status of the activities for the software requirements herein. The metrics shall be used to determine:

a. Cost effectiveness, and schedule of the Quality, Safety, and Mission Assurance activities.

b. Quality of the training program.

c. Functionality and quality of software products.

d. Maturity of developer's software processes.

3.3.3.11 Software Safety

The software safety program shall be performed in accordance with the safety provision (3.14).

3.3.3.12 Software Configuration Management

The developer's software configuration management activities shall be performed in accordance with provision 3.10 and the following requirements. The requirements of this section also apply to software and firmware items that are supplied as GFI, GFE, COTS, and NDI. Software products to be placed under Configuration Management (CM) control shall be identified in the CM plan along with the milestone associated with placing the product under control. Each project software library system used as a repository for software baselines shall be documented in the CM plan.

3.3.3.12.1 Software Configuration Items

For each Software Configuration Item (SCI), the developer shall identify its corresponding Software Components (SCs) and Software Units (SUs). For each SCI, SC, and SU the developer shall issue/obtain a software identifier, which consist of a name or number, and a version identifier, and relate the software to its associated software design documentation; revision; and release date. The developer shall embed the software and version identifiers within the source code, and provide a method for display of the software and version identifier data to the user upon command. The marking and labeling of software media shall include a software identifier and version.

3.3.3.12.2 Software Change Control Process

The developer's change control process (3.10.3) shall address how software changes are to be identified, documented, submitted, reviewed, approved or disapproved, implemented, verified,

and released. The configuration control board shall have appropriate disciplines represented to process software changes.

The developer's change control process shall ensure that only authorized changes are implemented into software or firmware products. The configuration control system shall ensure that any referenced version of software or firmware can be re-generated from backups.

The configuration control process shall address controls over and changes to tools used in code generation and testing of the deliverable software product. It shall also address how supplied (GFE, GFI, NDI, third party) software (e.g. source, executable, or data) shall be protected against corruption.

The configuration control process shall address variances (waivers and deviations) (3.10.3.4) associated with software activities and products. Variances shall be documented, reviewed, and resolved with the appropriate software engineering manager, project manager, and other appropriate groups.

3.3.3.12.3 Software Library

The developer shall establish and maintain a software library system to facilitate control of software products. Software library systems shall provide a method for storage of current and superseded versions of software programs, and software tools required to maintain and use the software. The library system shall provide for the following:

a. Maintenance of and controlled access to approved configurations of software programs and associated design disclosure documentation.

b. Maintenance of software tools and related documentation.

c. Controls to assure the integrity and security of the software programs and associated documentation.

The developer shall define how software products are created, entered, updated, and released from the software library. This definition shall address the various levels or types of controls imposed over various software products throughout their life.

The developer shall maintain a second off-site repository containing duplicate files of all software programs, design disclosure documentation, and support software or tools to allow for retrieval in the event of a disaster.

3.3.3.12.4 Software Configuration Audits

The developer, prior to the establishment of software baseline (allocated, requirements, architecture, design), shall perform a software baseline configuration audit to determine completeness, accuracy, consistency, and quality. Additionally, software configuration change audits shall be conducted to ensure that approved changes, and only approved changes, are incorporated into the software product or its technical descriptions. The developer shall support software functional and physical configuration audits in accordance with 3.10.5.2

Problems identified, as a result of the audits, shall be documented, tracked, controlled and resolved.

3.3.3.12.5 Software Status Accounting

The developer's status accounting system (3.10.4) shall provide management with records and reports to show the status and history of controlled items. Status reports shall include the status of proposed and approved changes for each Software Configuration Item, Software Components, and Software Units;

outstanding variances (waivers and deviations); outstanding problem reports; latest software item versions; release identifiers; number of releases; and comparisons of releases. Status accounting system shall provide a correlation between configuration status of the software program, its documentation, and associated hardware. Configuration management status reports shall be developed and made available to affected groups, individuals, and the cognizant MDA Deputates and Elements or designated representative upon request. The developer's software configuration status accounting file shall be available and up to date for each project milestone.

3.3.3.12.6 Software and Firmware Media Generation

The developer shall establish and maintain a controlled system to assure integrity of deliverable software and data during transfer to transportable media, firmware, deliverable hardware, or test and inspection equipment. This system shall include verification that each copy of software or data is an accurate replication of the master copy retained in the library. The results of these media generation and verifications shall be documented and maintained as quality records.

3.3.3.13 Software Documentation

The developer shall establish and maintain a documentation process for recording information produced by a software task, activity, or process. Process documentation shall define the set of activities, which plan, design, develop, produce, edit, distribute, and maintain those documents needed by managers, engineers, and users of the system or software product. Process or project documentation shall be updated to reflect improvements.

The developer's software process documentation shall include, as a minimum:

a. Software models, tools, and techniques approved for use.

b. Guidelines and criteria for tailoring the organization's standard software processes.

c. Requirements and guidelines for establishing and maintaining software process databases.

d. A library and guidelines for software process related documentation (e.g., procedures, manuals, project plans, standards).

Appropriate configuration management controls shall govern software documentation. The developer shall verify that software documentation is available, adequate, current, complete, and consistent.

3.3.3.13.1 Software Operation and Maintenance Documentation

Software operation and maintenance documentation shall be developed and delivered in accordance with end user requirements. Preliminary and/or draft versions of the documentation shall be made available at the critical design review for review and comment by end users, government, maintainers, and software quality. Documentation shall be updated based upon feedback from reviewers.

Final operation and maintenance documentation presented for government acceptance shall:

a. Define delivered capabilities and limitations.

b. Be proofed or qualified during software qualification testing.

3.4 Technical and Mission Assurance Reviews

The developer shall support technical and mission assurance reviews to ensure the design meets mission requirements and to reduce mission risk to acceptable levels. Reviews shall be tailored to specific needs of each program and mission based on risk, schedule, and funding. Results from these reviews shall be documented and action items tracked to resolution.

3.4.1 Technical Reviews

The developer shall support technical reviews to determine design maturity and to ensure the design is technically adequate and meets requirements. Technical reviews shall be event driven and conducted at appropriate points in development when progress merits review to check design maturity, review technical risk, and determine whether to proceed to the next level of development. Technical reviews shall be integrated into the systems engineering process and conducted by a joint IPT composed of government and supplier personnel, and attended by non-advocate technical personnel. Formal technical reviews shall be preceded by a series of technical interchange meetings where issues, risks, problems, and concerns are surfaced and addressed. The technical review is used as a confirmation of completed effort, not a forum for problem solving. In preparation for technical reviews the developer shall make available to review participants necessary documentation, material, and analyses regarding the design in advance of the scheduled event to allow sufficient time to examine the information in detail. Information such as specifications, results from tradeoff studies and design analyses, drawings, manuals, schedules, design and test data, risk analyses and mitigation activities, interface specifications, test methods and plans, code, technical plans, and metrics may be needed for review. Technical reviews may be conducted at both developer and supplier sites, as appropriate. For each review, the developer shall, as appropriate:

a. Provide necessary agenda, plans, administrative support, and facilities.

b. Ensure participation by subject matter experts including suppliers.

c. Provide information and items necessary to demonstrate and confirm that accomplishments associated with the specific review event have been satisfied.

d. Substantiate trade-off decisions with technical details and associated rationale.

e. Document proceedings with associated rationale for key points, decisions, and issues.

f. Document all open and unresolved items with their closure requirements and due dates, and assign organizations' responsibilities.

g. Identify risks, including safety risks.

Technical reviews will be tailored to specific needs of the program. The following set of reviews depicts a normal sequence in terms of assessing technical progress from concept through production. The developer shall participate in the following reviews as directed by the cognizant MDA Deputates and Elements. Additional reviews may be required on a program-by-program basis.

3.4.1.1 Systems Requirements Review

Systems Requirements Reviews (SRR) are conducted to demonstrate progress in converging on viable, traceable system requirements that are balanced with cost, schedule, and risk. The developer shall participate in the SRR. The following are examples of documentation typically reviewed at the SRR:

a. System operational requirements.

b. Draft system specification and any initial performance item specifications.

c. Functional analysis (top level block diagrams).

d. Results of trade studies and critical technology assessments.

e. System maintenance concept.

f. System design criteria.

g. System engineering planning.

h. Test and evaluation master planning.

i. Draft top-level technical performance measurements.

j. System design documentation.

k. Total ownership costs.

l. Critical accomplishments, success criteria, and metrics, risks and mitigation activity.

m. Safety requirements.

3.4.1.2 System Functional Review

System Functional Reviews (SFR) are conducted to establish and verify an appropriate set of functional and performance requirements for the system, to demonstrate convergence on achievability of system requirements, and readiness to initiate preliminary design. The developer shall participate in an SFR. The following are examples of inputs to the SFR:

a. Verification that the system specification reflects requirements that will meet user expectations.

b. Functional analysis and allocation of requirements to items below system level.

c. Draft item performance and some item detail specifications.

d. Design data defining the overall system.

e. Verification that the risks associated with the system design are at acceptable levels for engineering development.

f. Verification that conceptual design selections have been optimized through appropriate trade study analyses.

g. Supporting analyses, (e.g. logistics, human systems integration) and plans are identified and complete where appropriate.

h. Technical performance measurement data and analysis.

i. Pre-planned product and process improvement and evolutionary development requirements and plans have been defined.

j. An audit trail from SRR is established with changes substantiated.

k. Risk handling approach has been defined for the next phase or technical effort including safety.

l. Implementation requirements for technology transition have been defined.

m. Critical accomplishments, success criteria, and metrics have been defined for the next phase or continued technical effort.

n. Identification of COTS and NDI hardware and software.

3.4.1.3 Software Specification Review

Software Specification Reviews (SSR) are conducted to demonstrate convergence on computer software subsystem requirements as an integrated part of system and subsystem requirements, and readiness to initiate preliminary design for the computer software subsystem. The developer shall participate in SSRs. The following are examples of inputs to SSRs:

a. Evaluation of the maturity of software requirements.

b. Validation that the software requirements specification and the interface requirements specification reflect the system-level requirements allocated to software.

c. Analysis of computer hardware and software compatibility.

d. Evaluation of human interfaces, controls, and displays.

e. Assurance that software-related risks have been identified and mitigation plans established.

f. Validation that all designs are consistent with the Operations Concept Document.

g. Plans for testing and traceability matrix.

h. Review software development planning.

i. Safety issues addressed.

3.4.1.4 Preliminary Design Assessments/Critical Design Assessments

The developer shall perform Preliminary Design Assessments (PDA) and Critical Design Assessments (CDA) with participation by the cognizant MDA Deputates and Elements and designated technical representatives. PDAs are focused, in-depth working-level technical design reviews, conducted incrementally, that support the evolving design and development of a product and occur prior to a Preliminary Design Review (PDR). CDAs are focused, in-depth working-level technical design reviews, conducted incrementally, that support the detailed design development of a product and occur prior to Critical Design Review (CDR). Both assessments address specific functional areas or aspects of a design to demonstrate requirement satisfaction. These assessments are an outgrowth of technical working groups and follow the format and guidelines described for PDR and CDR. Data generated in preparation for and as a result of these reviews will support preparations for PDR and CDR. These are working-level meetings and can represent a dry run of PDR and CDR presentation material. Independent technical experts (representatives not assigned to the project) ensure that all requirements are met, the design approach is verified, risks are identified, and mitigation plans are generated.

3.4.1.5 Preliminary Design Review

Preliminary Design Reviews (PDR) are conducted by the cognizant MDA Deputates and Elements prior to the detail design process to evaluate progress and technical adequacy of the selected design approach; determine its compatibility with the performance requirements of the specification; and establish the existence and physical and functional interfaces between the item and other items of equipment or facilities. A series of PDRs are normally held for each Configuration Item (CI) or aggregate of CIs, or

subsystem, leading to a system PDR for completion. The developer shall participate in the PDR and should be prepared to present and discuss the following items, as applicable:

a. Item performance specifications.

b. Operational Concepts document.

c. Draft item detail, process, and material specifications.

d. Design data defining major subsystems, equipment, software, and other system elements.

e. Allocated requirements baseline for the subsystem.

f. Analyses, reports, reliability, maintainability, testability, availability, producibility, supportability analyses, trade studies, logistics support analysis data, and design documentation.

g. Technical Performance Measurement data and analysis.

h. Laboratory and test models, mockups, and prototypes to support the design.

i. Supplier data describing specific components.

j. Risk and mitigation status.

k. Critical accomplishments, success criteria, and metrics are valid for continued technical effort.

l. Safety item status.

3.4.1.6 Critical Design Review

Critical Design Reviews (CDR) are conducted when detail designs are essentially complete, configuration documentation is ready for release, and the configuration item is ready for fabrication or coding. CDRs are conducted to determine detail designs satisfy the design requirements established in the specification and establish the interface relationships. The developer shall participate in the CDR and should be prepared to present and discuss the following items, as appropriate:

a. Producibility and risk areas.

b. Subsystem and functional issues have been resolved.

c. Preliminary product specifications review.

d. Test and evaluation analysis results support critical subsystem design and interface requirements and design constraints.

e. Verification and flowdown of requirements.

f. Critical accomplishments, success criteria, and metrics are valid for continued technical effort.

g. Known safety defects and risks.

h. Design performance (by inspection, test, demonstration, or analysis) meets requirements.

3.4.1.7 Test Readiness Review

Test Readiness Reviews (TRR) are conducted for each critical subsystem to confirm completeness of test procedures, to ensure that the subsystem/system is ready for testing, and to ensure that the performing activity is prepared for formal testing. The TRR will be conducted after the critical gate process (3.7.7.1) is completed. The developer shall participate in the TRR and should be prepared to present and discuss the following items, as appropriate:

a. Test procedures comply with test plans and descriptions, demonstrate adequacy to accomplish test requirements, and satisfy subsystem specification requirements for verifications.

b. Pre-test predictions and informal test results indicate testing will confirm necessary performance.

c. Test support equipment, facilities, and procedures required to accomplish planned test and evaluation are available and satisfy their requirements.

d. Required operation and support documents are complete and accurate.

e. Data acquisition, handling, and analysis provisions have been met.

3.4.1.7.1 MDA Executive Level Flight Test Reviews

The developer shall support the MDA executive level flight test reviews as described in Guideline for MDA Formal Flight Test Reviews. The MDA executive level flight test review process provides the structure needed to assure MDA senior leadership that critical issues involved in the planning, preparation, and execution of a flight test are satisfactorily resolved prior to the test event and that the flight test results yielded the desired data and analysis. Each flight test review briefing is intended to provide the proof for MDA senior leadership to authorize proceeding with the Flight Test. The flight test review process includes a series of four flight test reviews. The post-test reporting requirements include a series of four reporting activities. These guidelines apply to all MDA elements and system level flight tests. For other critical ground based tests, the MDA/D may determine that these reviews are warranted. This process can be tailored to cover a flight test campaign or series of tests over a short time that demonstrates a significant capability.

3.4.1.8 System Verification Review

System Verification Reviews (SVR) are conducted following a successful Functional Configuration Audit (3.10.5.2) to demonstrate that the total system (people, products, and processes) is verified to satisfy requirements in the functional and allocated configuration documentation and to confirm readiness for production, support, training, deployment, operations, subsequent verifications, additional development (if any), and disposal. SVRs determine that the system produced is capable of meeting the technical performance requirements established in the specifications and test plans. SVRs are conducted incrementally, starting with early stages of manufacturing planning and progressing to a more detailed level as the design matures. The developer shall participate in the SVR and should be prepared to present and discuss the following items, as appropriate:

a. Readiness issues for continuing design, subsequent verifications, fabrication, training, deployment, operations, support, and disposal have been resolved.

b. Configuration audits, including completion of all change actions, have been completed for all configuration items.

c. Systems engineering and risk management planning has been updated for fabrication.

d. Critical achievements, success criteria and metrics have been established for fabrication.

e. Risk mitigation for high and moderate risks are in place.

f. Safety assessment is completed.

SVRs are also known as Production Readiness Reviews (PRR).

3.4.2 Mission Assurance Reviews

Mission Assurance Reviews shall be performed to clarify and ratify mission requirements (planning and design), and to discuss issues and approaches, and to communicate decisions. Reviews shall ensure known issues and problems are dispositioned prior to each critical event. Cognizant MDA Deputates and Elements shall provide technical experts as panel members. Completion of activities necessary to fulfill specific readiness review criteria shall also be accomplished during Mission Assurance Reviews. The developer shall participate in Mission Assurance Reviews as required. The following Mission Assurance Reviews are used as a sequential process to mitigate risk and assure mission success. The developer may propose a Mission Assurance Review process tailored for a particular system as an alternate to the process described below.

Mission Assurance Reviews do not replace other government reviews or certifications required by federal regulation or law.

3.4.2.1 Mission Readiness Review

The developer shall support and participate in a Mission Readiness Review (MRR) conducted 4-6 weeks prior to launch. The review shall address all components of mission readiness: project status, test objectives and mission performance, instrument readiness, launch vehicle readiness, ground system readiness, launch service readiness and launch site assessment, resolution of all open items, liens and waivers, public affairs plan, safety assessment, and other topics to ensure all aspects critical to mission success have been reviewed. The MRR results shall be presented to the mission review board for review and certification of the readiness of all mission components to proceed toward launch.

3.4.2.2 Pre-Environmental Review

The developer shall participate and support a Pre-Environmental Review (PER) to assess readiness of flight hardware, software, and required environmental test facilities to begin acceptance testing. The PER shall occur prior to the start of environmental testing of the prototype or flight system. The primary purpose of this review is to establish readiness of the system for test and evaluate environmental test plans. The PER shall be held prior to full system integration and functional test in preparation for environmental testing.

3.4.2.3 Pre-Shipment Review

A Pre-Shipment Review shall be conducted prior to shipment of flight test assets for integration with the ground support system/launcher at the test range. This review shall include scrutiny of hardware build-up and acceptance test pedigree data for the flight hardware as well as reports on any ground test anomalies. Hardware on-times/cycle-times shall also be reviewed and shown to be within acceptable limits. The developer shall provide status of the tracking of the safety items listed in the validation tracking log, status of deliverable documents to the launch range, and any subsequent launch range issues or necessary approvals prior to sending flight hardware to the range. A panel of cognizant government representatives will assess the data and provide recommendation to ship the hardware.

3.4.2.4 Flight Operations Review

The developer shall participate and support a Flight Operations Review (FOR) to assess the adequacy of final operation plans and compatibility of flight components with ground support equipment and ground network, including results of network compatibility tests. The FOR is held after the system has been configured for launch. The purpose of the FOR is to: (1) examine demonstrations, tests, analyses, and

audits which determine system readiness for safe and successful launch and subsequent flight operations; (2) ensure that all flight and ground hardware, software, personnel, and procedures are ready and compatible.

3.4.2.5 Pre-Flight Readiness Reviews

Prior to the decision to conduct a flight test, the developer shall support a Pre-Flight Readiness Review during which a government panel will conduct a detailed review of the readiness of the flight test design, test asset, target, and test range for conduct of the mission. Developmental and system readiness testing for all test-unit components and associated support sub-elements shall be reviewed, along with results from any flight test unit problems/anomaly investigations, associated resolutions, and documentation (including a summary of the Pre-Shipment Review conclusions). As part of this review process, the developer shall provide results from qualification testing to demonstrate that all critical components have been fully qualified for expected flight conditions, including margins (at least 3 dB above predicted environments) to handle unexpected conditions. Target test and pedigree data will be reviewed as well. Ground system acceptance testing and integrated testing of ground support systems shall be reviewed along with data for all system interfaces. In addition, range countdown and launch procedures and processes will be reviewed. Flight safety analyses (addressing destruct limit lines and debris patterns) shall be confirmed to be acceptable. This process shall also include certification of the flight test scenario and associated flight objectives. Accredited modeling and simulation results (including hardware-in-the-loop test data) shall be used to provide flight test performance predictions and demonstrate that all flight test objectives will be met. This overall process shall culminate in a series of meetings conducted to obtain MDA Director approval to perform the flight mission.

3.4.2.6 Launch Readiness Review

The developer shall participate and support a Launch Readiness Review (LRR) to assess the overall readiness of the total system to support the flight objectives of the mission. The LRR shall be held at the launch site not less than two to three days prior to launch. The review shall cover all activity since Pre-Shipment Review (PSR), closure of any required actions, and summation of all testing and launch operations planning and rehearsals to the present. Any residual risks including safety shall be presented at this time. Closure of this review and any actions generated from the review indicate the mission is ready for launch.

3.4.2.7 Mission Operations Review

The developer shall participate and support a Mission Operations Review (MOR) prior to significant integration and test of flight systems and ground data systems. The MOR shall establish status of the system components, including the ground data systems and its operational interface with flight systems. Discussions shall include mission integration, test planning, safety assessment, and status of preparations for flight operations.

This Page Intentionally Left Blank

3.5 Reliability, Maintainability, and Availability

The developer shall establish and maintain a Reliability, Maintainability, and Availability (RM&A) program as an integral part of system design and development process to track the designs ability to meet or exceed the product's mission needs and objectives. The program shall provide tools used to create designs at reduced total ownership cost, and identify problem areas in the design that require additional development. The RM&A program executed during design and development will improve operational readiness and mission success; reduce item demand for maintenance and logistic support; and provide essential risk management information.

3.5.1 Reliability, Maintainability, and Availability Program Plan

The developer shall develop and maintain a coordinated Reliability, Maintainability, and Availability (RM&A) Program Plan that describes the planning and implementation of RM&A activities, including tests, analyses, and associated ground rules and assumptions. In addition, the plan shall include specific reliability design criteria that define both appropriate and inappropriate devices, materials and processes for the design's application based historical data and new technology assessments. The coordinated RM&A Program Plan shall be submitted for review and approval to the cognizant MDA Deputates and Elements.

3.5.1.1 Reliability, Maintainability, and Availability Program Planning

The planning shall identify the Reliability, Maintainability, and Availability (RM&A) tasks to be performed, and describe how the RM&A tasks will be implemented and controlled. The RM&A program planning shall identify scheduling of RM&A tasks relative to project events. The planning effort shall identify the activities that ensure RM&A functions are an integral part of design and development processes and that RM&A functions interact effectively with other project disciplines, including systems engineering, hardware, software, logistics, safety, design, and mission assurance. The planning effort shall also identify how reliability assessments will be integrated with the design process and other assurance practices to maximize the probability of meeting mission success criteria.

3.5.2 Supplier Reliability, Maintainability, and Availability Requirements

The developer shall establish and maintain management procedures and design controls including allocation and flow down of Reliability, Maintainability, and Availability (RM&A) requirements to mission critical suppliers. Plans and data to support specified RM&A requirements shall be made available to the cognizant MDA Deputates and Elements or designated representative upon request.

3.5.3 Failure Reporting Analysis and Corrective Action System

As part of the overall problem failure reporting and corrective action system (3.1.9), the developer shall establish and maintain a closed loop Failure Reporting Analysis and Corrective Action System (FRACAS), which serves as a management tool to identify, correct, and prevent further recurrence of all failures occurring in hardware and software during system debugging, engineering tests, qualification tests, ESS, receiving inspection & test, fabrication, acceptance tests, flight tests, and field failures returned to the factory. FRACAS shall ensure that failures are documented, analyzed, and timely corrective actions are taken to reduce or prevent recurrence. The closed-loop feature of FRACAS requires information obtained during failure analysis be disseminated to all decision-making program engineers, managers, and the cognizant MDA Deputates and Elements or designated representative.

3.5.4 Failure Review Board

The developer shall establish a Failure Review Board (FRB) to review all failure data documented in Failure Reporting Analysis and Corrective Action System (FRACAS). The review process shall include the assessment of failures, failure trends, and corrective action status and effectiveness. The FRB shall examine data including a description of test conditions at the time of failure, symptoms of failure, failure

isolation procedures, and known or suspected causes of failure. The FRB's objective shall be to improve reliability and maintainability of hardware and associated software by use of failure and maintenance data.

Open FRB items shall be tracked until root cause failure mechanisms have been satisfactorily identified, corrective action initiated and verified. FRB members shall include the cognizant MDA Deputates and Elements or designated representative, and representatives from reliability, design, manufacturing, quality, and system safety, as appropriate.

3.5.5 Reliability Modeling, Allocation, and Prediction

The developer shall develop and maintain a reliability model (reliability block diagrams and math models) for each system, subsystem, and lower levels with associated allocations and predictions for all items in each reliability block. Each block shall include function and item identification to a level consistent with design maturity. Basic reliability or mission reliability requirements shall be used to establish baseline requirements for designers.

Reliability allocation shall be based on mission and configuration item reliability requirements. Reliability requirements shall be allocated to each indenture level and imposed on suppliers, as applicable.

Reliability predictions shall be derived and applied through the level of individual piece parts. The developer shall update reliability predictions as the design matures and valid test data becomes available. The results of reliability predictions shall be used as inputs in formulating decisions for product design, safety, maintenance, logistics, and availability analyses. Failure rate data used in the predictions shall be selected from sources that reflect the intended application. Predictions for electrical, mechanical, structural, optical, and electro-mechanical equipment shall be made with either data or alternatives, both of which shall be identified in the Reliability, Maintainability, and Availability Program Plan.

All Major (Class I) engineering changes shall be assessed to determine the need for additional modeling and predictions. Reliability modeling, allocation, and prediction status and analysis for new and redesigned systems, subsystems, and equipment shall be provided at design reviews.

3.5.5.1 Reliability Prediction Methodology

Reliability predictions for mission and safety critical electronic and mechanical equipment, including COTS, shall be made using parts stress analysis methodology, or intended environment field data.

Reliability predictions for non-mission critical electronic equipment, including COTS, shall be made using parts count or similar item methodology, or intended environment field data.

3.5.6 Reliability Analyses

Reliability analyses shall be performed concurrently with design so that design deficiencies can be addressed. The following are reliability analyses required on systems, subsystems, and assemblies.

3.5.6.1 Failure Modes, Effects, and Criticality Analysis

The developer shall conduct a Failure Modes, Effects, and Criticality Analysis (FMECA) to identify potential failure modes of the product design for each mission phase and to estimate the effect of failure modes on mission success and safety. Failure modes shall be identified at the piece part level for newly designed and modified mission critical items. A functional FMECA shall be performed for existing and off-the-shelf mission critical equipment, products and systems. Each failure mode shall be assessed and analyzed for the effect at each level of the assembly up to the end item. The failure mode shall be assigned a severity category based on the most severe effect caused by a failure. The developer shall initiate a FMECA early in the design phase and update the analysis to reflect affected changes to design configuration. As a minimum, FMECA shall include:

a. Identification number.

b. Item/functional identification.

c. Failure modes and causes.

d. Mission Phase / Operational Mode affected.

e. Failure Effect.

f. Severity Classification.

g. Failure detection methods.

h. Compensating Provisions.

i. Impact on safety, mission success, readiness, and demand for maintenance/logistics support.

j. Criticality Analysis.

k. Methods and results for obtaining probability of occurrence and recommended actions to preclude or reduce probability of occurrence.

The FMECA shall be scheduled and completed concurrently with the design effort so that designs reflect analysis conclusions and recommendations. When fault trees are used to aid in the FMECA, they shall be documented to the level where recommended action can be taken. The results and current status of FMECA shall be used as inputs to system engineering process. Results and methods of the analysis shall be documented and the FMECA report shall be submitted to the cognizant MDA Deputates and Elements for review and approval. Severity ranking criteria shall be determined as follows:

a. *Category I (Catastrophic)*. A failure, which can cause death or system loss (e.g., aircraft, satellite, missile, ship).

b. *Category II (Critical).* A failure, which may cause severe injury, major property damage, or major system damage, which will result in mission loss.

c. *Category III (Marginal).* A failure, which may cause minor injury, minor property damage, or minor system damage, which will result in delay or loss of availability or mission degradation (including loss of redundancy).

d. *Category IV (Minor).* A failure not serious enough to cause injury, property damage or system damage, but which will result in unscheduled maintenance or repair.

The severity ranking criteria shall be included in the FMECA ground rules within the Reliability, Maintainability, and Availability Program Plan, and coordinated with safety, software, and logistics disciplines. All items with failure modes that are assigned to Severity Categories I and II shall be itemized on a Mission Critical Items List and maintained with the FMECA report.

3.5.6.2 Fault Tree Analysis

The developer shall perform a Fault Tree Analysis (FTA) to support Failure Modes, Effects, and Criticality Analysis, design studies, and failure investigations, as appropriate. Beginning with each undesired event, the fault tree shall be expanded to include all credible combinations of events/faults and environments that could lead to the undesired event. Component hardware/software failures, external hardware/software failures and human factors shall be considered in the analysis. Typical candidates for FTA are functional paths or interfaces which could have critical impact on flight safety, munitions handling

safety, safety of operating and maintenance personnel, and probability of error free command in automated systems in which a multiplicity of redundant and overlapping outputs may be involved. Other candidates for an FTA include troubleshooting, repair of products, and prediction and quantifying risk.

3.5.6.3 Finite Element Analysis

The developer shall use Finite Element Analysis (FEA) as a tool to analytically assess the behavior of engineering components, subsystems, and systems under various conditions of use through the knowledge of fundamental physics and advanced numerical techniques. FEA is performed to analyze the effects of stress (e.g. thermal, physical, natural frequencies) on parts. Typical candidates for FEA include devices, components, or design concepts that:

a. Are unproven and for which little or no prior experience or test information is available.

b. Use advanced or unique packaging or design concepts.

c. Will encounter severe environmental loads.

d. Have critical thermal or mechanical performance and behavior constraints.

3.5.6.4 Sneak Circuit Analysis

The developer shall perform a sneak circuit analysis on mission critical circuitry affecting system performance and safety where failure results in a Category I or II severity. Sneak circuit analysis shall identify latent paths, which cause unwanted functions or inhibit desired functions, assuming all components are functioning properly. A list of those circuits and functions analyzed, priorities given each subassembly in the analysis, and supporting rationale for the selections shall be maintained and presented at design reviews.

3.5.6.5 Worst Case Analysis

The developer shall perform worst case analyses where failure results in a Category I or II severity. The most sensitive design parameters shall be analyzed, including those subject to variations that could degrade performance. The adequacy of design margins in the electronic circuits, optics, electro-mechanical and mechanical items shall be demonstrated by analyses, test or both. The analyses shall consider all parameters set at worst case limits and worst case environmental stresses. Part parameter values for analyses shall include manufacturing, temperature, and cumulative radiation variability, and aging effects of environment. The analyses shall be updated with design changes. The analysis results shall be presented at design reviews. When a worst case analysis cannot be performed, which may be the case with COTS/NDI, an alternative approach shall be proposed to the cognizant MDA Deputates and Elements.

3.5.6.6 Electrical, Mechanical, and Thermal Stress Analyses

The developer shall perform electrical, mechanical, and thermal parts/circuits analysis for new designs and proposed design changes. These analyses shall be documented and results reported at design reviews. Electrical, mechanical, and thermal stress analyses may be performed on sheltered equipment, when appropriate, to minimize program risk.

3.5.6.6.1 Thermal Stress Analysis

The developer shall conduct thermal stress analysis to determine anticipated operational and self-induced temperatures for each mechanical or electrical assembly and component involved in a new design or proposed design change. The stress analysis shall be conducted at worst case environmental and load conditions. The criteria shall ensure that assemblies and components are capable of functioning in a temperature environment that does not exceed the item's derated limits including safety margins.

3.5.6.6.2 Mechanical Stress Analysis

The developer shall conduct a mechanical/structural stress analysis to verify that appropriate margins of safety exist. The stress analysis shall be conducted at worst case environmental and load conditions. Unacceptable stress conditions shall be eliminated.

3.5.6.6.3 Electrical/Electronic Stress Analysis

The developer shall conduct electrical/electronic stress analysis on all new designs including designs incorporating COTS/NDI and design modifications to determine, from the circuit and the operating conditions of a given application, the actual stresses induced on each part. The stress analysis shall be conducted at worst case environmental and load conditions. Unacceptable stress conditions based on derating (3.6.12.2) criteria shall be eliminated.

3.5.7 Mission Critical Items

The developer shall utilize Failure Modes, Effects, and Criticality Analysis and criteria below to identify and control mission critical items.

a. Impact of potential failure on safety, readiness, mission success, and demand for maintenance/logistics support.

b. Item has a critical failure mode and a relatively high failure rate.

c. Item is destroyed or expended upon being activated (one-shot device), item performance approaches its design limits, or transient stress degrading the item's inherent reliability.

d. Item's criticality ranking is not mitigated by Pre-Launch Built-In-Test (BIT) or other means to prevent or substantially reduce the probability of a hazardous launch.

e. Application of new technology, new materials, new processes, or advanced state-of-the-art techniques.

f. Complex production or technical complexity.

g. Limited source, limited material, or sole source availability.

h. Item has exhibited an unsatisfactory operation history.

i. Physical properties of the item are stability sensitive requiring tight process control.

Methods for controlling and testing mission critical items shall be established and documented. Controls may include supplier surveillance, configuration control/process change reporting, problem reporting, and agreed to tests and inspections. A list of mission critical items shall be made available to the cognizant MDA Deputates and Elements or designated representative upon request.

3.5.8 Effects of Functional Testing, Storage, Handling, Packaging, Transportation, and Maintenance

The developer shall establish and maintain a system to analyze effects of functional testing, storage, handling and transportation on the system reliability. Lifting devices shall be certified for load capability (3.12.11.2), safety critical operations will be monitored and shipping containers (for environmentally sensitive equipment) will be instrumented to monitor and record appropriate environmental parameters (e.g. temperature, contamination, and vibration) during shipment. The results of this effort shall be used to support long term failure rate predictions, design trade-offs, definition of allowable test exposures, retest after storage decisions, packaging, handling, or storage requirements, and refurbishment plans.

3.5.9 Controlled and Limited Life Items

The developer shall establish and maintain a system for the determination and identification of controlled and limited-life items and criteria for their storage (e.g., First-In-First-Out), control, and use. The system shall:

a. Include all subsystems, parts, devices, items, or materials whose useful life expectancy is limited or must be controlled.

b. Provide for establishing, validating, and updating the life expectancy of each limited-life or controlled item.

c. Prevent issuance, usage, and provide for the removal, replacement, review, and disposition of limited-life or controlled items whose specified useable life has expired.

Records shall be maintained that allow evaluation of cumulative stress (time and/or cycles) for controlled items, starting when useful life is initiated and indicating the project activity that stresses the items. The use of a controlled item whose expected life is less than its mission design life shall be approved by Parts and Materials Control Board (3.6.2).

3.5.10 Reliability Growth Testing Program

The developer shall implement a Reliability Growth Test Program in which the systems and subsystems are tested under actual or simulated operational conditions for the purpose of disclosing design deficiencies and defects and enhancing system reliability through the analysis, and correction of defects and the verification of the corrective action effectiveness. The reliability growth program and planning shall include methods for achieving reliability growth and for assessing reliability growth progress consistent with program needs including hardware and program duration. This program and planning are expected to include product availability, the test procedures to be used, criteria for correcting failure modes, applicable exit criteria, expected test times and sample sizes, and methods of analyzing test data and reporting results. The developer is expected to use industry best practices designed to minimize the program risks by achieving maximum reliability growth. Reliability growth approach shall be incorporated into the Reliability, Maintainability, and Availability Program Plan.

Proposed COTS/NDI shall be subjected to Reliability Qualification Testing (RQT) when the particular hardware or software has not been used under the worst case environments defined by the system specification and when there is not sufficient analytical data to support the hardware's allocated reliability to comply with the overall system reliability.

3.5.11 Accelerated Life Testing

The developer shall establish and maintain an Accelerated Life Testing (ALT) program to detect and correct any inherent design and manufacturing flaws and to determine product robustness of mission critical items. The developer shall establish selection criteria to identify ALT candidates. Criteria and candidates shall be made available to the cognizant MDA **Deputates and Elements** or designated representative upon request. ALT shall be used during development in an iterative fashion beginning at lower levels of assembly and progressing to higher levels of assembly until sufficient margins have been verified. Test methods shall include a series of individual and combined stresses applied in steps of increasing intensity (well beyond the expected field environments) until failure or a malfunction is obtained. Failure modes shall be analyzed for root cause and corrective action, as appropriate.

3.5.12 Process Failures Modes and Effects Analysis

The developer shall perform a process Failures Modes and Effects Analysis (FMEA) to determine potential product failure modes caused by fabrication processes. A process FMEA shall be performed prior to and support process qualification for new, ordnance, safety critical, and high volume production

processes. Results of process FMEA shall be made available to the cognizant MDA Deputates and Elements and designated representative upon request. As a minimum, process FMEA shall:

a. Identify potential product related process failure modes.

b. Assess potential end user effects of the failures.

c. Identify the potential fabrication process causes and process variables on which to focus controls for occurrence reduction or detection of the failure conditions.

d. Develop a ranked list of potential failure modes, thus establishing a priority system for corrective action considerations.

e. Documents the results of the fabrication process.

3.5.13 Environmental Stress Screening

The developer shall establish and maintain an effective Environmental Stress Screening (ESS) program so that workmanship failures can be identified early and removed from equipment. The program shall include development of ESS profiles based on thermal and vibration surveys and equipment response analyses. As a minimum, power on and performance monitoring shall be performed at two levels of assembly. The program shall consider equipment design, part/component technology, and production fabrication techniques.

The developer shall track effectiveness for each level of screening and establish metrics to support appropriate tailoring of existing screening profiles. To determine the most effective screening profiles, the ESS program shall include feed back of latent and intermittent failures, previously undetected design defects, previously undetected failure modes, and workmanship defects into Failure Reporting Analysis and Corrective Action System. The developer may use Accelerated Life Test results as a baseline for determining initial ESS profiles.

The developer shall document the ESS program in the Reliability, Maintainability, and Availability Program Plan. As a minimum, the program shall address the following:

a. Description of environmental stress types, levels, profiles, and exposure times to be applied.

b. Identification of level (e.g., parts, printed wiring assemblies, subassembly, system, including spares) at which testing will be accomplished.

c. Identification of item performance and stress parameters to be monitored during ESS.

3.5.14 Reliability Qualification Test Program/Demonstration

The developer shall perform Reliability Qualification Testing (RQT) supplemented by analysis on the end item to determine if the specified reliability requirements have been achieved. RQT shall be performed using items representative of the approved operational configuration, to determine compliance with specified reliability requirements. The developer shall establish, maintain, and make available to the cognizant MDA Deputates and Elements or designated representative a RQT Plan, which shall include the following:

a. Test objectives and selection rationale.

b. Identification of the equipment to be tested (with identification of the computer programs to be used for the test, if applicable) and the number of test items of each equipment.

c. Test duration and the appropriate test plan and test environments. The test plans and test environments (if life mission profiles are not specified).

d. A test schedule that is reasonable and feasible, permits testing of equipment, which are representative of the approved operational configuration.

Detailed test procedures shall be prepared for the tests that are included in the RQT Plan. An outline of these tests shall be addressed in the Reliability, Maintainability, and Availability Program Plan, and the details addressed in the Integrated Test and Evaluation Program Plan (3.7.1).

The RQT shall be integrated with the overall system/equipment qualification testing program.

3.5.15 Maintainability Allocations and Predictions

The developer shall allocate quantitative maintainability objectives down to the lowest replaceable unit and ensure inclusion in specifications as design criteria for hardware and diagnostic software. The maintainability objectives shall address servicing, preventive and corrective maintenance in terms of allowable downtime with consideration for required manpower, skill levels, special tools and test equipment, and diagnostic capabilities.

The developer shall perform maintainability prediction early in the design phase and update prediction throughout the development effort. The predicted values shall reflect applicable experience in previous programs and related hardware design and capabilities of support equipment, including diagnostic software. The predictions shall consider and identify pertinent requirements for accessibility and shall consider human factors. Maintainability predictions shall be used in formulating design decisions, maintenance planning, and logistics planning.

3.5.16 Maintainability Analysis

The developer shall perform maintainability analysis on subsystems, equipment, and assemblies to the lowest replaceable unit of assembly. Maintainability analyses shall be performed concurrently with design, and in conjunction with the reliability effort, so that identified problem areas can be addressed for timely consideration of corrective action. Analysis procedures shall include the examination and evaluation of proposed and actual designs, including software, in order to establish the most effective and efficient design for preventive, progressive, and corrective maintenance, and to identify maintenance resource requirements (e.g. repair parts, skills, and equipment). The analysis shall consider the requirements for failure detection and isolation, the extent of built-in test capability, the input and output media, and the results of the reliability analyses. When it is determined by the analysis that a proposed or actual design is deficient in meeting qualitative (e.g. access, space, standardization) or quantitative maintainability objectives, the results of the analysis shall drive additional design or redesign.

3.5.17 Maintainability Demonstration

The developer shall plan and execute a maintainability demonstration to verify compliance of the system to the maintainability requirements of the specification. The developer shall use the reliability predictions and other pertinent considerations to identify and list the most probable anticipated failures of mission critical real-time system functions. From this list, the developer shall identify and scope a group of candidate maintainability demonstration tests from which a selection shall be made of specific tests to conduct prior to deployment.

Maintainability demonstration tests shall verify the capability of the planned maintenance activities to meet the operational availability/mean down times required for identified system functions. The tests shall also verify the adequacy of fault detection or isolation methods and the ability to achieve lowest replaceable unit replacements or on-site repairs to meet criteria stated in the maintenance plan.

The approach and the details of demonstration, including the selection of demonstration personnel, technical manuals, support equipment shall be described in a maintainability demonstration plan that shall

be prepared and made available to the cognizant MDA Deputates and Elements or designated representative upon request. The plan shall describe candidate failure scenarios and identify and outline the test specification requirements of each candidate individual demonstration.

A maintainability demonstration report shall be made available to the cognizant MDA Deputates and Elements or designated representative upon request. The report shall include data collected, results of data analysis, and conclusions and recommendations. In the event of failure to meet specified maintainability objectives, the report shall present the corrective action planned to overcome the deficiencies encountered and a schedule for demonstrating the effectiveness of changes.

3.5.18 Availability Allocations and Predictions

The developer shall allocate availability objectives, including related reliability and maintainability objectives, to minimize total ownership costs and to meet program specifications. Reliability and maintainability objectives shall be derived from and directly support MDA system, subsystem, and assembly availability objectives (inherent availability (Ai) and operational availability (Ao)). The developer shall design for and track both Ai and Ao throughout the life cycle. Ao shall be monitored for deployed equipment when the developer is involved in support and maintenance logistics.

Starting early in the design phase, availability predictions shall be performed for the product and its elements. Availability predictions shall be maintained to reflect the current design. The results of availability predictions shall be used as inputs in formulating decisions for product design, safety, maintenance, logistics, and availability analyses.

3.5.19 Availability Assessment

The developer shall assess availability of the product beginning with the design and test programs and continuing through the operational phase. The assessment process shall incorporate and integrate the results of reliability and maintainability analyses, engineering analyses, testing, valid operating data from previous generations, and applicable test and usage data for the quantitative measurement of product availability.

3.5.20 Reliability, Maintainability, and Availability Metrics

The developer shall establish and maintain a system for the collection and analysis of Reliability, Maintainability, and Availability (RM&A) metrics. As a minimum, the set of metrics and frequency of collection shall be representative of the development effort and all phases of the acquisition process. The RM&A metrics to be collected and analyzed may include: Operational Availability (Ao), Mean Maintenance Man hours (M-MMH), Preventive Maintenance (PM), Mean Time Between Failures (MTBF), Mean Time To Repair (MTTR), Mean Administrative Delay Time (MADT), and Mean Logistics Delay Time (MLDT).

3.5.21 Reliability, Maintainability, and Availability of Government Furnished Equipment/ Information

When the overall system includes components or other elements furnished by the government, the developer shall be responsible for identifying and requesting adequate Reliability, Maintainability, and Availability (RM&A) data on the items. Data shall be used for performing RM&A analyses. When examination of data or testing indicates reliability, maintainability, or availability of GFE/GFI (including COTS/NDI) is inconsistent with RM&A requirements of the overall system or is unavailable, the cognizant MDA Deputates and Elements shall be promptly notified.

3.6 Parts and Materials Control Program

The developer shall establish and maintain a Parts and Materials Control Program (PMCP) to ensure the selection and use of parts, devices, and materials, including commercial and non-developmental items, meet specified performance, quality, reliability, safety, supportability, and configuration management requirements throughout the life cycle of the system. The program shall include provisions for mitigating the impact of parts obsolescence on product integrity. The developer shall prepare a PMCP Plan describing the approach and methodology for implementing a PMCP.

For mission critical hardware, the program shall have a documented approach for the approval, selection, acquisition, handling, packaging, screening, derating, qualification, traceability, standardization, and storage of parts and materials in development and fabrication.

The PMCP is intended to mitigate risk and enhance the probability for mission success for all MDA systems, subsystems and assemblies.

3.6.1 Parts, Materials, and Processes Control Program Plan

The developer shall prepare and maintain a Parts and Materials Control Program (PMCP) Plan in coordination with the cognizant MDA Deputates and Elements, describing the approach and methodology for implementing the parts, and materials control program. The PMCP Plan shall describe the parts and materials selection process, including responsibilities for the evaluation, documentation and notification of part changes, manufacturing processes, and material changes. This includes the process for evaluating performance parameters and the process for evaluating new parts, materials, and processes including qualification and functional verification at next higher assemblies.

The PMCP Plan shall include a description of the methods of qualification used, including description of part evaluation procedures. In addition, the plan shall include flow-down requirements for parts and materials including COTS/NDI, to suppliers for assemblies procured. The plan shall be submitted to the cognizant MDA Deputates and Elements for review and approval.

3.6.2 Parts and Materials Control Board

The developer shall establish and maintain a Parts and Materials Control Board (PMCB) to facilitate management, selection, standardization, and control of parts, material, processes, and associated documentation. The PMCB composition shall include qualified representatives from design, reliability, quality, manufacturing, test, purchasing, and logistics disciplines and include a technical government representative. The PMCB shall allow for supplier representation. The PMCB shall be responsible for the review and approval of parts, material, and processes for conformance to program requirements, and for developing and maintaining a Parts and Materials Selection List for use on MDA programs. In addition, the PMCB shall support all parts and materials failure investigations, and problem resolutions. PMCB findings, decisions, and directions shall be documented and maintained and shall be binding on all applicable developers and suppliers. PMCB operation procedures shall be included as part of the PMCP Plan.

3.6.3 Parts and Materials Selection List

The developer shall establish and maintain a qualified Parts and Materials Selection List (PMSL) for MDA programs. The PMSL is a tool to promote standardization, and shall be maintained to give designers and suppliers visibility of parts preferred for use.

Parts and packaging technology shall be selected based on their intended use considering, but not limited to, performance, environmental, criticality, and lifetime requirements. Only parts evaluated and approved by the Parts and Materials Control Board (PMCB) shall be listed in the PMSL. Risk levels (3.6.12.1) for Electrical, Electronic, and Electromechanical parts shall be identified in the PMSL. The PMCB shall ensure standardization and the maximum use of parts listed in the PMSL. The PMSL shall be updated to

minimize part obsolescence impact. The PMSL and all subsequent revisions shall be made available to the cognizant MDA Deputates and Elements or designated representative upon request.

3.6.4 Diminishing Manufacturing Sources and Material Shortages

The developer shall establish and maintain a Diminishing Manufacturing Sources and Material Shortages (DMSMS) management program to ensure continued availability of high quality and reliable parts, assemblies, and materials for MDA throughout the product's life. DMSMS management procedures shall address proactive forecasting and associated design tradeoffs to minimize life cycle vulnerability. Reactive efforts shall address cost effective solutions for DMSMS events, including obsolescence identified during fabrication or in fielded units.

The developer shall advise the cognizant MDA Deputates and Elements of any actual or potential loss of parts or material sources and document as an advisory in accordance with (3.1.10). Mitigation plans for resolving and achieving solutions to DMSMS impacts as they occur shall be prepared and made available to the cognizant MDA Deputates and Elements or designated representative.

DMSMS program activities shall be included as a regular analysis element of the Risk Management Program (3.1.5).

3.6.5 Parts and Materials Process Monitoring Program

The developer shall establish and maintain a parts and materials process monitoring program which defines the specific control points, specific measurement data to be collected, and the intended use and analysis of each measurement. The program shall apply statistical methods to identify and reduce variability in key characteristics of parts and materials.

3.6.6 Alternate and Substitute Part and Material Selection

The developer shall establish and maintain a system for the selection and qualification of alternative parts and materials. In selecting the part and material replacement, the developer shall ensure the parts and materials are qualified to meet or exceed the functional and performance requirements for the intended application. Once approved by the Parts and Materials Control Board (PMCB), the alternative part or material shall be identified as alternate in the Parts and Materials Selection List (PMSL).

A substitute part or material is a part or material whose performance may be less capable than items specified on a PMSL. For parts or material controlled by a government approved design, a substitute part or material may be used if a waiver (3.10.3.4.1) or deviation (3.10.3.4.2) is granted. For parts or material controlled by supplier approved design, the substitute part or material may be used if approved by the PMCB.

3.6.7 COTS Parts and Materials Management

The developer shall establish and maintain a system to control the selection, evaluation, acceptance, and supportability of COTS parts and materials. COTS management shall be addressed in the Parts and Materials Control Plan.

Additional requirements for use and control of COTS products are found in provisions 3.2.9 and 3.3.2.1.1.

3.6.8 Plastic Encapsulated Microcircuit Qualification

The developer shall demonstrate to the Parts and Materials Control Board (PMCB) that Plastic Encapsulated Microcircuits (PEM) use meets the quality, reliability, environmental, and survivability requirements of the contract end item for the intended application. PEM qualification shall address life cycle environments including synergistic effects of humidity and temperature cycling, through sequential environment exposure. The following tests shall be performed for qualifying PEMs, as a minimum.

a. Pre and post-electrical test per device specification at room, and temperature end points.

b. Preconditioning test. Performed on surface mount devices only and before c, d, and e.

c. Highly Accelerated Stress Testing (HAST).

d. Autoclave.

e. Temperature cycling.

f. High temperature storage.

g. High temperature operating life test.

h. Data retention (memory products).

i. Latch-up (CMOS devices).

The PMCB shall define the appropriate additional qualification tests for PEMs as warranted by the intended environment.

PEMs shall be re-qualified when the manufacturer changes the package material, die size, fabrication process, PEM materials, PEM design, assembly plant, or other relevant factors that make the original qualification data obsolete. PEM qualification procedures shall be included as part of the Parts and Materials Control Program Plan.

3.6.9 Environmental (Lead Free)

The developer shall not use lead free alloys or surface finishes on EEE and mechanical parts utilized in safety (3.14) and mission critical designs. However, for EEE and mechanical parts that do not have safety and mission critical applications, and when lead free is the only available surface finish, the developer shall document the risks and prepare risk mitigation planning. Mitigation planning shall address life cycle environmental qualification tests including temperature cycling and vibration effects. The risk management boards shall monitor the lead free risks and risk mitigation plans.

3.6.10 Parts and Materials Qualification

The developer shall define methods for qualifying parts and materials based on their intended application. Parts and material qualification consists of mechanical, electrical, environmental inspection and test, and analysis. Inspections and tests shall verify that materials, design, performance, and long term reliability of the part and material are consistent with the specification and intended application. Qualification is considered destructive and samples shall be segregated to preclude incorporation into the end item.

When approved by the Parts and Materials Control Board, qualification of parts and materials by usage history or similarity to qualified parts, may be accomplished by any of the following:

a. *History*. A part can be considered qualified if it has been successfully used in (1) identical applications, or (2) a different application where the derating, environmental conditions, and other qualification parameters are fully documented and more severe than the proposed application. The part must have been successfully used for a minimum of 2 years in its intended application. The same manufacturer must have built the part in the same facility using the same materials and processes to an equivalent source control drawing. The developer shall have documented evidence of historical data.

b. *Similarity*. A part can be considered qualified if it is similar to a part for which qualification test data exists, and the test data (1) satisfies the requirements for the intended application, (2) is available and

is less than 2 years old relative to the lot date code of parts used in identical applications. In order to be considered similar, the part shall be made by the same manufacturer on the same manufacturing line.

c. *Existing Test Data*. Parts and materials can be qualified by existing test data, which meets the requirements, specified in its intended application. Lot specific data indicates that mission critical parts have the same lot date code as the qualification samples. Lot specific data is acceptable in place of qualification testing when it meets the MDA system, subsystem, or assembly requirements.

Parts and materials manufacturing process qualification (3.12.6.1) shall be performed to maintain the integrity of the materials and to avoid introducing property changes that could cause adverse effects.

3.6.11 Reuse of Parts and Materials

Parts and materials, which have been installed in an assembly shall not be used again in any mission critical hardware without prior Parts and Materials Control Board approval. Request for reuse shall be based on the submission of evidence that this practice does not degrade the system performance.

3.6.12 Electrical, Electronic, and Electromechanical Parts Program

The developer shall establish and maintain a Electrical, Electronic, and Electromechanical (EEE) parts program to ensure all EEE parts are selected, derated, screened, and qualified to meet or exceed the quality, reliability, environmental, and survivability requirements of the contract end item for the intended application. EEE parts include items such as capacitors, connectors, crystal oscillators, diodes and transistors, fiber optics, filters, fuses, hybrid microcircuits, monolithic microcircuits, magnetics, relays, resistors, thermistors, and wire and cable.

3.6.12.1 Electrical, Electronic, and Electromechanical Parts Selection

Electrical, Electronic, and Electromechanical (EEE) parts selection shall be in accordance with the Parts and Materials Selection List. EEE parts risk levels are described below.

a. *Level 1*: Parts used in mission critical space, flight, missile, and ground applications. Level 1 parts are inherently low risk and are suitable for use in all applications including single-point failure and life support.

b. *Level 2*: Parts used in non-mission critical space, flight, missile, and ground applications. Level 2 parts are inherently moderate risk and not recommended for life support, mission critical, and single-point failure applications.

The inherent risk of the parts selected shall be mitigated to meet application needs by qualification and additional testing (when appropriate). The Parts and Materials Control Board (PMCB) may recommend further mitigation strategies. The PMCB shall approve the risk level selected for the application.

3.6.12.2 Electrical, Electronic, and Electromechanical Parts Derating

The design standards shall include derating provisions to limit the operating stresses of electrical parts. All Electrical, Electronic, and Electromechanical (EEE) parts shall be derated in accordance with criteria for the appropriate environment as defined in SD-18 or a cognizant MDA Deputates and Elements approved alternative. Derating criteria for Plastic Encapsulated Microcircuits in a severe environment shall be determined and approved by the Parts and Materials Control Board (PMCB). Parts stressed beyond their derating criteria, shall be submitted to the PMCB for review and require PMCB approval prior to use. Expected operating temperatures and environmental conditions shall be established and identified by designers.

3.6.12.3 Electrical, Electronic, and Electromechanical Parts Screening

Electrical, Electronic, and Electromechanical (EEE) part screening shall be performed to remove early and potential failures due to latent part defects, nonconforming parts, or workmanship defects. The developer shall ensure that qualified personnel using appropriate equipment and software perform EEE part screening. As a minimum, screening shall consist of 100 % electrical parameter tests, as defined in the component specification, for each device type. Screening shall be performed on mission critical parts in accordance with Parts and Materials Control Board approved military or industry specifications. Particle Impact Noise Detection (PIND) testing will be required on every lot for airborne and space applications.

3.6.12.4 Electrical, Electronic, and Electromechanical Parts Qualification

Electrical, Electronic, and Electromechanical (EEE) parts shall be qualified in accordance with 3.6.10. As a minimum, approved specifications used for qualification shall include test requirements for:

a. Hermetic and hygroscopic nature of unique package types.

b. Operating characteristics over entire temperature range.

c. Packaging capability for handling thermal shock.

d. Internal circuitry and connection resistance to contamination and corrosion (passivation).

e. Internal connection fatigue life.

f. Levels of inherent contamination in packaging.

g. Solderability of leads.

h. Electrostatic Discharge.

3.6.12.5 Destructive Physical Analysis

Destructive Physical Analysis (DPA) shall be used as a tool for evaluating suppliers whose product quality is unknown and as a root cause diagnostic tool for component failures. DPA shall also be used:

a. To baseline the construction of the component.

b. To analyze part degradation after test flows.

c. As a tool for monitoring supplier changes.

d. To look for construction problems on a lot-by-lot basis by sampling each lot date code of microcircuits, hybrids, semiconductors, relays and filters.

e. On all other parts with failure history, GIDEP Alerts, or other concerns.

Variations to the DPA sample size due to part complexity, availability, or cost shall be approved by the Parts and Materials Control Board on a case-by-case basis.

For mission critical applications five devices from each lot date code of received electrical and electronic devices (including connectors) shall be held for DPA. These parts shall be retained and shall be available for a period of four years.

3.6.12.6 Parts Age Control

Parts drawn from controlled storage after four years from the date of the last full screen shall be subjected to a part verification screen and sample DPA. As a minimum, the part verification screen shall consist of a visual examination, electrical measurements at 25 degrees C, and hermeticity testing as applicable. The Parts and Materials Control Board (PMCB) shall approve the specific screening requirements for each part type.

Parts drawn from controlled storage after eight years from the date of manufacture shall not be used without PMCB approval. If usage of these parts is approved, as minimum, the parts shall be subjected to a full rescreen and sample DPA. Parts stored in other than specified conditions, which affect aging shall not be used without prior approval of the PMCB. The details for parts age control shall be addressed in the Parts and Materials Control Program Plan.

3.6.12.7 Radiation Hardness and Survivability Assurance

All parts requiring radiation hardening and Nuclear Hardening and Survivability (NH&S) (3.2.17) shall be identified as Hardness Critical Items (HCI). All parts shall be selected to meet their specified intended application in the predicted mission radiation and nuclear environments. As applicable, the developer shall consider radiation and nuclear effects such as, total ionizing dose, neutrons, prompt dose rate, and single-event phenomena. The developer shall document the analysis for each HCI with respect to each effect considered. HCIs shall be tracked throughout the life cycle to ensure no degradation to radiation hardening and NH&S has occurred due to changes in manufacturing processes & materials by the vendor.

The developer shall ensure that this requirement is flowed down to suppliers, and Department of Energy certified labs in accordance with 3.13.4.

3.6.12.8 Traceability and Lot Control

The developer shall develop and maintain traceability and lot control for all Electrical, Electronic, and Electromechanical (EEE) parts, including COTS, in accordance with the requirements contained in paragraph 3.10.2 and as specified below. Traceability to the serial number of an individual device or to a lower level of assembly shall be determined and specified by the Parts and Materials Control Board. Identification and serialization data for EEE parts shall be maintained in the manufacturing and processing records and contain lot date code, lot and purchase order numbers, and manufacturer of the part. The developer shall ensure markings for small devices are recorded in the manufacturing and processing records prior to use.

3.6.12.9 Custom Devices

In addition to the screening and qualification requirements, any custom microcircuits, hybrid microcircuits, Multi-Chip Modules (MCM), or Application Specific Integrated Circuits (ASIC) planned for use shall be subjected to an internal review process as part of the Parts and Materials Control Board activity. The review shall address, at a minimum, derating of elements, method used to ensure each element reliability, assembly process and materials, firmware configuration, and method for assuring adequate thermal matching of materials.

3.6.13 Failure Analysis

As required by the Parts and Materials Control Board/Failure Review Board, failure analysis shall be performed on part and material failures experienced during inspection, assembly, test, and field operations. Failures shall be analyzed to the extent necessary to understand the failure mode and root cause, to detect and correct out-of-control processes, and to determine the necessary corrective actions. The developer shall review the failure analysis and corrective action and determine lot disposition for each part failure. Failure data shall be periodically reviewed for trends. When trends are identified,

corrective action shall be initiated. All failures, diagnostic data, results of final failure analysis, and corrective actions shall be documented in the Failure Reporting, Analysis, and Corrective Action System (FRACAS). Failure analysis reports shall be retrievable and made available to the cognizant MDA Deputates and Elements or designated representative upon request.

3.6.14 Material and Finishes Selection

The developer shall establish and maintain a system for selecting materials and finishes, which assures that they meet or exceed end item requirements. In order to anticipate and minimize material problems when selecting materials and finishes during development, fabrication, and operation, the developer shall consider potential impacts related to: radiation effects, thermal cycling, stress corrosion cracking, galvanic corrosion, hydrogen embrittlement, lubrication, contamination of cooled surfaces, composite materials, atomic oxygen, useful life, vacuum out-gassing, toxic off-gassing, flammability, and fracture toughness, as well as the properties required by each material usage or application.

3.6.14.1 Material Outgassing

Based on each component operating conditions and impact on the mission, materials shall be screened for outgassing. Established material outgassing data shall be verified and made available to the cognizant MDA Deputates and Elements or designated representative upon request.

3.6.14.2 Thermal Vacuum Bakeout

The developer shall perform thermal vacuum bakeouts of contamination sensitive hardware. The parameters of such bakeouts (e.g., temperature, duration, outgassing requirements, and pressure) must be individualized depending on materials used, fabrication environment, and established contamination allowance. Thermal vacuum bakeout results shall be verified and made available to the cognizant MDA Deputates and Elements or designated representative upon request.

3.6.15 Storage and Handling/Material Protection

Storage and Handling/Material Protection procedures shall be established and maintained to prevent part and material degradation. These procedures shall be used throughout inspection, storage, kitting, and assembly and identified on fabrication documentation. Storage and handling/material protection of parts and materials shall be in accordance with 3.12.5.2 and 3.12.10.2.

This Page Intentionally Left Blank

3.7 Integrated Test and Evaluation Program

The developer shall establish and maintain an integrated test and evaluation program to ensure that hardware and software meet mission specifications and requirements. The integrated test and evaluation program shall ensure hardware, software, interface, and interoperability capabilities are validated and qualified against requirements identified in the item's configuration documentation. The requirements traceability and verification matrix (3.2.10) shall establish traceability between item requirements and the tests used as a basis for validation and qualification decisions. Test programs to be conducted during fabrication and deployment shall be planned, developed, and, as appropriate, exercised during the development phase so that seamless transitions occur. The developer shall develop and maintain integrated test and evaluation program policies, organizational responsibility, and implementing procedures to ensure:

a. Effective use and control of test resources including the control, maintenance, and reuse of test data. Test records, including data, shall be maintained as part of the MIS (3.1.4).

b. Establishment and evaluation of objectives, plans, and schedules, including requirements for test documentation, test facilities, test equipment, and test samples. The developer shall establish uniform test program requirements, guidelines, and instructions for use in test planning.

c. Methodologies and strategies for testing commercial and non-developmental items to ensure that the items meet functional and performance characteristics and that those characteristics are retained in the procured items.

d. Development and approval of test plans and procedures.

e. Review and evaluation of test results to determine appropriate action, if any.

The Integrated Test and Evaluation Program includes:

a. Engineering Evaluation Tests.

b. Qualification Tests.

c. Acceptance Tests.

d. Production Assessment Tests.

e. Surveillance and Service Life Evaluation Tests.

f. Flight Tests.

These tests are discussed in the following paragraphs. In addition, the overall test and evaluation program includes other types of tests that are discussed elsewhere in this document. These tests include those aimed at addressing reliability [such as Reliability Growth Testing (3.5.10), Reliability Qualification Testing (3.5.14), and Accelerated Life Testing (3.5.11)]; tests related to maintainability (3.5.17), tests associated with software development (3.3.2), and tests related to safety (3.14).

3.7.1 Integrated Test and Evaluation Program Plan

The integrated test and evaluation program shall be described in an Integrated Test and Evaluation Program Plan, which shall be submitted to MDA RTO for information and the cognizant MDA Deputates and Elements for approval. The Integrated Test and Evaluation Program Plan shall be expanded in detail as product and process development progresses. Planning for hardware, software, and system integration testing shall be included in the Integrated Test and Evaluation Program Plan to ensure

effective control and use of test resources and complete coverage of testing activities throughout the product's life. The plan shall:

a. Describe the organization and management of the integrated test and evaluation program.

b. Include a summary of tests, including the type of test, test level, and the test objective.

c. Include schedules for tests, relating test program milestones to major program milestones.

d. Include schedules for special test facilities/equipment, test items, and test documentation.

3.7.2 Engineering Evaluation Tests

The developer shall perform engineering evaluation tests to: mitigate risk; perform design and development verification (3.2.11); determine sensitivity of the design to varying levels, combinations, and sequences of electrical, mechanical, and environmental stress; and verify acceptable levels of design and performance margins. These tests may be performed on prototype hardware/software and assemblies/subsystems as seen appropriate. Final engineering evaluation tests, used for design and development verification purposes, shall be performed on items that are representative of deliverable hardware and software configurations at the highest assembly levels practical to verify functional compatibility and assess interface interactions at the level tested. A test plan (3.7.9), test procedures (3.7.10), and a test report (3.7.11) shall be prepared for each design verification test and made available to the cognizant MDA Deputates and Elements or designated representative upon request.

3.7.2.1 Integration Tests

The developer shall perform integration tests on items that are representative of deliverable hardware and software configurations to verify functional performance and interfaces (e.g., mechanical, electrical, and optical) meet system requirements. These tests shall reflect a systematic, documented method for verifying interface and functional compatibility. Integration tests shall also verify the test equipment to unit under test interfaces while proofing the test procedures.

3.7.2.2 Interoperability Tests

The developer shall conduct or support tests to demonstrate compliance with interoperability requirements. This activity shall include testing at various maturity levels and levels of integration/assembly to establish confidence prior to integrated ground and field/flight tests. Requirements related to interface requirements, data definition, timing, scale factor compatibility, and error reporting and handling shall be verified through thorough integrated testing at various levels. Verification activities should be closely coupled with systems engineering to ensure the implementation of meaningful verification activities. Collectively, these interoperability tests demonstrate compliance with specified interoperability certification criteria.

3.7.2.3 Test-Like-You-Fly Approach

The developer shall perform engineering evaluation tests using a test-like-you-fly approach. This approach ensures ground support equipment and hardware/software units under test are evaluated in a configuration matching, as close as possible, to the expected operational configuration and environment. The support equipment, hardware, and software configurations shall be well defined, documented, under configuration management control, and known to the test engineers. Test setups shall be configured to provide expected operational scenarios and operational environments with margin. Final engineering evaluation ground tests conducted prior to flight testing shall be performed with the flight software in an operational ("non-test") configuration to reflect the actual software execution paths to be exercised during the flight. Any non-flight-representative exceptions to the flight hardware/software configuration or test environment and their impact on the test objectives shall be identified prior to test conduct and approved by the test readiness review panel.

3.7.3 Qualification/Re-Qualification Test Program

The developer shall establish and maintain a program for the qualification and requalification of hardware and software (3.3.2.6). The qualification program shall ensure that all system elements meet their specification requirements when placed in the operational environment. A requalification decision, including supporting rationale, shall be approved by the cognizant MDA Deputates and Elements whenever any of the following occur:

a. Change in hardware or software design.

b. Change in supplier.

c. Change in manufacturing processes or plant location.

d. Interruptions of the manufacturing processes greater than six months.

e. Increases in the manufacturing rate.

f. Disqualification of a product.

g. Changes to equipment, procedures, or software used to test the qualified product.

3.7.3.1 Qualification Program Plan

The developer shall prepare, maintain, and submit a Qualification Program Plan to the cognizant MDA Deputates and Elements for approval. The Qualification Program Plan shall:

a. Describe the organization and management of the qualification program including applicable policy statements and management directives.

b. Identify the system elements to be qualified, including reference documentation and qualification methods. Items considered qualified by virtue of previous qualification shall be identified with supporting justification.

c. Describe the entrance and exit criteria for qualification test requirements, test locations and equipment, and system element qualification.

d. Include schedules for preparing qualification test plans, procedures, and qualification for each system element.

3.7.3.2 Qualification Tests

Qualification tests shall be performed on configuration items to demonstrate system requirements have been met and ensure that associated procedures and processes for fabrication, test, and inspection are satisfactory. Qualification tests shall be performed on samples that are consistent with deliverable hardware and software configurations. Qualification tests on items procured from different suppliers shall include samples from every source of each configuration. When a family of items is being qualified, the qualification test specimens shall include a sampling of the full range of values being considered to satisfy design requirements. Environmental qualification tests shall be performed in accordance with test methods found in Mil-STD-1540 for space systems and in MIL-STD-810 for airborne and ground systems. Qualification tests shall be conducted to the most severe stress levels, with margins, identified in system, subsystem, and software specifications to ensure they fully envelop the expected operational levels, sequences, and combinations of stresses. The test facilities shall be capable of providing the required range of operational demands and environmental levels. The mission profile environments and qualification test environments shall be modified to reflect field test data as it becomes available. The qualification test environment shall simulate the operational environment. When qualification tests are conducted at locations other than the developer facilities, the developer shall ensure establishment of

controls over the supplier's test program that are equivalent to those for tests conducted within the developer facilities.

Test plans (3.7.9) and test reports (3.7.11) shall be prepared for each qualification test and submitted to cognizant MDA Deputates and Elements for approval and MDA RTO for information. Test procedures for each qualification test shall be prepared, maintained, and made available to the cognizant MDA Deputates and Elements or designated representative upon request.

The developer shall evaluate qualification test results to assess complete test coverage and conformance to expected results.

3.7.3.2.1 Qualification by Similarity

Qualification by similarity may be acceptable when the hardware has met all of the following conditions:

a. Used in a similar application in the intended environment.

b. Based on similar functional characteristics and was tested to stress levels at least as severe as those specified for the part to be qualified.

c. Tested under program controls commensurate with those imposed on the qualification test program.

d. Manufactured by the same supplier using similar processes, materials, and quality control; and used in a similar application.

Decisions to qualify based on similarity shall be documented, including detailed engineering justifications, and shall be subject to cognizant MDA Deputates and Elements approval.

3.7.4 Acceptance Tests

The developer shall conduct acceptance tests to demonstrate the ability of deliverable items to meet specification requirements. Requirements for the acceptance test equipment shall be derived from allocated design requirements for the particular unit being tested. Whenever possible, acceptance tests shall environmentally stress the hardware to the maximum conditions expected for all operational events, including transportation and handling. Acceptance tests shall be performed incrementally starting at the component and subassembly levels and progressing to the major assembly and system levels to assure that items are tested at points, which assure that the acceptability of each item characteristic is completely verified. Acceptance tests shall be performed consistent with fabrication and quality (3.12) plans and procedures.

Final system-level acceptance tests shall be performed prior to units being shipped for field/flight testing. During a developmental flight test program, problems identified during acceptance testing of the unit to be flown shall be identified and the impact reviewed with the cognizant MDA Deputates and Elements prior to the decision to proceed with field/flight testing.

3.7.5 Production Assessment Tests

For programs that have a production phase, selected parts, devices, materials, and assemblies shall be sampled and a production assessment test performed to assess whether production process changes are occurring that have a detrimental effect on the completed product. These tests shall thoroughly evaluate selected characteristics including the application of appropriate stress levels to ensure continued compliance with design criteria. A Production Assessment Test Program Plan shall be developed, coordinated with the cognizant MDA Deputates and Elements, and maintained to indicate the selection process, products selected, and the production assessment tests that will be conducted to verify that the required performance, quality, reliability, and safety aspects of the product are maintained. Criteria for selecting parts, components, or assemblies for production assessment testing shall be based upon:

a. Susceptibility to environmental conditions.

b. Effect on mission.

c. Normal process variability relative to specified tolerances.

d. Sensitivity to changes in processing variables.

e. Complexity of the manufacturing or production process.

f. Production quantity and duration.

If an item is produced on more than one processing line or procured from more than one source, sample selection shall cover all lines or sources. The nature of the tests, number of test samples selected for each assessment, and frequency of test shall be compatible with the complexity of the production process and its controls.

Test plans (3.7.9) and procedures (3.7.10) shall be established and maintained for the control of each production assessment test. The results of each production assessment test shall be documented in a test report (3.7.11), which shall be made available to the cognizant MDA Deputates and Elements or designated representative upon request. Actions shall be initiated when data indicates degradation in quality, reliability, or safety of the product or the processes used to fabricate or produce it.

3.7.6 Surveillance and Service Life Evaluation Tests

The developer shall support or conduct life cycle surveillance and accelerated service life evaluation testing program. Surveillance and accelerated service life tests are performed on selected completed items so that timely management decisions can be made to maintain system reliability and operational readiness. These tests shall be performed to: anticipate and detect aging and degradation of items; assess the effects of operational and storage environments on the product's quality, reliability, safety, and service life; establish controls for items that are calendar age, operating time, or cycle-time sensitive; ascertain aging or environmentally induced trends, service life limits, and other issues affecting life cycle reliability and operational readiness. Test plans (3.7.9), test procedures (3.7.10), and test reports (3.7.11) for each surveillance and service life evaluation test shall be prepared, maintained, and made available to MDA Systems Engineering & Integration, the cognizant MDA Deputates and Elements, and designated representative upon request for each conducted test.

3.7.7 Ground and Flight Tests

The developer shall support ground and flight testing to execute, demonstrate, and characterize critical aspects of the performance of the system, its subsystems and its interfaces. Ground and flight test activities include range testing of land, air, sea, or space based systems. Testing shall be performed not only to demonstrate performance of the product but also its interoperability with other affected MDA systems. Ground and flight tests shall be conducted in accordance with cognizant MDA Deputates and Elements approved test plans (3.7.9) and test procedures (3.7.10). A test report (3.7.11), including a detailed post-test performance analysis, shall be prepared, maintained and submitted to the cognizant MDA Deputates and Elements for approval and MDA RTO for information. In the event of ground and flight test failures, the developer shall convene a formal failure review board (3.5.4). Preparations for ground and flight testing and post-test activities are described in the following sections.

3.7.7.1 Critical Test Gate Process

The developer shall conduct a series of informal working-level data reviews that gate critical steps in the build-up and check-out of ground and flight test units to avoid premature flight testing of hardware and software. These working-level review meetings shall include representatives from all affected engineering, test, and safety organizations and allow for participation by cognizant MDA Deputates and Elements or designated representatives. If testing is done too soon and the flight design subsequently

changes, it could result in the need to retest hardware or the temptation to skip steps in the rebuild process. Repeated retesting can result in over-test of flight hardware, which could lead to an associated reduction in reliability and subsequent increase in the risk of flight test failure.

The developer shall conduct test process reviews with specific predetermined exit criteria at "gating" points in the test asset build-up process to ensure adequate proofing of test equipment and test software prior to subjecting actual flight hardware to the associated ground test. Of particular concern are tests that involve stressing environments. Test processes that verified proper performance of hardware/software (such as qualification tests, section level acceptance testing, electromagnetic vulnerability testing, and live-battery testing) shall also be reviewed (along with their results) prior to flight unit build-up/test to ensure that the flight configuration is final and will not subsequently change prior to the mission (thereby requiring rebuild/retest). The developer shall be accountable for tracking progress through this "gate" process and ensuring that the impact of any "liens" incurred through this process are understood, documented, and resolved prior to flight.

3.7.7.2 Post-Test Performance Analysis

The developer shall conduct a comprehensive post-test evaluation addressing all aspects of the mission conduct and including not only specification-compliance of the system but also system robustness and margin. Results that are analyzed shall include all data from command and control systems operations, system performance during ground and flight tests, performance of all associated subsystems, target, and ground support equipment operation. Analysis shall verify that environments (e.g., shock, acoustics, and loads) are within analysis expectations. Any out-of-family performance, anomalies, and nonconformances for the test unit and critical ground systems shall be identified and assessed. A mechanism shall be implemented for capturing lessons learned (3.1.4) during post-test reviews for future use.

3.7.7.3 Failure Review Process

In the event of failures (ground or flight equipment) during government scheduled tests, the developer shall convene a formal failure review process and board with technical government representation to ensure high-level management concurrence with the identified root cause and corrective action. A failure review team consisting of contractor and government technical experts shall be assembled to investigate all failures. Full fault trees shall be defined by the team early in the process. All branches of the tree shall be investigated and closed only after persuasive technical justification has been reached. Fault insertion testing shall be performed if possible to demonstrate that all aspects of the failure can be reproduced by the most-likely failure mode. For flight failures, closure actions shall be approved by a top level management, MDA level Failure Review Board (FRB), which includes experienced individuals from both supplier and government organizations. Corrective action shall be defined and approved by the MDA FRB prior to design implementation. If all fault tree branches cannot be ruled out, corrective action to address all remaining potential fault mechanisms shall be implemented.

3.7.8 Modeling and Simulation

The developer may use Modeling and Simulation (M&S) (3.2.14) prior to, during, and after the completion of ground and flight tests, as a method of demonstrating critical aspects of the performance of the system, subsystems, and interfaces. Models and simulations shall be verified, validated, and accredited. M&S outputs shall be reported, correlated, and validated against actual test data to increase confidence levels, reduce test costs, and support government evaluation decisions. Verification is the process of determining that a model or simulation implementation accurately represents the conceptual description and specifications. Validation is the process of determining the degree to which a model or simulation is an accurate representation of the real world from the perspective of the intended uses. Accreditation is the formal certification that a model or simulation is acceptable for use for a specific purpose.

3.7.9 Test Plans

The developer shall prepare and maintain a test plan for each test required by the integrated test and evaluation program. Test plans shall include, as a minimum:

a. Identification of the item and quantity to be tested.

b. Test objectives.

c. Test requirements, including parameters to be measured, environments to be simulated, test time, facilities, test and measurement equipment, and software.

d. Requirements for data collection, analysis, and reporting.

3.7.10 Test Procedures

The developer shall prepare and maintain procedures for each test required by the integrated test and evaluation program. Test procedures shall include, as a minimum:

a. Characteristics to be tested or measured, including tolerances.

b. Input and range of test parameter values, including tolerances.

c. Identification of test and measuring equipment, tools, jigs, fixtures, recording equipment, and supporting software.

d. Identification of special equipment or facilities.

e. Method to be used in test performance, including sequential steps.

f. Verifications to be made before conduct of test.

g. Instructions for data recording.

h. Actions to be taken in the event of test interruptions.

i. Pass or fail criteria.

j. Applicable safety precautions for personnel and facility protection.

k. Diagram or detailed description of the test setup such as interconnection information, relative equipment placement, mounting of sensors, and grounding points.

l. Parts, devices, and material protection requirements.

3.7.11 Test Reports

Developer shall document and retain test data, including test conditions, significant events, and problems in a report. Deviations from required test equipment configuration, test item configuration, and test environment shall be documented and reconciled.

Test reports shall include, as a minimum:

a. A reference to the applicable test plan and procedures.

b. Copies of waivers, deviations, engineering change requests, and failure reports pertaining to the test.

c. Identification of significant events, problems, and any variances from the test procedure.

d. Identification of the specific test equipment used including the due date for its next calibration.

e. Results of data analysis, failure diagnosis, conclusions, and recommendations.

f. Reconciliation of test item configuration and the items configuration baseline.

g. Reconciliation of actual test environment with the planned test environment.

h. Reconciliation of the environmental test stress state with the functional accumulated stress state.

3.8 Test, Measuring, and Diagnostic Equipment and Standards

The developer shall establish and maintain a system for the definition, selection, design, evaluation, approval, maintenance, calibration, use, and control of Test, Measuring, and Diagnostic Equipment and Standards (TMDES) necessary to verify adequacy of processes and product conformance during all phases of the program. The system shall comply with the requirements of ANSI/NCSL Z540-1-1994, Calibration Laboratories and Measuring and Test Equipment – General Requirements. Test, measuring, and diagnostic equipment and standards includes: test and inspection equipment, test support equipment, gages, and equipment used to monitor and control production processes. TMDES also includes, production tools, jigs, fixtures, and personally-owned measuring equipment used to measure, test, verify, calibrate, diagnose, or otherwise examine materials, supplies, and equipment to determine compliance with product and process specifications.

3.8.1 Selection and Design

The developer shall establish and maintain a system for the selection, design, and evaluation of Test, Measuring, Diagnostic Equipment (TMDE), including related software, used to verify conformance to product and process specifications. The selection and design system shall give preference to selection of Commercial-Off-The-Shelf equipment, standards, software, fixtures, cables, and materials rather than items that are unique or proprietary. TMDE shall have the accuracy, range, resolution, repeatability, reliability, and stability required to provide measurement accuracy of at least 20 percent of the tolerance of the characteristic being tested or measured. For single limit parameters, the required accuracy shall be specified. If measurement or calibration accuracies cannot be achieved due to technology limitations, a variance (3.10.3.4) shall be submitted to the cognizant MDA Deputates and Elements for approval.

3.8.1.1 Test, Measuring, and Diagnostic Equipment Configuration Documentation

The developer shall develop and maintain configuration documentation for uniquely designed Test, Measuring, and Diagnostic Equipment (TMDE) items and test stations. The maximum permissible variation among test stations shall be specified in its configuration documentation. The configuration documentation package shall be controlled in accordance with 3.10 and shall include, as a minimum:

a. Calibration procedures.

b. Operating instructions.

c. Programming instructions.

d. A Measurement Accuracy Report.

e. Configuration documentation for the TMDE including, top assembly drawings, interface control drawings, lower level design drawings, and specifications.

f. Configuration documentation for any developed software. This includes source codes for, but is not limited to, operating system, test executive, test, calibration, diagnostics, test libraries, and instrument drivers.

3.8.1.2 Evaluation of Test, Measuring, and Diagnostic Equipment

Test, Measuring, and Diagnostic Equipment (TMDE) used to verify conformance to product and process documentation shall be evaluated to ascertain the item will provide the inputs, loading, and measurement capabilities required. Evaluation shall consist of a two-part effort: preliminary uncertainty analysis and verification testing.

a. The preliminary uncertainty analysis shall compare proposed TMDE capabilities with product or process parameter tolerances. The TMDE capabilities shall be based on equipment uncertainty

information specified by manufacturers of commercial equipment, data available from previously used equipment, and engineering estimates for new design equipment. TMDE uncertainty for each input, load, and measurement shall be compared with each respective specification tolerance to calculate accuracy ratios. For new TMDE designs, the preliminary uncertainty analysis shall be performed concurrent with the release of the design.

b. Verification testing shall be conducted on the first unit of TMDE containing a new design to determine inherent uncertainties and to verify uncertainties that cannot be verified in the preliminary uncertainty analysis. Verification testing shall also be conducted on commercial equipment where the preliminary uncertainty analysis was inconclusive. Verification testing shall be done under required environmental operating conditions. These tests shall be of sufficient scope and duration to demonstrate compliance with accuracy and repeatability requirements.

The TMDE uncertainty obtained from any verification testing shall be used to complete the uncertainty analysis. Records shall be kept of all uncertainty analyses and verification testing results and made available to the cognizant MDA Deputates and Elements or designated representative upon request.

3.8.1.3 Proofing, Qualification, and Correlation

The uniquely designed Test, Measuring, and Diagnostic Equipment (TMDE) or test stations shall be proofed and qualified to ensure the effectiveness in measuring the product's compliance to configuration documentation. Proofing and qualification (3.7.3.2) shall be performed under actual operating and environmental conditions to verify completeness and adequacy of the equipment, software, material, personnel training, and documentation related to station calibration, station maintenance, and product verification. In addition, proofing and qualification shall include, but not be limited to, verification of the supporting test disciplines covering test performance, calibration performance, maintenance performance, environmental controls, configuration controls, security, and safety. The developer shall conduct accuracy, reproducibility, repeatability, and trend analysis to assess the uniformity, consistency, and stability of the TMDE or test station. TMDE or test stations shall be re-proofed and re-qualified if changes or modifications affect functionality and usage. The developer shall notify the cognizant MDA Deputates and Elements and designated representative of operational proofing and qualification events to allow for government participation. Operational proofing and qualification results shall be documented and submitted to the cognizant MDA Deputates and Elements for review prior to use.

The developer shall perform correlation to detect and correct conditions contributing to significant variation in results among test stations. Methods and results for each correlation analysis shall be documented and made available to the cognizant MDA Deputates and Elements or designated representative upon request.

3.8.2 Calibration and Maintenance

The developer shall establish and maintain a system for calibration and maintenance of Test, Measuring, and Diagnostic Equipment and Standards (TMDES) that is in compliance with ANSI/NCSL Z540, and as supplemented by the following.

3.8.2.1 Calibration and Maintenance Procedures

The developer shall document and maintain procedures for calibration and maintenance of Test, Measuring, and Diagnostic Equipment and Standards (TMDES). Calibration and maintenance procedures shall be made available to the cognizant MDA Deputates and Elements or designated representative upon request. Calibration and maintenance procedures shall specify or contain the following in addition to ANSI/NCSL Z540:

a. Description of preparations that must be made before calibration is started.

b. Descriptions or diagrams of the equipment setup, as necessary.

c. Environmental conditions required and stabilization period.

d. Step-by-step instructions for performing calibration and maintenance activities.

e. Data to be recorded, reports or certificates to be prepared, and method of analysis.

3.8.2.2 Records and Analysis

Records shall be retained for the calibration and maintenance system. In addition to the ANSI/NCSL Z540 records and analysis requirements, data shall also be recorded documenting the condition of nonadjustable or fixed-value equipment. Calibration data shall be analyzed in order to identify trends indicating deterioration and to provide for revision of intervals to ensure continued accuracy and reliability of Test, Measuring, and Diagnostic Equipment and Standards, where reliability is defined as the probability that the item will remain in-tolerance throughout the established interval. Individual records used to maintain the calibration system shall be made available to the cognizant MDA Deputates and Elements or designated representative upon request.

3.8.2.3 Out-of-Tolerance Conditions

As a supplement to ANSI/NCSL Z540, the developer shall remove and segregate, where practical, nonconforming Test, Measuring, and Diagnostic Equipment and Standards (TMDES) from service. When TMDES are found to be nonconforming during calibration, an analysis shall be performed to determine the impact on product and the need for subsequent corrective action. When the analysis indicates that the nonconformance could recur, corrective action shall be performed immediately to identify and correct root cause of the problem. The developer shall maintain a record of the out-of-tolerance condition, significance of the nonconformance, and the corrective actions taken. The cognizant MDA Deputates and Elements shall be informed of occurrences affecting MDA products and a record documenting the event, subsequent analyses, and any actions taken shall be made available.

3.8.2.4 Calibration Standards and Reference Materials

Calibration standards and reference materials used for calibrating Test, Measuring, and Diagnostic Equipment and Standards items shall have the accuracy, stability, range, and resolution required for the intended use. The collective uncertainty of the individual or grouping of calibration standards and reference materials shall not exceed 25 percent of the acceptable tolerance for each characteristic being calibrated. Calibration standards and reference materials held by the laboratory shall be used for calibration or verification of working level TMDE only and for no other purpose, unless it can be demonstrated that their performance as standards has not been invalidated. Requests for variance (3.10.3.4) from these requirements, with supporting justification, shall be submitted to the cognizant MDA Deputates and Elements for approval.

3.8.3 General Test, Measuring, and Diagnostic Equipment and Standards Requirements

The developer shall establish and maintain a system to control the usage of Test, Measuring, and Diagnostic Equipment and Standards in compliance with ANSI/NCSL Z540 and the following paragraphs. The system shall provide for accountability for usage both inside and outside the calibration laboratory and shall ensure accuracy and integrity of the resulting measurements and test data.

3.8.3.1 Intervals and Recall

The developer shall calibrate and maintain Test, Measuring, and Diagnostic Equipment and Standards (TMDES) at periodic intervals established on the basis of stability, purpose, and degree of usage. The developer shall establish and maintain an interval adjustment system that is based upon a verifiable statistical methodology appropriate for the type of equipment being controlled. Calibration intervals may be lengthened when the results of preceding calibrations provide definite indications that such action will not adversely affect confidence in the accuracy and reliability of TMDES. The system for establishing

calibration intervals, adjustments, and subsequent revisions, including reliability targets shall be made available to the cognizant MDA Deputates and Elements or designated representative upon request.

The developer shall provide for the mandatory recall and calibration of TMDES within established intervals. The recall system shall provide for accountability information, current calibration status, location and identification of all TMDES, and indicate when all items are due for calibration, service or functional check, or out-of-tolerance condition.

3.8.3.2 Labeling

Calibration labels, traceable to the calibration organization and to the item's calibration record, shall be affixed to Test, Measuring, and Diagnostic Equipment and Standards (TMDES). The developer shall establish traceability between TMDES items and their respective calibration records. When a test or inspection station is calibrated as a single unit, a label shall be affixed to the console frame or similar permanent, common item. TMDES not requiring calibration shall be clearly identified as such. The developer shall ensure access and use of labels is controlled and restricted to authorized personnel.

3.8.3.3 Sealing for Integrity

Tamper-resistant seals shall be affixed to operator accessible controls or adjustments affecting calibration of Test, Measuring, and Diagnostic Equipment and Standards items or test stations. For equipment with removable covers, any internal controls or adjustments are also considered to be operator accessible. Seals shall be designed to destruct on entry. If the seal has been broken, lifted, or is otherwise suspect, the item shall be considered suspect and treated as if the calibration is void. Cabinets, consoles, doors, access covers, and equipment cases may be secured and sealed in lieu of sealing individual equipment controls and adjustments or test station components provided operator accessibility is prevented. The developer shall establish and maintain a system for ensuring that the access to and use of seals is controlled and restricted to authorized personnel.

3.8.3.4 Removal of Test, Measuring, and Diagnostic Equipment and Standards

Test, Measuring, and Diagnostic Equipment and Standards (TMDES) not calibrated and maintained in accordance with the established intervals shall be physically removed from service, where practical, or appropriate tags shall be attached. TMDES items found with broken calibration seals or suspected to be malfunctioning because of mishandling, damage, misuse, or unusual results shall be removed from service or tagged and controlled to prevent further use.

3.8.3.5 Test Station Logs

The developer shall establish and maintain test station logs to record station history including: station operational proofing, calibration of equipment, seal integrity, equipment servicing and replacement, explanations for modifications and breaks-of-station, and any other pertinent information on unusual events or circumstances. Log entries shall be signed or otherwise traceable to the person making the entry and shall include date and time of the event.

3.9 Interface Management

The developer shall establish and maintain an interface management program that assures effective system integration, interoperability, accountability, and timely dissemination of related changes. The interface management program shall establish methods for identification, controlling, verification and flow down of interface requirements found in documents, drawings, software, data, and other Technical Data Package (TDP) elements. The program shall also control or address interfaces in subcontracted items, Government Furnished Equipment or items, and facilities.

3.9.1 Interface Control Plan

The developer shall establish and maintain an Interface Control Plan (ICP) that assures integration, interoperability, and compatibility of all mechanical, electronic, electrical, optical, software, data interfaces, and other MDA system interfaces, as applicable. The ICP shall define the requirements to manage and control mechanical, electronic, electrical, and optical Interface Control Documents and Drawings (ICD), software Interface Design Descriptions (IDD), Interface Requirement Specifications (IRS), data interfaces, including BMDS, MDA element, and system interfaces. The ICP shall clearly delineate interface responsibilities flowed down to suppliers. The ICP shall be made available to the cognizant MDA Deputates and Elements or designated representative upon request.

3.9.1.1 Interface Control Plan Development

The developer in coordination with MDA Systems Engineering & Integration and the cognizant MDA Deputates and Elements shall develop the Interface Control Plan (ICP), identifying roles and responsibilities for each activity involved in the interface management system including schedule and milestones for completion of ICP activities. Additionally, the ICP shall contain the following elements:

a. *Identify Internal and External Interface Requirements.* When identifying internal and external interfaces the developer shall take a top down approach from the most generic to the most specific. Interfaces shall be classified as either internal or external. Internal interfaces are defined by mechanical, electronic, electrical, and optical Interface Control Documents and Drawings (ICD), and software Interface Requirement Specifications (IRS), and Interface Design Descriptions (IDD) that are internal to each individual system. Internal interfaces are synonymous to correlation interfaces. External interfaces are defined by mechanical, electrical, and optical ICDs, software IRSs, and IDDs that control interoperability, interchangeability, and compatibility between subsystems and other systems. External interfaces are synonymous to coordination interfaces. Internal and external interfaces shall be defined, identified and documented at all levels affecting coordination and correlation. The resulting interfaces shall be defined and managed according to the ICP.

b. *Identify all Interface Documentation (ID).* The ICP shall identify all configuration IDs used to manage and control internal and external system interfaces.

c. *ID Incorporation.* The ICP shall define the methods for flowing Interface Documentation (ID) into Technical Data Packages (TDP).

d. *Verification Process.* The ICP shall define the verification process and methods used to ensure that all internal and external interfaces meet the specified requirements. Verification of internal and external interfaces shall be performed at the lowest level where the interface characteristic can be completely verified. Interface requirements shall be included in the requirements traceability and verification matrix 3.2.10 and 3.3.3.2.

3.9.2 Interface Documentation

The developer shall develop, maintain, and control the following Interface Documentation (ID) in accordance with 3.10. The IDs shall be developed according to requirements flowed down from top-level requirements and include:

a. Interface Control Documents and Drawings (ICD), which shall define and establish functional, electrical, mechanical, optical, and data interface requirements. Interface Control Drawings define detailed external and/or internal interface dimensions, parameters, characteristics, requirements, and configurations.

b. Interface Requirement Specification (IRS), which shall define and establish external and/or internal software and data interface requirements for various system, computer, processor, and data interfaces.

c. Interface Design Description (IDD), which shall define and establish detailed design for one or more interfaces incorporated into computer software configuration items, components, and units. The IDD documents the design for those interfaces specified by IRSs.

Interface documentations (e.g., ICD, IRS, IDD) shall be submitted to the cognizant MDA Deputates and Elements for review and approval.

3.9.3 Interface Control Working Groups

The developer shall develop an Interface Control Working Groups (ICWG), which manages interfaces to ensure integration and compatibility within the developer system(s) and to external systems. ICWG membership shall include representatives within the developer's organization, affected suppliers, and a technical government representative. ICWG members shall be selected based on their knowledge and experience and shall have authority to act for their respective organizations. Records of ICWG minutes and actions shall be maintained and provided to MDA Systems Engineering & Integration ICWG, the cognizant MDA Deputates and Elements, and/or designated representative upon request. The developer shall also provide representation to the MDA Systems Engineering & Integration ICWG, cognizant MDA Deputates and Elements, and designated representative level ICWGs upon request.

The ICWG shall:

a. Plan, schedule, execute interface definition activities, and resolve interface issues.

b. Ensure that their actions do not affect safety, quality, or mission assurance.

c. Provide technical support to other system level ICWGs.

d. Communicate all issues related to interface control to program management representatives and other ICWGs, as necessary.

e. Ensure the design meets interface requirements and coordinate proposed interface changes to the Technical Data Package.

f. Coordinate with affected organizations to discuss and resolve technical problems or issues.

3.9.4 Interface Change Notice

The developer shall generate an Interface Change Notice (ICN) to incorporate Interface Control Working Group approved changes resulting from integration incompatibility issues and changes to Interface Documentation (ID). The purpose and rationale for the requested change will be detailed in the ICN. All ICNs must include the exact text, figure, or drawing change to be incorporated and will be related to the existing ID. The ICN shall be submitted to the appropriate change board for disposition and classification. Approved ICNs shall be processed in accordance with configuration management change control process (3.10.3). This process shall be defined in the Configuration Management Plan (3.10.1.2). An ICN, its contents, or any attachments thereto, shall not be used to alter or attempt to alter the existing contractual obligations.

3.10 Configuration Management

The Configuration Management (CM) provisions contained herein were derived from basic CM principles defined in ANSI/EIA-649, National Consensus Standard for Configuration Management. ANSI/EIA-649 is a guidance document. The standard explains the major CM functions rather than mandates them and is not intended for use as a compliance document or an evaluation mechanism for CM programs; rather it is intended as a source document.

In accordance with the ANSI/EIA-649, MDA has applied the recommended principles and practices applicable to a product environment(s) in which a robust configuration management approach is necessary. MDA product environments typically involve complex products such as electronic systems, a military weapon, an air or land vehicle, or any product containing hardware, software, or combination of both which must be supported over the product's life.

3.10.1 Configuration Management and Planning

The developer shall establish and maintain a system for planning Configuration Management (CM) processes for the context and environment in which they are to be performed and manage those processes in accordance with the planning: assign responsibilities; train personnel; measure performance; and assess measurements/trends to effect process improvements.

CM management and planning shall include:

a. Assurances that appropriate CM processes and activities are applied.

b. Established organizational responsibilities for CM activities.

c. Identification and allocation of necessary resources and facilities.

d. A basis for continuous improvement.

e. Enhanced maturity of the developer's process.

3.10.1.1 Identifying Context and Environment

When flowing down Configuration Management (CM) requirements, the developer shall determine the specific CM value added functions and levels of emphasis for a particular product, and identify the context and environment in which CM is to be implemented.

Selection of appropriate CM requirements to be imposed on suppliers shall be in accordance with EIA-649, Annex B.

3.10.1.2 Configuration Management Plan

The developer shall develop and maintain a Configuration Management (CM) Plan, which describes how configuration management is accomplished and how consistency between product definition, product configuration, and configuration management records is achieved and maintained throughout applicable phases of the product's life.

The CM Plan shall be submitted to MDA Systems Engineering & Integration for information and the cognizant MDA Deputates and Elements for approval and, as a minimum, include the following:

a. General product definition and scope.

b. Description of CM activities and procedures for each major CM function, including:

1) Configuration planning and management.

2) Configuration Identification.

3) Configuration change management.

4) Configuration status accounting.

5) Configuration verification and audit.

6) Configuration management of digital data.

c. Organization, roles, responsibilities, and resources.

d. Programmatic and organizational interfaces.

e. Deliverables, milestones, and schedules.

f. Supplier flow down.

3.10.1.3 Implementation Procedures

The developer shall establish and maintain procedures that define how each Configuration Management (CM) process will be accomplished. Procedures shall be evaluated to ensure consistency with CM planning.

3.10.1.4 Training

The developer shall conduct training (3.1.8.1) so that all responsible individuals understand their roles and responsibilities and the procedures for implementing configuration management processes.

3.10.1.5 Performance Measurement

The developer shall assess the effectiveness of Configuration Management (CM) Plan implementation and performance of the configuration management discipline with defined metrics (performance indicators) (3.1.3).

Data derived from metrics shall be used to understand problems and inefficiencies in products and processes, to assess the extent of those problems and inefficiencies, and to provide insight in making necessary corrections and improvements. CM processes and procedures shall be reviewed and revised, using metrics data. Metrics shall be adjusted for the program environment and product life cycle phase. CM metrics include, as a minimum:

a. Number of configuration documentation releases (Scheduled/Actual).

b. Number of engineering changes (by product, by classification, by phase, by time period).

c. Average engineering change cycle time (by product, by classification, by major process step including customer review and approval where applicable).

d. Average revisions per engineering change (in-house, after submittal to customer).

e. Number of variances (by product type, by phase, by time period, per delivered unit).

f. Average number of unincorporated changes (attachments) per engineering drawing.

3.10.1.6 Supplier Configuration Management

The developer's system for Configuration Management (CM) shall include responsibility for the configuration management performance of suppliers.

CM requirements appropriate to the product being acquired shall be flowed down to lower level suppliers, via purchase order or other supplier agreement instruments (3.13.4). Suppliers shall be monitored via data reviews, configuration change management, design reviews, product test results, configuration audits, and supplier evaluations.

3.10.2 Configuration Identification

The developer shall establish and maintain a system for configuration identification as a basis from which the configuration of products are defined and verified; products and documents are labeled; changes are managed; and accountability is maintained.

The system for configuration identification shall include:

a. Selecting products base on functionality, verifiability, supportability, complexity, risk, and management activity.

b. Determining the structure (hierarchy) of a product and the organizational relationships of its configuration documentation and other product information.

c. Documenting the performance, interface, and other attributes of the product.

d. Determining the appropriate level of identification marking of product and documentation.

e. Providing unique identity to a product or a component part of a product.

f. Providing unique identity to the technical documents describing a product.

g. Modifying identification of product and documents to reflect incorporation of major changes.

h. Maintaining release control of documents for baseline management.

i. Enabling a user or a service person to distinguish between product versions.

j. Enabling a user or a service person to correlate a product to related user or maintenance instructions.

k. Facilitating management of data and information including those in digital format.

l. Correlating individual product units to warranties and service life obligations.

m. Enabling correlation of document revision level to product version/configuration.

n. Providing a reference point for defining changes and corrective actions.

3.10.2.1 Product Information

The developer shall establish and maintain a system for the development and control of product information consisting of configuration documentation and operational information. Configuration documentation defines the functional, performance, and physical attributes of a product. Operational information is derived from configuration documentation. Configuration documentation consists of:

a. Requirements documentation defining the performance (capabilities) and functional boundaries of the product and its physical and functional interfaces. Requirements documents are typically in the form of specifications and interface documents.

b. Design documentation defining the physical and functional attributes necessary to completely define the design details of the product. Examples of design documentation are engineering drawings and parts lists (see ASME Y14.24M and Y14.34M), and software design documents.

Operational information, as well as build and test information, includes information necessary to use and operate the product (e.g. operating procedures) and other documentation necessary to service and maintain the product.

3.10.2.2 Product Structure

The developer shall define the product composition (i.e., relationship and quantity of parts that comprise the product) as determined from its configuration documentation.

The product structure shall consist of a representation of the breakdown hierarchy (i.e., product tree/pyramid) of a complex product, from the top down to the lowest level. Each level shall reference associated configuration documentation (e.g. engineering drawings, bill of material, specifications, software requirements and design requirements, and processes/procedures). The product structure shall also indicate the top-down relationships among the various parts that make up the product and the quantity of each. The product structure shall be considered complete when all parts and configuration documentation are included.

The developer shall utilize the product structure in determining the recommended Configuration Identification (CI) level(s) at which to apply Configuration Management, and in evaluating the impact(s) of proposed changes to the product.

The cognizant MDA Deputates and Elements will approve the final CI selection.

3.10.2.3 Product Identifiers

The developer shall assign unique identifiers to all products so that one product can be distinguished from other products; one configuration of a product can be distinguished from another; source of a product can be determined; and the correct product information can be retrieved.

The product, and each of its component parts shall be assigned unique identifiers, as follows:

a. Unique identifiers shall be assigned for all new products down to the lowest level in each branch of the product structure for which the developer has development responsibility.

b. Already developed products used as components of the product shall retain their existing identifiers unless modified to the extent that interchangeability is affected.

c. Parts of the product developed by, or acquired from, suppliers shall retain the unique identifiers assigned by the supplier. For product requiring special requirements, the developer shall provide a unique identifier in addition to the supplier's identifier, to correlate the part to its specification requirement.

d. When a change is applied to a product or part of a product, its descriptive configuration documentation (engineering drawing, product model) shall be updated to reflect the change. The unique identifier assigned to a product, or part, and the marking on the part itself, shall be changed to distinguish one configuration of the product from another, when any:

1) New or updated part is no longer interchangeable functionally or physically with the previously delivered part.

2) New or updated part is no longer interchangeable functionally or physically with the previous undelivered parts that will remain in a different configuration.

3) New part requires new or revised testing, maintenance, repair, training, operating procedures, equipment, or software.

4) Part is altered, selected, or is a source controlled item (ASME Y14.24M).

5) Updated part has different application, use, safety, or other restrictions.

e. When a repair part within a product is changed so that it is no longer interchangeable with its previous version, it shall be assigned a new identifier. Specifically, the developer shall re-identify the next higher assembly and all subsequent higher assemblies up to and including the level at which interchangeability is re-established, or an identifiable end product (against which configuration changes are tracked) is reached.

Unique software identifiers shall be assigned for each software product and for the software products used in the software engineering and test environments. The software identifier shall include the version of the entity. Software units shall be assigned a name or number that is unique within the software product. The marking and labeling of software shall be accomplished as follows:

a. The software identifier shall be embedded in the source and executable code header.

b. Each software medium shall be labeled with the supplier's code/name identification and the software identifiers of the software product it contains. If it is impracticable to include all software identifiers, the medium shall be labeled with a reference to an embedded list (such as a readme.txt file) containing the identifiers.

c. Wherever possible, electronically reprogrammable hardware with resident software shall be labeled with the software identification number of the resident software in addition to the hardware part number.

3.10.2.3.1 Identifying Individual Units of Product

The developer shall assign each individual unit of a product a unique product unit identifier (serial number) when there is a need to distinguish one unit of the product from another unit of the product. When a product is modified, the developer shall retain the products original product unit identifier (serial number) even though its part identifying number is altered to reflect a new configuration.

3.10.2.3.2 Identifying Groups of Units of a Product

The developer shall assign a series of like units of a product a unique product group identifier (lot or batch number) when it is unnecessary or impracticable to identify individual units but nonetheless necessary to correlate units to a process, date, event, or test.

The lot or batch number shall use an identifier as a base. When a product is modified, it shall retain its original product group identifier even though its part identifying number is altered to reflect a new configuration, unless the modification involves a new grouping. A lot number shall be changed if two or more lots, or parts of lots, are reworked as a new amalgamated lot.

3.10.2.4 Document Identification

The developer shall uniquely identify all documents reflecting product performance, functional, or physical requirements and other product information so that they can be correctly associated with the applicable configuration of the product.

As a minimum, document identification shall include the originator/design organization, a unique identifier assigned by the originator/design organization (number, alpha numeric identifier, or title/subject) and a revision indicator (number, letter, or date). The document identifier shall also include the type of document and/or a customer's contract or purchase order number.

3.10.2.5 Baselines

Baselines shall be established, which identify an agreed-to description of the attributes of a product at a point in time, and provides a known configuration to which changes are addressed.

Baselines shall provide an assurance of the stability and consistency of information needed for the subsequent activities and permit the transfer of authority over all or a portion of a product's definition.

3.10.2.5.1 Establishing Baselines

Baselines shall be established by agreeing to the stated definition of a product's attributes.

Baselines shall be established for agreed points of departure. For new products, the definition of desired performance attributes shall be established as the initial baseline. For complex developments, intermediate baselines shall be used to decompose an overall design approach and allocate requirements to various subdivisions. Initial baselines shall be later supplemented with a detailed description (e.g. set of drawings) of the resulting product. The documentation defining a product's baselines shall be mutually consistent and compatible. Each more detailed baseline level must be traceable to, and be a detailed extension of, its predecessor(s).

The developer shall review any document or data set being considered for inclusion in a baseline to ensure the document is complete, valid, and suitable for use. A release system/process shall be employed to validate the document and file integrity. The system for configuration change management (3.10.3) shall be used in managing changes to baselines.

Regarding baselines, the configuration management system shall define:

a. What baselines are to be established.

b. When and how baselines will be defined.

c. The process for assuring document and file integrity

d. The authority to approve baseline and changes.

e. If and when change authority will transfer.

f. The process by which proposed changes will be dispositioned.

3.10.2.5.2 Types of Baselines

The configuration of any product, or any document, plus the approved changes to be incorporated shall be considered the current baseline.

As determined by the cognizant MDA Deputates and Elements, the developer shall establish and maintain the following types of baseline configuration documentation as required:

a. *Requirements Baseline*. Requirements baseline is a product of the concept phase. Requirements baseline(s) shall document the common understanding of what the product is expected to do (its functional and performance requirements) and define the capabilities the government expects to receive from the product. The requirements baseline shall provide a basis for commencing the definition phase. The developer shall establish subordinate requirements (allocated requirements) for major elements of complex products, where it is determined that separate specification of performance requirements for selected elements is appropriate. The developer is responsible for assuring allocated requirements provided to suppliers is consistent with the requirements baseline for the complex product. Allocated baselines may be required by the cognizant MDA Deputates and Elements to incrementally verify performance of the product.

b. *Design Release Baseline*. Design information shall be created, reviewed, and incrementally released over the product's life. Once design information is released it shall become part of the design release baseline controlled by the developing activity. The developer shall establish and maintain a process for the initial release of design information and for the release of approved engineering changes to the design information.

c. *Product Configuration Baseline*. A baseline shall be established for a product, when the design of the product is frozen. The product baseline becomes the basis for commitments for such things as production resources, materials, training devices and material, and operating and service instructions. The product baseline is defined by the complete set of current product configuration documentation. Once the product configuration is frozen, it shall be changed only through appropriate configuration change management (3.10.3).

d. *Additional Operational and Disposal Phase Baselines*. Supplemental baselines may be required that are either location oriented views (extracts) of the product configuration baseline, or that add supplemental information of concern to the product operation, support, or maintenance. These baselines are typically identified and controlled at operational sites.

3.10.2.6 Product Identification Recovery

The developer shall establish and maintain a system for maintaining product information to avoid time consuming and expensive recovery if records of operational units of a product do not match the actual units (as reported by maintenance activities) or such records do not exist.

3.10.2.7 Interface Control

For product interfaces external to the developer, an interface agreement and a mutually agreed to documentation of common attributes shall be established.

A mutually agreed upon interface definition (including performance, functional and physical attributes) shall be detailed in an interface document or drawing. Additional interface management requirements are found in section 3.9.

3.10.3 Configuration Change Management

The developer shall ensure changes to a product are accomplished using a systematic, measurable change process.

The configuration change management system shall manage product configuration and variances for all changes to current requirements, design release, or product configuration baselines. The system shall include identifying the need for a change; documenting change impact; evaluating and coordinating the proposed change (including approval/disapproval); incorporating the approved change in the product and its related configuration documents; and implementation of variances from configuration baseline requirements.

3.10.3.1 Change Identification

The developer shall ensure each change is uniquely identified.

As part of change identification, the developer shall include a description of the requested change and its impact sufficient for evaluation, determine the appropriate level of approval, choose the appropriate format (information content) for describing the change, and provide a unique identifier for the change request.

3.10.3.1.1 Requesting Changes

The developer shall ensure changes represent opportunities for improvement.

Preliminary judgments shall be made to identify the proper change authority and to determine the processing method and document format that are most appropriate. When making this judgment consideration shall be given to:

a. The need for the requested change.

b. The basic scope and description of the requested change.

c. The definition of its impacts.

d. The desired effectivity.

e. Its urgency and importance.

3.10.3.1.2 Classifying Changes

The developer shall classify requested changes to aid in determining the appropriate levels of review and approval.

The developer shall use the following classifications to differentiate between Major and Minor changes. Criteria are provided for each classification to determine if government review in addition to internal review is required.

a. *Major (Class I)*. A change classified as Major is a change to the requirements of baselined configuration documentation (requirements, design release or product configuration baselines) that has significant impact. It requires coordination and review by all affected functional groups or product development teams and approval by a designated approval authority (usually an individual who can authorize the resources need for change implementation).

   The following factors/characteristics of engineering change shall constitute the need for a Major change.

   1) Affects approved baseline specification requirements such as performance, reliability, maintainability, weight, balance, moment of inertia, interface characteristics, electromagnetic characteristics, and other technical requirements and specifications.

   2) Affects one or more of the following, after product baseline:

      (a) Products furnished by the government.

      (b) Safety.

      (c) Compatibility with interfacing products (including such products as test equipment, support equipment and associated software).

(d) Delivered operation or servicing instructions for which there are no planned and funded update requirements, such as for periodic or continual maintenance of the instructions.

(e) Preset adjustments to the extent that product identification should be changed.

(f) Interchangeability or substitutability of replaceable products, assemblies, or components.

(g) Change to a previously non-selected supplier, where supplier selection is specified.

(h) User skills or physical attributes.

(i) Operator or maintenance training.

3) Requires retrofit of delivered products, by product recall, modification kit installation, or attrition (replacement during maintenance by modified spares).

4) Affects cost/price to government (including incentives and fees), guarantees, warranties, contracted deliveries or milestones; and is an engineering change that does not impact factors (1) through (3).

A Major change shall require MDA Program Change Board (PCB) approval.

b. *Minor (Class II)*. A Minor change corrects or modifies configuration documentation (released design information), processes or parts but does not impact any characteristics that would cause it to be classified as Major (factors (1) through (4) above). Minor changes do not impact government requirements, but should be visible to the government, and require internal approval at an appropriate level commensurate with their impact.

A Minor change shall require government involvement under the following circumstances:

1) The product configuration baseline is established, and

2) The government is concerned with (or controls) the product's detail design in addition to its performance and interface attributes, and has imposed government management procedures on the detail design; and

3) The contractual agreement stipulates either that the cognizant MDA Deputates and Elements or designated representative must review the change for classification, or must approve minor changes.

3.10.3.1.3 Documenting Requests for Changes

The developer shall clearly document change requests.

Changes requests shall be documented and describe even minor changes so that an audit trail can be constructed.

Documentation of major changes shall include the following information to allow for an informed evaluation of the change and to identify the change:

a. Unique change identifier.

b. Originator organization and responsible individual.

c. Class of change.

d. Product(s), major components, interfacing products affected.

e. Contract and configuration documents affected.

f. Scope and description of change.

g. Effects on specified performance, operation, maintenance, servicing, operation and maintenance training, spare and repair parts, and support and test equipment.

h. Reason and justification for the change; consequences of not doing the change.

i. Priority/urgency of the change.

j. Proposed change effectivity.

k. Requested approval date.

l. Change implementation and delivery schedules.

m. Estimated cost increase or savings.

n. Alternatives.

Minor changes shall be documented in the format used to release and communicate design changes. As a minimum, the following information is required:

a. Unique change identifier.

b. Originator organization and responsible individual.

c. Class of change.

d. Product(s), assemblies, and components affected.

e. Configuration documents affected.

f. Description of change.

g. Reason for the change.

h. Proposed change effectivity.

3.10.3.2 Change Evaluation and Coordination

The developer shall consider technical, support, schedule, and cost impacts of a requested change before making a judgment as to whether the change should be approved for implementation and incorporation in the product and its documentation.

The evaluation and review process shall encompass reviewing the preliminary impact assessments; determining the required change effectivity; establishing the cost/price; and dispositioning (approving, deferring for more research, or disapproving) the change.

3.10.3.2.1 Change Impact Assessment

The developer shall determine all potential effects of a change and coordinate potential impacts with the impacted areas of responsibility.

The impact assessment shall include details of what would be affected by the change and ensure that all potential effects are known.

The developer shall establish a Configuration Control Board (CCB) with the authority for achieving the coordination necessary to evaluate a change and assess its impact. The CCB shall:

a. Be chaired by someone with the authority to commit the resources to implement the change.

b. Include members, which represent the functional activities (e.g., Engineering, Quality, Safety, Maintenance, Operations) and include technical government representation.

c. Provide review agendas and documents to board members prior to the meeting.

d. Document and disseminate board direction and decisions to all affected activities and retain them as a record.

The developer shall support government PCB activities, as required.

3.10.3.2.2 Change Effectivity Determination

The developer shall ensure change documentation delineates which unit(s) of the product are to be changed. Change effectivity shall include fabrication break-in and retrofit/recall, as applicable.

A changed product shall not be distributed until required support and service areas are able to support it.

When determining the effectivity of a change, the developer shall consider, as a minimum:

a. Urgency of the change, e.g., is safety involved?

b. Parts and materials on hand.

c. Need to support multiple configurations when all existing units of the product will not be updated, or will not be updated at the same time.

d. Timing of the introduction of the changed product with respect to government preferences and needs.

3.10.3.2.3 Change Cost/Price Determination

The developer shall ensure the decision-maker is aware of all cost factors in making the decision.

The decision shall be based on a cost/benefit analysis covering the remaining product life. Cost estimating and pricing of a change shall be based on the knowledge resulting from the impact assessment and effectivity determination. Immediate costs and expected costs that will be incurred in the future as a result of the change shall be considered. Cost factors to be considered include:

a. Increased/decreased price for new/replacement parts.

b. Scrapping of parts on hand.

c. New tooling/software required for manufacturing the product.

d. Recall and retrofit of already purchased units.

e. Fulfillment of service contracts.

3.10.3.2.4 Change Approval Authority

The developer shall ensure that change approval decisions are made by an appropriate authority who can commit and allocate resources to implement the change.

The developer shall determine change authority levels taking the following factors into consideration:

a. Phase of the product's life.

b. Extent of technical, support, schedule, quality, safety, and cost impacts.

c. Government's involvement.

d. Nature of the product attributes.

e. Organizational structure and relationships within.

f. Various levels at which change authority is vested.

g. Period of performance during which the level of control applies.

The Configuration Control Board (CCB) shall disposition changes within its defined limited authority or to the portion of the "system" under its cognizance. Those changes that exceed the change approval authority of CCB shall be elevated to a higher level.

3.10.3.3 Change Implementation and Verification

The developer shall implement an approved change in accordance with documented direction approved by the appropriate level of authority.

Implementation of a change shall include the identification and release of new or revised configuration documentation including requirements and design information. The release process shall correlate the document revisions to the change, or changes, incorporated. Document change notices that establish a permanent record of the specific changes shall be used in disseminating document changes. For changes affecting interface, an Interface Change Notices (ICN) shall be generated and submitted to the appropriate Configuration Control Board (CCB).

The developer shall verify implementation of a change to ensure consistency between the product, its documentation, and its support elements.

The developer shall perform verification of change implementation in the first affected unit to ensure consistency between the product, its documentation, and its support elements. Depending on the nature of the product and the complexity of the change, verification shall involve a detailed audit of the product against its documentation; a validation of operation, maintenance, installation, or modification instructions; or a simple inspection.

When the change is being introduced into a production line the developer shall ensure that the manufacturing instructions contain the change, are released for use, and that the first articles produced are inspected for compliance. If support elements are impacted by the change, or the change is being retrofitted over a period of time to a large number of units, a change implementation plan shall be developed. The plan shall define the extent to which the change to each unit or support commodity is to

be verified, and the records to be maintained. If the total quantity of materials, parts, or kits, is ordered in incremental stages, the developer shall verify that the incremental ordering and supply operations are being completed.

3.10.3.4 Change Management Process Applied to Variances

If the developer considers it necessary to temporarily depart from specified baseline requirements, a variance (waiver or deviation) shall be documented and authorized by the appropriate level of authority.

Products that incorporate a known departure from requirements (even if the requirements are internally specified) shall not be delivered to the government unless a variance has been documented and authorized. Authorized variances are temporary departures from requirements and do not constitute a change to the configuration documentation. If it is decided the departure will be permanent, an engineering change is required. Similarly, variances shall not be processed if it would affect operation, support, or maintenance; or if it would include the entire remaining number of deliverable units of the product. Rather, an engineering change shall be proposed.

Requests for a variance shall be documented and include the following information:

a. Unique identifier for the variance.

b. Originator organization and responsible individual.

c. Classification of variance.

d. Identifiers of the product(s) and components affected.

e. Description of the variance, including any impacts to performance, operation, maintenance, servicing, quality, safety, operation and maintenance training, spare and repair parts, and support and test equipment.

f. Reason/justification for the variance.

g. Priority/Urgency.

h. Proposed effectivity of the variance (limited quantity or time).

i. Corrective action to prevent recurrence and/or to eliminate the variance.

j. Consideration, if any, for accepting variant products.

k. Alternatives.

Variances shall be classified to facilitate determination of the appropriate level of approval required for the variance and action to be taken to prevent recurrence. After approval of the variance, the developer shall take corrective action to prevent recurrence or to eliminate it completely (e.g. engineering change).

3.10.3.4.1 Request for Waiver

The developer shall process a request for waiver if, during or after fabrication of an item which incorporates a known departure from requirements, it is determined that the item is considered suitable for use-as-is or repairable by a non-standard method.

3.10.3.4.1.1 Restrictions on Waivers

a. *Effectivity.* The effectivity of the request for waiver shall not include unprocessed units still deliverable under contract. If that is the case, an engineering change or deviation shall be submitted.

b. *Recurring.* The developer shall identify and minimize recurring waivers. A proposed waiver is considered recurring if it is a repetition or extension of a previously approved waiver. If it is necessary to request a waiver for the same situation more than once then the need for an engineering change, rather than a waiver, shall be addressed.

c. *Software*. Waivers for software code listings shall not be submitted.

3.10.3.4.1.2 Classification of Waivers

Each request for waiver shall be designated as critical, major, or minor by the originator in accordance with this document. Classification disagreements shall be referred to the cognizant MDA Deputates and Elements for decision:

a. *Critical.* A waiver shall be designated as critical when:

1) The waiver consists of acceptance of an item having a nonconformance with contract or configuration documentation involving safety, or

2) The nonconformance impacts a product characteristic, which is classified as critical (3.2.15.1).

Note: Unless unusual circumstances exist, requests for waivers affecting safety or which would affect service operation or maintenance will not be authorized.

b. *Major.* A waiver shall be designated as major when:

1) The waiver consists of acceptance of an item having a nonconformance with contract or configuration documentation requirements involving: occupational health; performance; interchangeability, reliability, survivability, or maintainability of the item or its repair parts; effective use or operation; weight; or appearance (when a factor), or

2) The nonconformance impacts a product characteristic, which is classified as major (3.2.15.1).

c. *Minor*. A waiver shall be designated as minor when:

1) The waiver consists of acceptance of an item having a nonconformance with contract or configuration documentation, which does not involve any of the critical or major factors listed above.

2) The nonconformance impacts a product characteristic, which is classified as minor (3.2.15.1).

3.10.3.4.2 Requests for Deviations

The developer may request a deviation prior to fabrication of the item, if it is considered necessary to temporarily depart from the mandatory requirements of the specification or drawings.

3.10.3.4.2.1 Restrictions on Deviations

a. *Effectivity.* The effectivity for the request for deviation shall be either the minimum number of units to support Engineering Change Proposal generation and approval or the minimum number of units necessary to support return to the approved baseline configuration.

b. *Recurring.* A proposed deviation is considered recurring if it is a repetition or extension of a previously approved deviation. If it is necessary to request a deviation for the same situation more than once then the need for an engineering change, rather than a deviation, shall be addressed.

c. *Software*. Deviations for software code listings shall not be submitted.

3.10.3.4.2.2 Classification of Deviations

Each request for deviation shall be designated as critical, major, or minor by the originator in accordance with this document. Classification disagreements shall be referred to the cognizant MDA Deputates and Elements for decision.

a. *Critical.* A deviation shall be designated as critical when:

1) The deviation consists of a departure involving safety, or

2) The departure impacts a product characteristic, which is classified as critical (3.2.15.1).

Note: Unless unusual circumstances exist, requests for deviations affecting safety or which would affect service operation or maintenance will not be authorized.

b. *Major. A deviation shall be designated as major when:*

1) The deviation consists of a departure involving: health; performance; interchangeability, reliability, survivability, maintainability, or durability of the item or its repair parts; effective use or operation; weight and size; or appearance (when a factor), or

2) The departure impacts a product characteristic, which is classified as major (3.2.15.1).

c. *Minor. A deviation shall be designated as minor when:*

1) The deviation consists of a departure, which does not involve any of the critical or major factors listed above, or

2) The departure impacts a product characteristic, which is classified as minor (3.2.15.1).

3.10.3.4.3 Review and Approval of Variances

Unless otherwise specified by the procuring activity, “minor” variances will be dispositioned by the cognizant Contract Administration Service (CAS) component. Variances classified as “critical” or “major” will be reviewed by the CAS component. However, only the cognizant MDA Deputates and Elements, or a specifically designated representative, may approve “critical” or “major” variance requests.

3.10.4 Configuration Status Accounting

The developer shall establish and maintain an accurate information base concerning a product and its associated product information throughout the product’s life.

Configuration information shall be collected while performing activities associated with the Configuration Management (CM) processes (planning and management, identification, change management, and verification and audit). The developer shall establish and maintain a Configuration Status Accounting (CSA) system, which correlates, stores, maintains, and provides ready access to information about a product and its documentation throughout the product’s life. The CSA system shall ensure the storage and maintenance of the following information:

a. Configuration documentation such as document identifiers and effective dates.

b. Product configuration such as part numbers and changes installed in a given unit.

c. Product operational and maintenance documentation such as documents affected by each change and their update status.

d. CM process such as the status of change requests.

The CSA system shall support supplier, and cognizant MDA Deputates and Elements or designated representative access to information as needed and improve the capability to identify, produce, inspect, operate, maintain, repair, and refurbish products. The CSA system shall provide for:

a. Retrieval of information concerning change decisions.

b. Inquiries concerning future planning of design changes, investigations of design problems, warranties, and shelf and operating life calculations.

c. Access to complete configuration information (configuration pedigree) on a product, any individual product unit, or group of product units.

d. Access to accurate identification of the configuration of each delivered product unit (or batch/lot of product units).

e. Availability of accurate information on spare parts and maintenance support.

f. A source for configuration history of a product and all its configuration documentation.

3.10.4.1 Configuration Status Accounting Information

The developer shall systematically record, safeguard, validate, and disseminate, configuration information appropriate to the product.

The configuration information content evolves and shall be captured over the product's life as tasks occur.

The developer shall capture the following information into the Configuration Status Accounting (CSA) during each of the life cycle phases.

a. *Concept Refinement and Technology Development Phase*. Initial requirements documents are generated during this phase. Information about those requirements documents and their change history shall be recorded in the CSA system.

b. *System Development and Demonstration Phase*. The product structure is created and dynamically updated as the product's remaining requirements documents and detailed configuration documents are generated, released, and baselined. Information about configuration documentation (specifications, engineering drawings, software design documents, software code), test plans, procedures and any other documents required to design, develop, build (fabricate), test, and verify the product configuration shall be recorded in the CSA system. All change activity information shall be captured including the status, history, and effectivity of all changes that are proposed, dispositioned, and incorporated in the product and affected documentation. The status and history of variances shall be captured as well. As the design of the product evolves, the CSA shall contain a record of the release of each configuration document and each subsequent revision to update the design release baseline. Data accompanying the release record shall provide the applicability and effectivity of parts, or software units. At or near the end of the system development and demonstration phase or the beginning of the production (fabrication) and deployment phase, the current design release baseline shall be verified and become the product configuration baseline. During the verification process, the CSA shall capture the results of configuration verification and audits (including action item status).

c. *Production (Fabrication) and Deployment Phase*. CSA information accessible during the prior phases shall continue to be available in the build phase. Additional information about the product, and changes to the product are recorded and tracked. Information that shall be available for extraction includes:

1) As-built configuration of each unit (by serial number), including installation and removal of serialized and lot numbered components.

2) Identifiers and product unit serial numbers which constitute effectivity of each approved major engineering change; the identifiers of the changes which have been released for any specific serial-numbered unit of a product.

3) Superseded configuration records that reflect prior configurations of the product.

For products requiring installation, changes during the installation process that affects the product shall be added. Information that shall be available for extraction includes:

1) Customers and dates of delivery.

2) Installation configuration.

3) Warranty expiration dates for each unit delivered or installed.

4) Service agreement type and expiration.

d. *Operations and Support Phase*. Changes to the product, which occur due to parts replaced during maintenance, modification of the product by retrofit of engineering changes, or installation of hardware upgrades, shall be recorded in the CSA. Information that shall be available for extraction includes:

1) Product as-maintained and as-modified configuration.

2) Product operation and maintenance information revision status.

3) Product information change requests and change notices.

4) On-line information access directory or index.

5) Restrictions due to facility/product performance degradation.

e. *Disposal Phase*. CSA information to be retained depends on product type, and whether disposing of the product has adverse environmental implications; if there are product replacements; or if parts salvage is required. The developer shall retain disposal data as required by legal and contractual statutes.

3.10.4.2 Configuration Status Accounting System

The developer's data collection and information processing system requirements are determined by the need for configuration information.

Depending on the depth of product configuration needed, the Configuration Status Accounting (CSA) system shall consist of an information system such as a product data manager or workflow manager capable of providing storage and security of product information and traceability of product history. The system shall provide structured records on the product and its related documentation. When required, the system shall be capable of providing real time access and transfer of CSA information between customers, product development teams, suppliers, and others.

3.10.5 Configuration Verification and Audit

The developer shall perform verification that a product's requirement attributes have been met and that the product design meeting those attributes has been accurately documented to baseline the product configuration.

Configuration verification and audit shall be performed to establish that the performance and functional requirements defined in the configuration documentation have been achieved by the design and that the design has been accurately documented in the configuration documentation.

3.10.5.1 Design and Document Verification

The developer shall verify that a design is achieved by performing a systematic comparison of requirements with the results of tests, analyses, or inspections.

The developer shall verify the design of the product to ascertain that it has achieved specified requirements and desired goals; that the documentation of the design is accurate; and that the product can be produced from the documentation. Design verification (3.2.11) shall occur incrementally during the course of the definition phase and shall be incorporated into the design/manufacturing process flow, so that it occurs on a continuous basis.

Design verification methods shall be planned and documented to ensure all requirements are addressed and the individual verification methods chosen are appropriate. To facilitate the design verification process, the developer shall perform a requirements analysis and utilize test tools, which flow down, account for, and verify all design attributes. Design verification results shall be recorded to indicate each discrete requirement, the method of verification, verification procedure, and the verification results.

The developer shall ensure documentation of a product's definition is complete and accurate enough to permit reproduction of the product without further design effort.

The developer shall ensure the design output, consisting of the complete set of design information, is accurately documented to permit reproduction of the product without further design effort.

The developer shall ensure software products are in compliance with published design and coding standards so it can be maintained, modified, and upgraded. The following should be verified:

a. The documentation library control system.

b. Uniqueness of the product identifier.

c. Validity of interfaces.

d. Internal audit records of Configuration Management processes and procedures.

Documentation verification shall be performed to ensure documents are adequate for their intended purpose and reflect compliant design. To minimize cost in complex designs, the developer shall perform verification of design and documents incrementally during assembly of the product to avoid the need for disassembly. Verifications shall be considered complete upon resolution of discrepancies or departures found and correction of associated documentation.

3.10.5.2 Configuration Audit

As required, verification shall be accomplished by configuration audit.

The developer shall conduct and participate in configuration audits typically held at the conclusion of the system development and demonstration phase or at the start of the production (fabrication) and deployment phase. Audits include performance verification (functional configuration audit) and design

verification (physical configuration audit). The supplier, cognizant MDA Deputates and Elements, or a designated representative may conduct audits.

To the extent appropriate for the type and scope of audit, the developer shall ensure the following necessary resources and material are available.

a. An audit plan and agenda.

b. Adequate facilities and unencumbered access.

c. Assignment and availability of personnel.

d. Applicable specifications, drawings, manuals, schedules, design data, test results, inspection reports, process sheets, safety procedures, and other documentation deemed necessary.

e. Tools and inspection equipment necessary for evaluation and verification.

f. Access to the product(s) and detailed parts to be reviewed.

Audit results shall be maintained to record audit findings, conclusions, recommendations, and action items. Follow-up shall occur until all action items are complete.

The developer shall conduct configuration audits of supplier mission critical items.

3.10.5.3 Continuing Performance Audits and Surveillance

The developer shall conduct periodic reviews to verify continued achievement of requirements, identify and document changes in performance, and ensure consistency with documentation.

Evaluations (3.1.7) of all portions of the Configuration Management (CM) process and procedures shall be conducted periodically or when indicated by process metrics to enhance the effectiveness of the CM process.

3.10.6 Configuration Management of Digital Data

The developer shall apply configuration management principles to ensure the integrity of digital representations of product information and data.

Digital data includes all product information and data prepared and maintained by electronic means, and provided by electronic data access, interchange, transfer, or on electronic media.

The developer shall establish a system for data configuration management, which assures the integrity of digital data by providing:

a. Effective file and database management.

b. Unique identification.

c. Retention of essential file and version relationships.

d. Known data status.

e. Controlled access to digital data.

3.10.6.1 Digital Data Identification

The developer shall apply digital data identification rules to maintain document, document representation, and file version relationships.

The developer shall identify digital data files to differentiate between similar files and to maintain traceability to specific product configurations and representations. To facilitate the proper digital data relationships, the developer shall establish and apply the following digital data identification rules:

a. Assign a unique identifier to each file.

b. Assign a unique identifier to each document representation.

c. Assign a version identifier to each file.

d. Maintain, in a data file, the relationship between:

   1) Document identifier and its revision level.

   2) Associated document representation(s).

   3) File identifiers and versions.

e. Retain multiple versions of files as necessary to recreate prior document revisions and provide a traceable history of each document.

3.10.6.2 Data Status Level Management

The developer shall apply business rules using data status (state) levels for access, change management, and archiving of digital data documents.

The data status level management shall define and apply business rules based on the status of a digital data document to facilitate data workflow management and enhance data integrity. Data status levels include definition, working, released, submitted, approved, and change to digital documents. The developer shall apply the following rules to each document application status to determine:

a. Document/document revision identifier and file version identifier scheme.

b. Application software and data format.

c. If submittal to (or access by) government is required.

d. Who will be granted access privileges to the data in each of the application states.

e. The approved requirements (reviewers/approvers) and method of approval (electronic signature) to enter the released status; the approved status.

f. The archiving rules for this document type.

3.10.6.3 Maintenance of Data and Product Configuration Relationships

The developer shall maintain relationships between digital data, data requirements, and the related product configuration to ensure accurate data access.

The developer shall establish an effective system for managing the relationships of data files, document representations, and key data elements to ensure data can be accessed or retrieved in a controlled manner. The relationships (dependencies) shall ensure extracted information concerning a given product

or part is associated correctly with the configuration and effectivity (serial number) of the product that uses the part and not with an earlier or a later configuration that does not.

3.10.6.4 Data Version Control and Management of Review, Comment, Annotation, and Disposition

The developer shall apply disciplined version control to manage document review electronically.

When managing a document review process electronically, the developer shall maintain version control to establish an audit trail of comments or annotations by reviews, and the disposition of each product. Each version of each document representation provided to, or received from, each reviewer shall be uniquely identified.

3.10.6.5 Digital Data Transmittal

The developer shall ensure that a transmitted digital data product is usable.

When data is provided on transportable media, appropriate identification, similar to software media identification, shall be affixed to the media to clearly identify its contents. When it is impracticable to include all the file identifications, a reference to an accompanying listing or to a readme.txt file is required.

The developer shall ensure the deliverable digital data product can be recreated in readable form and processed by the user. Instructions (readme.txt files, reference to standard protocols, on-line help) shall include the following, as applicable:

a. Identification of the files included in the transfer by file name, description, version, data status level, and application/file type.

b. Applicable references to associate the data with the basis (requirement) for its transmittal, approval, and payment, where applicable.

c. If there are multiple files, such as separate test and graphics, how to assemble each included data item for reading, review or annotation, as applicable.

d. The naming convention for the file versions and data status level that distinguishes altered (red-lined/strike-out) file versions from altered files.

e. If and how changes from previous versions are identified.

f. How to acknowledge receipt of the data, provide comments, and/or indicate disposition of the data digitally.

g. Time constraints, if any, relating to review and disposition.

3.10.6.6 Data Access Control

The developer shall ensure digital data access fulfills requirements, preserves rights, and provides users with data they are entitled to in the correct version.

The developer shall employ an electronic data access process, which establishes access privileges to limit access to applicable users. Access privileges shall vary according to the data status level, the nature of the data, and the needs of the user.

This Page Intentionally Left Blank

3.11 Control of Nonconforming Items and Materials

The developer shall have a Nonconforming Items and Materials system that is compliant with the minimum requirements of AS9100, Aerospace Model for Quality Assurance in Design, Development, Production, Installation and Servicing and is supplemented by the following requirements. The system shall ensure that items or material found to depart from drawings, specifications, or other requirements are: conspicuously identified as nonconforming; segregated from conforming items or materials, when feasible; and retained in a hold status until officially dispositioned and corrected. The system shall ensure that effective corrective action is implemented to prevent recurrence. All specified tests and inspections impacted by subsequent repair or rework processes shall be repeated. The system shall include a nonconformance review process, which consists of a preliminary review and a Material Review Board (MRB). When nonconforming items or materials are detected after delivery to the customer or use has started, the developer shall notify cognizant MDA Deputates and Elements and MDA/QS of the issue, including a description of its effects, potential effects, and any recommended corrective and preventive actions.

3.11.1 Preliminary Review

The preliminary review process shall be initiated with the identification and documentation of a nonconformance. The preliminary review process shall be performed by authorized personnel to ensure that nonconformances are properly documented and that appropriate examination and analysis of nonconformances are performed to determine cause, implement corrective and preventive action, and specify disposition. The preliminary review shall result in one of the following dispositions:

a. *Remove from Use (Scrap).* Items or materials that are unfit for use and are not economically repairable shall be processed in accordance with approved procedures for identifying, controlling, and disposing of unusable material.

b. *Return for Rework.* Developer-manufactured items or materials, which are found to be incomplete or which can be corrected to completely conform to drawings, specifications, or other applicable requirements may be released for correction or completion of the remaining operations.

c. *Return to Supplier.* Nonconforming items or materials received from a supplier may be returned for rework or replacement. The developer shall provide the supplier with nonconformance information and applicable instructions for the re-submittal of corrected material and the associated corrective action reports.

d. *Standard Repair.* Developer personnel performing the preliminary review may authorize repair using cognizant MDA Deputates and Elements approved standard repair procedures included in the item's configuration documentation. Repairs made via approved standard repair procedures do not require a waiver to be processed.

e. *Submit to Material Review Board.* If none of the above dispositions is appropriate, the item or material shall be submitted for Material Review Board action.

3.11.2 Material Review Board

The developer's Material Review Board (MRB) shall consist of a core team of personnel who are responsible for assuring that MRB actions are performed in compliance with requirements of this provision. As a minimum, MRB core team shall have representation from the engineering and quality disciplines and a designated government representative. The core team may be supplemented with additional developer personnel to ensure timely determination, investigation, engineering analysis, implementation and closeout of recommended MRB disposition. All MRB members shall be selected on the basis of their technical competence. A list of authorized MRB core team members and alternates shall be maintained and made available to the cognizant MDA Deputates and Elements or designated representative upon request.

3.11.2.1 Material Review Board Dispositions

In determining the disposition of nonconforming items, the Material Review Board (MRB) shall consider the effect of nonconformance upon intended use and review any records of MRB actions on similar items. MRB review findings, recommendations, and disposition actions, including supporting information or analysis, shall be documented and are subject to review by the cognizant MDA Deputates and Elements or designated representative. The MRB is authorized to make those dispositions made at preliminary review (i.e., Remove from Use, Return for Rework, Return to Supplier, and Standard Repair). The MRB may also recommend the following disposition:

a. Nonstandard Repair. If repair to an acceptable condition is considered possible and desirable but a standard repair procedure approved by the cognizant MDA Deputates and Elements is not applicable, a waiver request shall be processed in accordance with 3.10.3.4.1.

b. Use-As-Is. If the nonconforming item is considered usable as is, a waiver request shall be processed in accordance with 3.10.3.4.1.

Government acceptance or rejection of waiver requests for repair or use-as-is dispositions on nonconforming items remains a separate and distinct government action from the MRB responsibilities.

The developer shall maintain metrics of MRB actions and dispositions and report in accordance with 3.1.3

3.11.3 Supplier Material Review Board

The developer, upon approval of a suppliers Material Review Board (MRB) system, may assign MRB responsibility to selected suppliers. When this responsibility is assigned, supplier's procedures addressing control of nonconforming items and materials shall be consistent with the provisions of this document. The supplier's procedures addressing control of nonconforming items and materials and all MRB decisions shall be subject to review by the cognizant MDA Deputates and Elements or designated representative.

3.12 Fabrication and Quality

The developer shall establish and maintain control systems for operations associated with fabrication and quality, including any related measurement and analysis, which support the fabrication process and ensure specification requirements are achieved, verified, and maintained. Fabrication and quality activities shall be planned, implemented, and controlled to provide for an efficient and effective program. Established techniques for monitoring fabrication processes shall be used to ensure process capabilities remain adequate to produce required product characteristics. Product conformance shall be verified through the application of quality verification techniques. Process and quality records generated during the fabrication process shall be retained.

The developer shall have a fabrication and quality process that is compliant with the minimum requirements of AS9100, Aerospace Model for Quality Assurance in Design, Development, Production, Installation and Servicing. Some assurance related activities are not covered by AS9100 requirements. These activities are identified in the following sections and shall supplement the AS9100 requirements.

3.12.1 Process and Quality Control Planning

The developer shall plan the necessary process and quality controls, including any related measurement and analysis, to be utilized throughout fabrication.

a. Establish levels, depth, and extent of process control, test, and inspection to be implemented based upon product and process specification requirements, classification of characteristics, and integrated test program results.

b. Utilize process flow diagrams, or equivalent, to identify processes, including critical and key characteristics, relating to fabrication, test, inspection, and acceptance.

c. Identify requirements for facility, equipment, and tooling, including related software and maintenance.

d. Identify any packaging, handling, transportation, and storage requirements from receiving through delivery.

3.12.2 Process Selection and Development

The developer shall establish and maintain a system for selection and development of fabrication processes concurrent with the evolutionary design of the product. The developer shall analyze the ability of proposed processes to fabricate quality hardware with minimum variability using design for producibility methods, and continuous process improvement.

3.12.2.1 Process Selection and Development Planning

The developer shall perform process selection and development planning to support fabrication efforts. Planning shall reflect a phased process maturity approach to support evolutionary acquisition including transitioning to mature production processes (3.2.18). As a minimum planning shall address:

a. Criteria and methods used to determine appropriate control of processes throughout development, such as process capability, inspectability, scrap and rework costs, and the level of quality and reliability required.

b. Criteria and methods for determining the level of development, characterization, capability demonstration, and process qualification to be performed throughout development.

c. Criteria and methods for determining which processes shall be controlled by specifications. The criteria shall be based upon the tolerances, criticality, and application of the product; developer

experience with the process; process complexity; operator skill level required; inspectability; and the extent of subsequent test and inspection.

d. Criteria and methods for determining and controlling mission critical processes whose failure can significantly affect system safety, mission success, availability, or total maintenance/logistics support costs.

e. Criteria and methods for determining special processes for which the resulting output cannot be readily or economically verified by subsequent monitoring or measurement.

3.12.2.2 Mission Critical Process Selection

The developer shall identify mission critical processes based on any one or appropriate combination of the following:

a. Outputs from Reliability Analyses; Failure Modes, Effects, and Criticality Analysis (FMECA); and Process Failure Modes, Effects Analysis (FMEA) (3.5.12).

b. Application of advanced state-of-the-art techniques.

c. Complex productivity or technical complexity.

d. Proprietary design.

e. Limited source, limited material, or sole source availability.

f. Past experience and judgment on similar processes warrants the process be identified as critical.

g. Physical properties of the item are stability sensitive requiring tight process control.

3.12.2.3 Special Processes

The developer shall identify and control special processes used to support fabrication of the item as required by AS9100. Special processes include, but are not limited to: anodizing, bonding, brazing, encapsulating, heat treating, plating, soldering, welding, non-destructive testing, and printed circuit fabrication. The developer shall certify and maintain special process tools and equipment, which assures quality of the end product. The developer shall certify personnel performing special processes in accordance with 3.1.8.2.

3.12.3 Product Test and Inspection Plan

The developer shall establish and maintain Product Test and Inspection Plan(s) (PTIP) to indicate tests and inspections to be conducted during all phases of fabrication, from source or receiving, through final acceptance. The fabrication points at which tests and inspections are to be made shall be specifically identified in fabrication flow documentation (e.g. travelers, operations sheets). Sufficient examination points shall be specified to ensure that tests and inspections are conducted prior to work operations that will preclude detection and correction of deficiencies or result in excessive rework, repair, or cost. The extent of test or inspection shall be consistent with the criticality of the characteristic. PTIPs shall be submitted to the cognizant MDA Deputates and Elements for review and approval and include:

a. Flow diagrams, or equivalent, indicating the sequence of production operations showing tests, inspections, and process control points.

b. Reference to procedures used for acceptance test and inspection.

c. Identification of the part or identifying number and name for each item.

d. Identification of items requiring environmental stress screening (3.5.13), burn-in tests, production assessment testing, and any other special tests.

3.12.4 Fabrication and Quality Procedures

The developer shall establish and maintain a system to develop and control fabrication, test and inspection procedures, and workmanship standards. Procedures and workmanship standards shall be readily available in the manufacturing, test, and inspection areas.

3.12.4.1 Fabrication and Process Procedures

The developer shall establish and maintain procedures for fabrication, processing, assembly, rework, repair, packaging, handling, transportation, and storage operations, as required by AS9100. In addition, these procedures shall contain or reference the following, as applicable:

a. Required workmanship standards and production aids, including material and process specifications and standards applicable to each process.

b. Step-by-step instructions for performing operations and methods for recording completion of each operation.

c. Identification of equipment, tools, and software required, including requirements and methods for certifying tools, equipment, and associated software.

d. Special conditions required to be maintained, such as: conditions required for parts, devices, and material protection; environmental conditions; safety controls; and equipment maintenance.

e. Required characteristics and tolerances, including identification of particular process variables to be controlled and methods by which the variables will be monitored.

f. Identification of mandatory inspections points.

g. Requirements for recording of process data, data analyses to be performed, and responsibilities and actions assigned to ensure control of the process.

h. Identification of any special handling devices required for movement of parts, devices, and material.

i. Identification of applicable personnel certification(s) and/or training required.

3.12.4.2 Test and Inspection Procedures

The developer shall prepare procedures for tests and inspections in compliance with AS9100. In addition, procedures shall include or reference the following, as applicable:

a. Tolerances, levels, or limits of inputs for the characteristics being tested or inspected.

b. Identification and setup of test and inspection equipment and related software.

c. Environmental stress levels required during test or inspections.

d. Method of performing the test or inspection, including sequential steps.

e. Special pretest and inspection instructions.

f. Required safety precautions.

g. Applicable personnel qualification or certifications required.

Acceptance test and inspection procedures used as a basis for government acceptance of contract end items shall be submitted to the cognizant MDA Deputates and Elements for approval.

3.12.4.3 Workmanship Standards

The developer shall use approved workmanship standards. Workmanship standards shall be referenced in fabrication, test and inspection procedures and shall be readily available in the production, test, and inspection areas. The developer may propose alternate standards for MDA/QS approval. Proposed standards shall be accompanied by objective data documenting that mission safety or reliability will not be compromised. As a minimum, the following processes shall be performed using an MDA/QS recognized workmanship standard.

a. Soldering, conformal coating, and staking (ANSI/J-STD-001, Class III and IPC/EIA J-STD-001CS).

b. Crimping, wiring, and harnessing (NASA-STD-8739.4).

c. Fiber optics assembly and repair (NASA-STD-8739.5).

d. ESD control (MIL-STD-1686).

e. Printed Board Design (IPC-2220 Series, Class III).

f. Printed Board Manufacturing (IPC-6011 Series, Class III).

3.12.5 Product Control during Fabrication

The developer shall establish and maintain a system for the control of parts, devices, and materials used throughout the fabrication process as required by AS9100 and as supplemented in the following paragraphs.

3.12.5.1 Product Identification and Handling

The developer shall establish and maintain a process for the identification and handling of parts, devices, and materials during fabrication. Controls shall ensure that:

a. Only authorized parts, devices, and materials that meet specified requirements are released to manufacturing operations.

b. Parts, devices, and materials excess to manufacturing operations are removed from the processing area and reviewed to determine need for re-inspection prior to returning to their respective stock points.

c. Parts, devices, and materials procured for the development phase are not installed in production end items without cognizant MDA Deputates and Elements approval.

d. Hardware items used as aids or tools are conspicuously marked to prevent installation in end items.

e. Deliverable hardware shall not be used as test, production, or troubleshooting aids.

f. Parts, devices, and materials sensitive to electrostatic discharge shall be identified and appropriate precautions incorporated into storage, handling, fabrication, and test operations.

3.12.5.2 Product Protection

As a supplement to the AS9100 requirements, the developer shall establish and maintain controls to ensure conditions required to attain the quality and reliability of the product are achieved and maintained.

Parts, devices, and materials subject to damage, deterioration, electrostatic discharge, contamination, shall be identified and protected throughout fabrication. Personnel working on MDA hardware shall use appropriate safeguards when handling hardware. Additionally, implementing procedures for parts, devices, and material protection shall, as a minimum, comply with specified material protection requirements for environment, cleanliness, and contamination control.

3.12.5.2.1 Electrostatic Discharge Controls

The developer shall establish and maintain an electrostatic discharge (ESD) control program in accordance with MIL-STD-1686. As a minimum, the ESD control program shall address training (3.1.8.1), protected work area procedures and verification schedules, packaging, facility maintenance, storage, and shipping.

3.12.5.2.2 Contamination Control Program

The developer shall establish and maintain a Contamination Control Program (CCP) appropriate for hardware. Contamination includes all materials of molecular and particulate nature whose presence degrades hardware performance. The source of the contaminant materials may be the hardware itself, the test facilities, and the environments to which the hardware is exposed. The program shall include a contamination control verification process, which considers the hardware's contamination sensitivity and allowance. The verification process along with the specific cleanliness requirements and approaches to be followed shall be documented in a CCP and made available to the cognizant MDA Deputates and Elements or designated representative upon request.

The CCP shall describe the methods used to measure and maintain the levels of cleanliness required throughout the item's life. Contamination control of hardware shall be compatible with the most contamination-sensitive components. The CCP shall include data on material properties, design features, test data, system tolerance of degraded performance, and methods to prevent degradation and allow for evaluation of contamination hazards.

3.12.5.2.2.1 Clean Rooms

The developer shall implement clean room standards, appropriate to product application and complexity, when handling contamination sensitive hardware. The contamination potential of material and equipment used in cleaning, handling, packaging, tent enclosures, shipping containers, bagging (e.g., anti-static film materials), and purging shall be described in detail for each subsystem or component at each phase of assembly, integration, test, and launch.

3.12.5.2.3 Foreign Object Elimination Program

The developer shall establish and maintain a Foreign Object Elimination (FOE) program, which systematically eliminates Foreign Object Damage and Debris (FOD) to preserve safety, quality, and reliability. The FOE program shall provide for a standardized approach that maintains awareness, prevention, compliance, and assures continued reinforcement. The FOE program shall also ensure operational processing areas maintain a safe, clean, and FOD-Free environment. The developer shall develop and maintain FOE program plan that specifies the requirements and techniques assuring FOE awareness and prevention. The FOE Program Plan shall be made available to the cognizant MDA Deputates and Elements or designated representative upon request.

3.12.5.3 Product Status Indication

The developer shall establish and maintain a system for product status indication that assures:

a. The inspection and test status of parts, devices, materials, and assemblies are clearly indicated throughout the entire fabrication cycle. Hard copy or electronic records indicating completion of all tests, inspections, and operations, which reference all discrepancy reports, shall be readily available in the area where the item is located.

b. The authority to designate parts, devices, materials, or assemblies as acceptable shall be restricted to authorized personnel. Records documenting each designation shall also provide traceability to the individual making the designation.

c. Stamps or other status indicators shall be of a design distinctly different from those used by the government. Where electronic, or alternate, methods of providing inspection status are used, access to government data fields shall be strictly controlled, allowing access by authorized government personnel only.

3.12.6 Fabrication Process Control

In addition to AS9100, the developer shall control fabrication processes in accordance with specified operating procedures. Process data shall be recorded and analyzed to ensure continued process control. The developer shall record process variables data necessary for analysis to determine trends and to maintain continued process integrity and control. Specific controls shall be consistent with:

a. The product characteristics and their associated tolerances, criticalities, sensitivity to process variation, inspectability, and testability.

b. The application and operational requirements of the product.

c. The extent and nature of subsequent test and inspection.

d. Operator skill required.

e. The results of process selection and development.

If it is determined that a process does not meet either specification or process control limits, the possible effect on items previously processed shall be determined and corrective action taken to ensure that items processed meet specification requirements or are identified as nonconforming (3.11). Documented notification shall be provided to the cognizant MDA Deputates and Elements whenever the analysis of process data indicates that the quality of processed items is in doubt.

3.12.6.1 Process Qualification Program

The developer shall implement a process qualification program to prove-in new or modified (e.g. material and process changes, technology insertion, redesign) fabrication processes, including test. Process qualification shall be performed using tools and equipment, software, personnel, material, and procedures used to fabricate and ensure the quality of the product. During process qualification, the developer shall identify and resolve potential fabrication process and product failure modes utilizing Process Failure Modes and Effects Analysis (3.5.12) to improve quality and reliability of the product. The developer shall re-qualify processes whenever a change, which may have an adverse affects on the product occurs. Process requalification is required for: material and process changes; tooling, dies, or fixture changes; equipment, procedure, or software changes; fabrication rate changes; relocation; and breaks in fabrication of greater than 6 months. Parts and Materials Control Board (3.6.2) shall approve the process qualification and requalification. The developer shall notify the cognizant MDA Deputates and Elements or designated representative of process qualification events to allow for participation. Process qualification events shall be documented and submitted to the cognizant MDA Deputates and Elements or designated representative for information.

3.12.6.2 Fabrication and Quality Metrics

The developer shall establish and maintain a process for the collection and analysis of fabrication and quality metrics. As a minimum, the set of metrics and frequency of collection should be representative of the development effort and all phases of the acquisition process. Examples of fabrication and quality metrics to be collected and analyzed include supplier quality data, in-process failures, test failure data,

and statistical process control data. The developer shall use fabrication and quality metrics as an input to the cost of quality metric (3.1.3.1.6).

3.12.6.3 Fabrication Defects

The developer shall detect, document, and correct (3.1.9) defects during fabrication and assess potential process improvement opportunities. When required, the developer shall conduct analysis meetings to determine defect root cause and take action to prevent recurrence. Data on defects as identified in inspections, document reviews, and testing shall be collected and analyzed by the developer. Defects shall not be reprocessed until they have been documented and dispositioned. The developer shall provide feedback on the status and results of defect preventive and corrective action to project personnel on a periodic basis. The developer shall use defect data as an input to the cost of quality metric (3.1.3.1.6).

3.12.6.4 Continuous Process Improvement

The developer shall establish and maintain a continuous process improvement program using proven industry techniques and statistical methods to reduce process variability and improve product quality. The developer shall determine key product characteristics and process parameters suitable for process control and monitor them using metrics. Process operations, parameters, and characteristics shall be determined on the basis of criticality, cost effectiveness, and technical considerations and included in the Transition to Operations or Production Plan (3.2.18.1). As a minimum, the continuous improvement program shall:

a. Provide a focus for product improvement through the identification sources of variation and key characteristics.

b. View the quality of a key characteristic as its conformance to nominal rather than merely achieving tolerance.

c. Locate a process on target with minimum variance.

d. Reduce the variation in key characteristics by: Improving the consistency of measurement systems; identifying, eliminating, and controlling sources of variation; and controlling the process rather than the product.

3.12.7 Fabrication Environmental Stress Screening

During fabrication, the developer shall implement the Environmental Stress Screening (ESS) program (3.5.13) to surface defects by stressing the item without degrading its inherent reliability. Environmental stresses may be applied in sequence or in combination, with the intent of stimulating hardware defects. ESS should not be used to simulate an operational environment. Results of ESS shall be used to continuously improve manufacturing processes.

3.12.8 Fabrication Quality Verification

The developer shall perform tests and inspections using documented procedures. Fabrication quality verifications, including in-process, acceptance, first article, and non-destructive tests and inspections, shall be performed to ensure conformance to product or process specifications. Personnel performing these verifications shall have the training, certification, independence, and authority to report problems and failures without concern for the cost, schedule, or technical implications of the reported problem or failure. When items rejected during fabrication quality verification are returned for completion of missed operations, rework, or repair, fabrication quality verification shall be accomplished not only for that specific characteristic but also for other characteristics that may be affected.

3.12.8.1 In-Process and Acceptance Test and Inspection

The developer shall perform in-process testing and inspection during fabrication to verify adequacy and control of operations. Tests and inspections used as a basis for government acceptance of contract end items shall be performed in accordance with procedures approved by cognizant MDA Deputates and Elements or designated representative. Tests and inspections shall:

a. Be performed at or before the last point at which the acceptability of the item or characteristic may be completely verified.

b. Provide a measure of product and process quality, which results in data suitable for analysis and timely correction of adverse quality trends.

c. Be performed in a manner and under conditions that simulate product end use to the highest degree practicable.

d. Be sufficient to provide assurance that the product conforms to specification requirements.

3.12.8.2 First Article Test and Inspection

First article test and inspection shall be performed on mission critical items (3.5.7) manufactured or purchased by the developer. First article test and inspection shall be conducted prior to initiation of a production run and on the first items produced using new or modified tooling or processes. First article test and inspection shall consist of a comprehensive test and inspection to verify production capability; proper use of materials, parts, and process controls; to demonstrate product compliance to specified requirements; and to verify the validity of applicable documentation.

3.12.8.3 Nondestructive Test and Inspection

Nondestructive tests and inspections shall be controlled by standards, specifications, and procedures, certification of personnel, and proper equipment controls.

3.12.8.4 Nonconforming Items Control

The developer shall control, review, and disposition nonconforming parts, devices, and materials used during fabrication in accordance with AS9100 and the requirements contained in 3.11.

3.12.9 Fabrication and Quality Records

The developer shall maintain fabrication and quality records in accordance with AS9100 and as supplemented in the following paragraphs. Additionally, fabrication and quality records shall be incorporated into the pedigree program (3.1.6).

3.12.9.1 Fabrication Records

The developer shall ensure that fabrication data, including defects, is recorded and retained in sufficient detail to indicate accountability for operations; provide for analysis to determine problem frequency and trends; and implement appropriate preventive and corrective actions. Records shall be traceable to specific personnel or equipment where personnel skills or equipment capability has a significant effect on product quality. The developer shall inform the cognizant MDA Deputates and Elements prior to disposal of fabrication records.

3.12.9.2 Quality Control Records

The developer shall maintain records of tests and inspections performed. Records shall be appropriate for the type, scope, and importance of the test or inspection performed and sufficiently detailed to provide objective evidence of conformance to requirements and to permit necessary analysis for further action.

Records shall include the inspection results, evidence of performance of required test or inspection, extent of nonconformance, disposition of nonconforming items, and responsibility for corrective action. Records for acceptance test and inspection shall include identification of specific equipment (e.g. model and serial number) used for acceptance so that recall of accepted products may be accomplished when out of tolerance conditions are noted during subsequent calibrations. The developer shall inform the cognizant MDA Deputates and Elements prior to disposal of quality control records.

3.12.10 Packaging, Handling, Storage, and Transportation

The developer shall establish and maintain a system for packaging, handling, storage, and transportation of product. The system shall comply with product specifications and regulations, and include documented procedures to prevent product degradation.

3.12.10.1 Packaging

The developer shall perform preservation packaging, packing, and marking processes (including materials used) in accordance with the item specification and system requirements.

3.12.10.2 Handling and Storage

The developer shall establish and maintain processes and procedures for handling and storage of product to prevent deterioration. The handling and storage procedures shall be adhered to and identified on fabrication documentation. The following criteria shall be used, as appropriate, for establishing handling and storage procedures for product:

a. Control of environment, such as temperature, humidity, contamination, and pressure.

b. Measures and facilities to segregate and protect product routed to different locations such as, to the materials review crib, or to a laboratory for inspection, or returned to the manufacturer from unaccepted shipments.

c. Easily identifiable containers, to identify products, shall be used.

d. Control measures to limit personnel access to product during receiving inspection and storage.

e. Facilities for interim storage of product.

f. Provisions for protective cushioning, as required, on storage area shelves, and in storage and transportation containers.

g. Protective features of transportation equipment design to prevent product from being dropped or dislodged in transit.

h. Protective bench surfaces on which product are handled during operations such as test, assembly, inspection, and organizing kits.

i. Required use of gloves, finger cots, or other means when handling product to protect it from contact.

j. Electrical, Electronic, and Electromechanical parts shall be kept in a temperature and humidity controlled environment to prevent the absorption of moisture. Plastic encapsulated devices are to be handled in such a way as to minimize moisture absorption and ionic contamination.

k. Products sensitive to electrostatic discharge shall be identified and appropriate precautions incorporated into storage, handling, fabrication, test, and shipping operations.

l. Unique product criteria.

3.12.10.3 Preparation for Shipment and Transportation

The supplier shall arrange for the protection of the quality of product after final inspection and test. Where contractually specified, product protection shall be extended to include delivery to destination.

Items shall be identified and packaged in accordance with contractual requirements and documented procedures. The developer shall inspect and control items being prepared for shipment and transportation to ensure that:

a. Items have satisfactorily passed applicable inspections and tests.

b. Items have been identified, preserved, packaged, and packed in accordance with applicable specifications and procedures.

c. Packaging and containers have been marked in accordance with applicable drawings, specifications, and procedures.

d. Environmental conditions of shipping containers are monitored during shipment, as appropriate.

The developer shall ensure that the accompanying documents for the product are present at delivery as specified in the contract/order and are protected against loss and deterioration.

3.13 Supplier Management

The supplier management program shall comply with AS9100 requirements and the following provisions. The developer shall establish and maintain a supplier management program to ensure selection of suppliers capable of attaining program cost, schedule, and technical objectives during development and production phases. The program shall focus on satisfying customer requirements pertinent to supplier management. Requirements include appropriate flow down requirements such as program quality, program unique terms and conditions, and any metrics to be communicated, tracked, and monitored during the planning and execution phase of the supplier management process. Throughout the acquisition process, the supplier management program shall be focused on providing effective material for deployment. A supplier base shall be developed that is capable and reliable in providing best value products and services.

3.13.1 Supplier Selection

The developer shall establish and maintain a process for evaluation and selection of procurement sources. The developer shall evaluate and select sources based on their technical capability and capacity to supply products (hardware, software, and services) with acceptable levels of quality and reliability in accordance with program requirements. Criteria for selection, evaluation, and re-evaluation shall be established and include, as a minimum:

a. Development, manufacturing, and verification capability.

b. Past performance/quality history, including field data.

c. Source inspection, receiving inspection, and test results.

d. On-time delivery performance.

e. Corrective action responsiveness.

f. Life cycle support processes.

g. Financial and organizational stability.

h. Fact finding visit results.

Records shall be supported by documented quantitative information. An onsite survey of the supplier's capabilities, facilities, and technical management program shall be conducted if no previous quality and reliability records are available, or if supplier performance has been marginal based on supplier ratings. Results of this survey and subsequent corrective action shall be documented and maintained.

The developer shall select suppliers based on overall best value in terms of performance, risk factors (e.g. reliability, technology, diminishing sources), cost or price, and quality factors.

The performance of each supplier shall be objectively evaluated by the developer on a continuing basis utilizing data from source inspection, receiving inspection, qualification, fabrication, assembly, acceptance test and inspection, on-site surveys, audits, field use, engineering and qualification, alerts, and any other available quality data. The developer shall periodically evaluate the supplier's financial and organizational stability and perform a study for qualifying other suppliers when necessary. Based on this evaluation, the developer shall prepare, maintain, and use approved source lists, or equivalent, organized by supplier, facility location, and each product type or service, and its intended application. Criteria for maintenance of the approved source list, including addition and removal of suppliers shall be documented. Records of selection, evaluation, and approval shall be maintained and made available to the cognizant MDA Deputates and Elements or designated representative upon request. The developer

shall maintain justification records for their selection of sole source suppliers. The developer shall consider the cost effectiveness of qualifying multiple sources for critical components.

3.13.1.1 Conditional Source Approval

The developer shall establish and maintain a process for approval of sources for emergency or conditional procurement to be used in deliverable product. Procurements from sources other than those on approved source lists shall not be made without appropriate engineering and quality review and written approval from program management. If use of an unapproved supplier is necessary on a conditional basis, steps shall be taken promptly to approve the source. The system shall include a process for storage, identification, tracking, and traceability of supplies from unapproved sources. The system shall prevent product from being shipped or presented to the government for acceptance until the supplier is approved. In the event the supplier fails to qualify, suitable corrective action shall be initiated prior to any subsequent purchase. Receiving inspection or validation for software shall be performed to ensure items procured on a conditional basis conform to purchase order, specification, and drawing requirements. Satisfactory performance of supplies purchased from an unapproved supplier shall not, in and of itself, constitute qualification of that supplier.

Purchases from unapproved sources may be made if material purchased is not included in deliverable product.

3.13.2 Supplier Ratings

The developer shall establish and maintain a supplier rating system that uses a continuous and standardized methodology for monitoring, evaluating, and improving supplier performance. The supplier rating system shall define minimum acceptable rating criteria for hardware, software (3.3.1.7), and services. The rating system shall be based on quality, delivery performance factors, and other sub-factors such as post acceptance events and responsiveness to corrective action requests. Factory failures and field data shall directly affect a supplier's overall rating. The system shall use a centralized repository that includes both historical and current data on supplier performance. Ratings shall be based on both recent and cumulative performance rather than solely on short-term windows of reference. The developer shall communicate ratings to their suppliers as part of the continuous improvement process to enable the supplier to proactively self-manage and improve performance. A poor quality rating shall prohibit the placing of any purchase order without further investigation, satisfactory corrective action from the supplier, and approval from top level management.

3.13.3 Supplier Evaluations

The developer shall establish and maintain a system to schedule and conduct on-site evaluations to ensure compliance with procurement document requirements. Evaluations shall provide the developer with an opportunity for monitoring ongoing effectiveness of suppliers Quality, Safety, and Mission Assurance performance with respect to flow down requirements. Evaluations also provide the developer an opportunity to reinforce with the supplier, the importance of good performance. The frequency, scope, and method for evaluating shall be based upon criticality or complexity of items being procured, known problems or difficulties, documented risks, and quality history. The planned coverage of each evaluation shall be documented. Coverage shall include examination of applicable program requirements, operations, parts, devices, materials, software, and documentation to determine compliance with established requirements. The developer shall document the rationale for reductions in frequency or scope of evaluation. Results of evaluations, with recommendations for corrective action, shall be documented and made available to the cognizant MDA Deputates and Elements or designated representative upon request. Follow-up shall be performed to verify that effective corrective action has been taken. The developer shall allow for cognizant MDA **Deputates and Elements** or designated representative access (3.1.13) and participation in supplier evaluations of hardware, processes, and software suppliers.

3.13.4 Supplier Program Requirements

The developer shall establish and maintain a system for specifying applicable program requirements and MDA Assurance Provisions (MAP) to suppliers. The system shall provide criteria for selection and flowdown of MAP requirements defined in the Mission Assurance Implementation Plan for specific program phases based on considerations of item complexity and criticality. The developer shall specify in the procurement document applicable MAP requirements defined in the Mission Assurance Implementation Plan to be imposed on the supplier. Suppliers shall in turn impose requirements on their procurement sources.

3.13.5 Procurement Process

The developer shall establish and maintain a process for generation, review, and release of procurement documents. This process shall be used to satisfy the responsibility of the developer for assuring supplier conformance to current configuration requirements. These controls shall be applied uniformly to all applicable suppliers, and they shall include provisions for assurance of mutual notification of changes, verification of incorporation of changes, and identification of hardware, software, and services involved.

3.13.5.1 Technical Requirements

Procurement documents shall include AS9100, paragraph 7.4 Purchasing, requirements and the following supplemental technical requirements:

a. Interface, special tooling, and test and measuring equipment requirements.

b. Specifications for special preservation and packaging required.

c. Requirements for the supplier to notify the developer of any proposed changes to developer approved design, parts, devices, materials, or fabrication methods or processes, and to obtain developer approval prior to change incorporation.

3.13.5.2 Detailed Provisions

The developer shall include the following statements or equivalent in the procurement document, as applicable:

a. *Government Source Inspection (GSI)*. GSI is required prior to shipment from your plant. Upon receipt of this order, promptly notify the government representative who normally services your plant so that appropriate planning for government inspection can be accomplished.

b. *Procurements Not Requiring GSI*. The government has the right to inspect any or all of the work included in this order at the supplier's facility.

c. *Developer Source Inspection*. A developer source inspection statement when source inspection is to be utilized.

d. *Raw Materials*. Chemical and physical test results shall be submitted or a certificate of compliance. Purchased raw materials, which are required to satisfy documented specifications shall be accompanied by a detailed analysis report.

e. *Raw Materials Used in Purchased Items*. Records of detailed results of chemical and physical analyses of acceptance test results on raw materials that are required to satisfy specification requirements employed in the manufacture of articles purchased on this contract or purchase order shall be maintained by the supplier and made available upon request.

f. *Process Control and Inspection*. Evidence of process controls and specific tests or inspections shall be provided to (developer). Records shall be maintained by (supplier), adequate to ascertain the quality level of production processes.

g. *Limited-Life Items*. Items determined to have characteristics susceptible to quality degradation with age or storage environment shall be marked in a manner to indicate date of manufacture, date at which useful life was initiated and will expire, and specific storage environmental restrictions.

h. *Resubmission of Rejected Items*. All items rejected by (developer) and subsequently resubmitted by (name of supplier) shall bear an adequate indication of such resubmission on those items or on the shipping document. Reference shall be made to the (developer) rejection document and evidence given that the causes for rejection have been corrected and actions taken to preclude recurrence.

i. *Certification of Manufacturer*. All items to be submitted by (name of distributor) shall be accompanied by a certification of the name and location of the item manufacturer.

j. *Supplier Requirements and Review*. The (supplier) shall, in the performance of the contract or purchase order, provide and maintain a program which is in conformance with the following applicable program and QSMA requirements (attached). (Cognizant MDA Deputates and Elements), MDA/QS, and (developer) may review (supplier) facilities to establish conformance to applicable program requirements.

k. *Product Changes*. The supplier shall notify (developer) of proposed changes to products including changes in design, fabrication methods or processes, materials, and changes, which may affect the quality or intended end use of the item. The supplier shall submit these changes to (developer) for processing and approval.

3.13.5.3 Procurement Document Review

The developer shall establish and maintain a process for independent (e.g., engineering, quality) technical review to ensure procurement documents are complete and correct. This review shall be accomplished prior to release of the purchase order and shall ensure that:

a. Appropriate program requirements are specified.

b. Technical requirements are included.

c. Applicable detailed provisions are specified.

d. The supplier is an approved source or that provisions to perform necessary tests and inspections are planned.

e. Applicable qualification requirements are satisfied.

Procurement documents and referenced data shall be made available to the government representative for review to determine compliance with contract requirements and need for government inspection at supplier facilities. These documents shall be furnished in accordance with instructions from the government representative.

3.13.5.4 Procurement Document Change Control

The developer shall provide for control and approval of changes to drawings, test procedures, specifications, and other procurement documents, and for incorporation of approved changes. For items procured to developer design, control shall include assurance of notification of change to the supplier, verification of the incorporation, and appropriate identification of those items on which the change is incorporated. When supplier design, fabrication methods, or processes have been approved or qualified

by the developer, controls shall be established to monitor and approve supplier notices of proposed changes.

3.13.6 Control of Customer/Government Furnished Material

The developer shall establish and maintain documented procedures to control receipt, verification, handling, preservation, storage, and maintenance of customer/government supplied material provided for incorporation into end items or for related activities. Any such material that is lost, damaged, or is otherwise unsuitable for use shall be recorded and reported to the customer/government. The developer shall verify quality of supplied items and services by performing inspections and tests either upon receipt at the developer's facility or at the supplier facility. Verification by the developer does not relieve the customer/government of the responsibility to provide acceptable material. When the overall system includes components or subsystems furnished by the government, the developer shall be responsible for obtaining from the government adequate reliability data on the items. When the developer's examination of data or testing indicates that Government Furnished Material reliability is inconsistent with overall system requirements, the cognizant MDA Deputates and Elements shall be formally and promptly notified.

3.13.7 Government Source Inspection

The government reserves the right to inspect, at the source, items not manufactured or services not performed at developer facilities. GSI performed at supplier facilities on items or services, shall not ordinarily constitute acceptance, replace developer inspection, nor in any way release the developer from his responsibilities for assuring quality of these articles. However, when direct shipments from supplier facilities are specified, GSI and acceptance may be performed at supplier facilities. GSI and acceptance can only be requested by or under authorization of the cognizant MDA Deputates and Elements or its delegated technical representative.

3.13.8 Developer Source Inspection

The developer shall ensure that suppliers comply with requirements of procurement documents by means of developer source inspection at the supplier's facility, when appropriate. The system shall include requirements for documenting, collecting, and submitting source inspection and surveillance procedures and data. In addition, records of inspections and tests witnessed by the source inspector, including quantities witnessed and nonconformance data, disposition made of nonconforming items, and corrective actions required of suppliers shall be maintained. Periodic reports from the source inspector to the developer concerning supplier operations monitored, including problems found and corrective actions taken, shall be provided.

Source inspection shall be performed when any of the following conditions apply:

a. Items are being procured at a level of assembly that prevents verification of quality at developer facilities.

b. Manufacturing processes have an effect on the item such that quality cannot be determined solely by examination or test of the completed item at developer facilities.

c. Destructive tests are necessary at supplier facilities.

d. Special test and inspection equipment and environments required cannot feasibly and economically be reproduced or made available at developer facilities.

e. Shipments of completed items are made to destinations other than the developer facility.

3.13.9 Receiving Inspection and Test

The developer shall establish and maintain a receiving inspection and test system in compliance with AS9100 and supplemented by the following:

a. Inspection and test of purchased items, including COTS/NDI, to verify compliance with specification and drawing requirements. The degree of inspection and testing performed shall be governed by article complexity; results from supplier, source, and previous receiving inspections; and product quality history.

b. Adequate equipment and instructions are available to perform tests and measurements.

c. Items have passed qualification, requalification, or first article tests.

d. Verification that required tests and inspections by the supplier have been performed, that processes are controlled, and that required data have been provided. The developer shall periodically validate test reports.

e. Procured articles subject to age deterioration are clearly marked and supported by information regarding life expiration date and need for any special environmental controls.

f. Prompt inspection of GFM including provisions for prompt feedback to the procuring contracting officer when nonconforming GFM is found.

g. Proper handling of purchased articles, including segregation and identification of those items awaiting inspection or test, those which have been accepted, those which have been rejected, and those awaiting material review action.

h. Evidence that required source inspection has been performed, and required data submitted.

i. Use of appropriate sampling plans, if applicable, including provisions for reduced and tightened inspection.

When the developer has a dock-to-stock program, the items shall be positively identified to permit recall in the event of nonconformity to specified requirements.

3.13.10 Intra-Corporate Work Transfers

Intra-corporate work transfers shall reflect prime contract program requirements, or the assigned corporation element shall be treated as a supplier and the provisions of supplier management shall apply.

3.14 Safety

Developers shall establish and maintain a system safety program, which ensures system safety throughout all phases of the system's life. The program shall apply engineering and management principles, criteria, and techniques to optimize all aspects of safety within the constraints of operational effectiveness, schedule, and cost. The system safety program shall ensure:

a. Safety, consistent with mission requirements, is designed into the system in a timely, cost-effective manner minimizing retrofit actions.

b. Hazards are identified, evaluated, and eliminated, or the associated risk reduced to a level acceptable to MDA and cognizant MDA throughout the entire life of a system. Actions taken to eliminate hazards or reduce risks are verified and documented.

c. Historical safety data, including lessons learned from other systems, are considered and used.

d. Safety risks are considered in accepting and using new designs, materials, and production and test techniques.

e. Changes in design, configuration, or mission requirements are accomplished in a manner that maintains a safety risk level acceptable to MDA and cognizant MDA.

The requirements of this document shall not exempt programs from meeting service specific safety requirements levied upon them or any other safety requirements imposed by law.

3.14.1 Safety Program Requirements

The developer shall establish and maintain a system safety management and engineering program in accordance with MIL-STD-882. The program shall include planning, analysis, disposition/reporting, sustained engineering, safety working groups, hazard tracking, assessments, reviews, audits, and verifications.

Developers and suppliers shall comply with MDA safety policies and requirements. In addition, developers shall:

a. Ensure that a systematic hazard analysis process is conducted and documented in accordance with MIL-STD-882. The safety hazard analysis process shall include system and sub-system hardware, software, the environment (in which the system will exist), and the intended use or application over the product's life cycle. The safety hazard analysis shall document and disclose known safety defects and deficiencies associated with the element/program.

b. Establish and maintain policies and procedures for formal review and approval of safety risk assessments.

c. Support system safety working groups.

d. Coordinate safety risk assessments with other internal engineering disciplines as well as MDA/QS and system-level safety organizations to ensure that safety risks are properly identified and documented.

e. Maintain and report safety metrics (3.14.21).

f. Document, contribute, and use safety related lessons learned to enhance safety throughout MDA using the MDA Lessons Learned (MDALL) Program.

g. Designate a qualified safety representative with specific responsibility for coordinating and executing a safety program within the developer's scope of responsibility. This representative shall have the requisite training and experience to assess and analyze safety issues. The qualified safety representative is the person who has supervisory responsibility/technical approval authority for the system safety work. This safety representative shall have, at a minimum, a BS degree in engineering, physics, mathematics, or other scientific/technical disciplines and a minimum of four years of system safety or related experience.

h. Ensure the prompt and accurate reporting, investigating, tracking, and closure of all mishaps, close calls, problems, nonconformances, and anomalies within the program that are safety related in accordance with written MDA policy.

i. Ensure all personnel understand that they are authorized to suspend any activity that presents an immediate and unacceptable danger to personnel, property, or operations without retribution.

j. Ensure appropriate corrective actions have been implemented prior to restarting any activity that has been suspended due to unacceptable danger to personnel, property, or operations.

k. Ensure compliance with all applicable Range Safety and Service Safety requirements.

l. Ensure a safety impact analysis is conducted on all software changes, requests for variances (waivers or deviations), and ECPs for engineering baselines under configuration management (3.10).

3.14.2 Safety Program Documentation, Reports, and Working Groups

The following minimum safety documentation shall be prepared by all developers and submitted to cognizant MDA for approval and MDA/QS for information.

a. System Safety Program Plan.

b. System Safety Hazard Analysis Reports (DI-SAFT-80101B).

c. Safety Assessment Reports (3.14.2.8).

d. Waiver or Deviation System Safety Report (DI-SAFT-80104B).

e. Engineering Change Proposal System Safety Report (DI-SAFT-80103B).

f. Integrated System Safety Program Plan.

g. Safety Metrics.

h. Health Hazard Assessment Report (DI-SAFT-80106B)

3.14.2.1 System Safety Program Plan

The developer shall establish and maintain a System Safety Program Plan (SSPP). The SSPP shall be submitted to the cognizant MDA for approval and MDA/QS for information. As a minimum, the SSPP shall:

a. Describe the developer's implementation of System Safety requirements defined herein. Describe the tasks and activities of system safety management and engineering and the interrelationship between system safety and other functional elements of the program. Identify each hazard analysis and mishap risk assessment process that will be used.

b. Describe the methods used to identify and apply safety/hazard control requirements and criteria for design of equipment, software, facilities, and procedures during the product's life.

c. Specify the analysis technique(s) and format used in qualitative and quantitative analysis to identify hazards, their causes and effects, and recommended corrective action.

d. Specify the depth within the system to which each analysis technique will be used including hazard identification associated with system, subsystem, components, software, personnel, ground support equipment, Government Furnished Equipment, facilities, and their interrelationship in the logistic support, training, maintenance, transportability, security, and operational environments.

e. Specify the integration of supplier hazard analyses and techniques with overall system hazard analyses.

f. Specify the technique for tracking hazards in a single closed-loop system.

g. Define how hazards and residual mishap risk are communicated to and accepted by the appropriate risk acceptance authority and how hazards and residual mishap risk will be tracked.

h. Specify techniques and procedures used to ensure objectives and requirements of system safety program are included in safety training for engineers, technicians, programmers, and testing, operating and maintenance personnel.

i. Specify safety techniques and procedures employed to ensure that the objectives and requirements of the system safety program are accomplished.

j. Define the mishap analysis process, including MDA and cognizant MDA notification.

k. Include an item-by-item accounting of all contractually required system safety requirements, tasks, and responsibilities.

l. Include information on system safety integration into the overall program structure.

m. Include system safety organization or function within the organization responsible for System Safety, its functional relationships, and lines of communication.

n. Include responsibility, authority, and accountability of system safety personnel, other developer organizational elements involved in the system safety effort, suppliers, and system safety groups.

o. Include organizational unit responsible for executing each system safety task, the position with the authority to resolve all identified hazards, and System Safety Program Manager contact information.

p. Include staffing of the system safety organization for the duration of the contract/development agreement to include manpower loading and the qualifications of assigned personnel.

q. Include process through which management decisions will be made to include notification of critical and catastrophic hazards, corrective action taken, mishaps or malfunctions, waivers to safety requirements, and program deviations.

r. Include verification requirements for ensuring safety.

s. Include procedures for ensuring:

   1) Feedback of test information for review and analysis on impact to safety.

   2) Safe conduct of all tests.

3) Hazards identified have been eliminated or controlled to an acceptable level of risk.

t. Include an integrated system safety schedule that supports the programs' engineering and programmatic milestones.

u. Include description of:

1) Approach for identifying, obtaining, researching, disseminating, and analyzing pertinent historical hazard or mishap data.

2) Interfaces between system safety and all other applicable safety disciplines, such as Nuclear Safety, Range Safety, Explosive and Ordnance Safety, Chemical and Biological Safety, Occupational Safety and Health, Laser Safety, Software Safety.

3) Interfaces between system safety and all other support disciplines, such as Maintainability, Quality Assurance, Security, Reliability, Human Factors Engineering, Transportability Engineering, and Medical Support (Health Hazard Assessments).

4) Procedures used to integrate and coordinate the system safety efforts including dissemination of the system safety requirements to action organizations and suppliers, coordination of supplier's system safety programs, integration of hazard analyses, program and design reviews, program status reporting, and system safety groups.

3.14.2.2 Safety Analyses

The developer shall perform safety analyses, which support cognizant MDA Deputates and Elements and BMDS safety assessments. Safety analyses shall:

a. Identify safety critical functions. For each safety critical function, the developer shall establish a process for analysis, design, test, and verification and validation for those functions.

b. Include tailoring and communication of generic safety related requirements and constraints to the system and software designers early in the acquisition process.

c. Identify, document, and track system and subsystem level hazards and their effects, including the human as an element of the total system.

d. Categorize each and every identified hazard in terms of severity and probability of occurrence (specify qualification or quantification of likelihood).

e. Identify each failure path and associated causal factors. This analysis shall be to the functional depth necessary to identify logical, practical, and cost-effective mitigation techniques for each failure path initiator (causal factor). This analysis shall consider all hardware, software, and human factor interfaces as potential contributors.

f. Derive safety-specific hazard mitigation requirements to eliminate or reduce the likelihood of each causal factor.

g. Provide engineering evidence (through appropriate inspection, analysis, demonstration, and test) that each mitigation safety requirement is implemented within design and system functions to meet safety goals and objectives. Any residual mishap risk shall be documented. All new hazards identified during testing shall be reported to cognizant MDA Deputates and Elements and MDA/QS.

h. Evaluate all hardware and software changes and defects for their potential safety impact.

i. Communicate a safety assessment of all residual safety risk after all design, implementation, and test activities are complete.

3.14.2.3 Safety Variance Reporting

Proposed variances (waivers or deviations) affecting safety requirements or certification criteria shall be submitted to cognizant MDA Deputates and Elements, MDA/QS, and appropriate safety review boards for review and comment prior to submittal to the variance approval authority. The appropriate authority shall disposition variances affecting safety requirements. Approval / disapproval of variances affecting safety requirements shall be reported to cognizant MDA Deputates and Elements, MDA/QS, and appropriate safety review boards following their disposition by the approval authority.

3.14.2.4 Safety Assessment Reporting

The developer shall ensure that results of safety assessments and supporting data are submitted to the cognizant MDA Deputates and Elements, MDA/QS, and appropriate safety review boards.

3.14.2.5 Sustained Engineering

Developer shall conduct a safety impact assessment on all hardware and software change requests, variances (waivers or deviations), and ECPs for products under configuration control. The developer shall prepare and submit Hazard Analysis Reports and/or Safety Assessment Reports associated with ECPs and variances to the cognizant MDA Deputates and Elements for approval and MDA/QS for information.

3.14.2.6 System Safety Working Groups

The developer shall form working group(s) with the government to address all aspects of safety including, but not limited to: system safety, test and evaluation safety, software safety, range safety, and occupational safety and health.

3.14.2.7 Hazard Identification and Tracking

The developer shall maintain and provide the cognizant MDA Deputates and Elements and MDA/QS access to the tracking system with current safety data including hazards, their closures, and residual mishap risk throughout the system life cycle.

3.14.2.8 Safety Verification

The developer shall verify the mishap risk mitigation through appropriate analysis, testing, or inspection. The developer shall document the residual mishap risk and shall report all new hazards identified during testing to the cognizant MDA Deputates and Elements and MDA/QS via system safety hazard analysis report.

3.14.2.9 Safety Assessment

A safety assessment shall be performed and documented by the developer to give a comprehensive evaluation of the residual mishap risk prior to test or operation of a system, prior to the next contract phase, or at contract completion.

Safety assessment shall also be performed and documented to identify all safety features of the hardware, software, and system design and to identify procedural, hardware, and software related hazards that may be present in the system being acquired including specific procedural controls and precautions that should be followed. The safety assessment report shall summarize:

a. Safety criteria and methodology used to classify and rank hazards, plus any assumptions on which the criteria or methodologies were based or derived. Classification shall be accomplished in accordance with 3.14.3.4, Risk Acceptance Authority.

b. Results of analyses and tests performed to identify hazards inherent in the system, including:

1) Hazards having residual safety risk.

2) Actions that have been taken to mitigate or eliminate hazards.

3) Results of tests conducted to validate safety criteria, requirements, and analyses.

c. Results of safety program efforts. Include a list of all hazards along with specific safety recommendations or precautions required to ensure safety of personnel, property, or environment. Categorize the list of hazards as to whether or not they may be expected under normal or abnormal operating conditions.

d. Any hazardous materials generated by or used in the system, including:

1) Identification of material type, quantity, and potential hazards.

2) Safety precautions and procedures necessary during use, packaging, handling, storage, transportation, and disposal (e.g., explosive ordnance disposal). Include all explosive hazard classifications.

3) Post launch safety related activity of expendable launch vehicles and their payloads including deployment, operation, reentry, and recovery (if required) of launch vehicles/payloads which do not attain orbit (either planned or unplanned).

4) Orbital safety hazard awareness associated with space systems such as explosions, electromagnetic interference, radioactive sources, ionizing radiation, chemicals, space debris, safe separation distances between space vehicles, and natural phenomena.

5) A copy of the Material Safety Data Sheet (OSHA Form 174, or equivalent manufacturers format).

e. Conclude with a signed statement that all identified hazards have been eliminated or their associated risks controlled to levels acceptable to MDA and the cognizant MDA Deputates and Elements and that the system is ready to test, operate, or proceed to the next acquisition phase. In addition, the developer shall make recommendations applicable to potential hazards at the interfaces with other BMDS programs.

3.14.2.9.1 Safety Defect / Deficiency Assessment

All known hardware and software defects / deficiencies shall be reviewed for potential safety implications. If safety impacts are identified, the developer shall notify MDA and the cognizant MDA Deputates and Elements of a decrease in the level of safety of the system. Defects / deficiencies impacting safety shall be included in the Hazard Analyses and Safety Assessment Reports.

3.14.2.10 System Safety Program Reviews/Audits

These requirements supplement the requirements in 3.1.7, Internal Evaluation Program. Developers shall perform and document system safety program reviews/audits and support MDA evaluations (3.1.13). These reviews/audits shall be performed on:

a. The developer's system safety program(s).

b. The supplier's system safety program(s).

The developer shall support presentations to government certifying/assessment activities such as phase safety reviews, munitions safety boards, or flight safety review boards. These may also include special reviews such as flight/test readiness reviews.

Desk audits, peer reviews, static and dynamic analysis tools and techniques, and debugging tools shall be used to verify implementation of design requirements in the source code with particular attention on implementation of identified safety critical computing system functions. Reviews of the software source code shall ensure that the code and comments within the code agree.

3.14.3 System Safety Requirements

Developer shall identify and understand known hazards and their associated risks. A systematic approach of hazard analysis and safety risk management in accordance with MIL-STD-882 shall be used by the developer to achieve acceptable safety risk. The developer shall identify and establish potential mishap risk mitigation alternatives and the expected effectiveness of each alternative or method.

The order of precedence for system safety hazard control shall be as follows:

a. *Eliminate Hazards Through Design Selection*. If unable to eliminate an identified hazard, the developer shall reduce the associated mishap risk to an acceptable level through design selection.

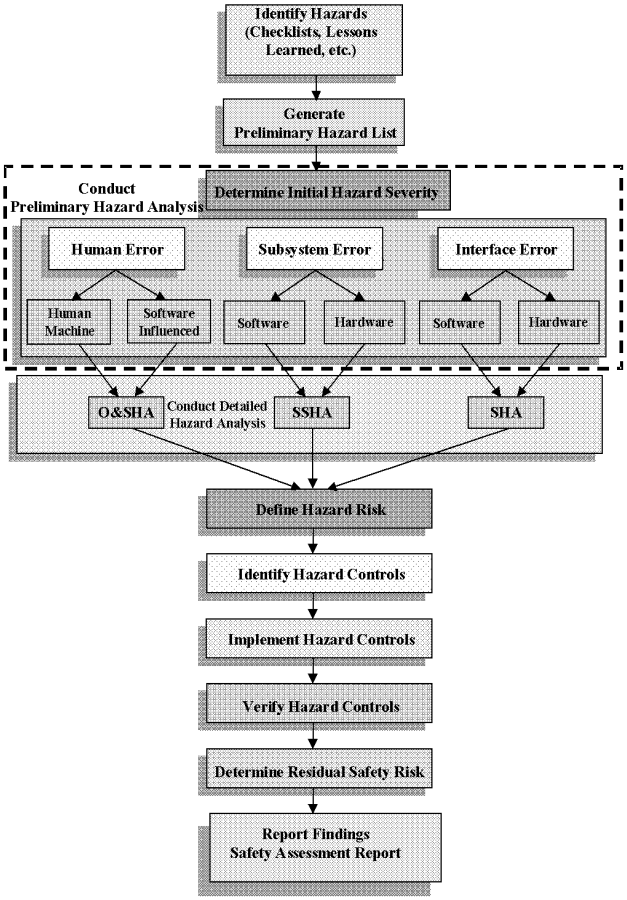
b. *Incorporate Safety Devices*. If unable to eliminate the hazard through design selection, the developer shall reduce the mishap risk to an acceptable level using protective safety features or devices.

c. *Provide Warning Devices*. If safety devices do not adequately lower the mishap risk of the hazard, the developer shall include a detection and warning system to alert personnel to the particular hazard.

d. *Develop Procedures and Training*. Where it is impractical to eliminate hazards through design selection or to reduce the associated risk to an acceptable level with safety and warning devices, the developer shall incorporate special procedures and training. Procedures shall include the use of personal protective equipment. For hazards assigned “Catastrophic” or “Critical” mishap severity categories, the developer shall not use “warning”, “caution”, or “other written advisory” as the only risk reduction method.

3.14.3.1 System Safety Engineering Approach

The system safety program shall follow the general process outlined in the figure below.



3.14.3.2 System Safety Hazard Identification and Analysis Methodology

Safety analyses shall be performed by the developer to identify hazards through a systematic hazard analysis process encompassing detailed analysis of system hardware and software, the environment (in which the system will exist), and the intended use or application. The developer shall consider and use historical hazard and mishap data, including lessons learned from other systems. During hazard identification, consider hazards that could occur over the system life cycle.

The hazard analysis methodology shall identify and document the specific elimination, mitigation, or control requirements to ensure that the residual safety risk is acceptable to MDA and the cognizant MDA Deputates and Elements. For every hazard causal factor identified that increases the potential for mishap, there shall be specific mitigation planning identified to successfully control the mishap/hazard to acceptable levels. In most instances, there will be multiple control requirements identified, analyzed, and tested. These hazard control requirements shall be prioritized in accordance with the hazard control order of precedence defined in paragraph 3.14.3, (a) through (d). In some instances, causal factors will be identified as and categorized as low risk safety related issues that fail in a safe condition. Although these are valid safety considerations, they may not have specific mitigation plans assigned to control the hazard. For these cases, acceptance of risk by the proper authority may be the only action taken.

3.14.3.3 Assessment of Mishap Risk

The developer shall assess the severity and probability of the mishap risk associated with each identified hazard. The Risk Acceptance Authority paragraph shall be used to determine the potential negative impact of the hazard on personnel, facilities, equipment, operations, and the public, as well as on the system.

3.14.3.4 Risk Acceptance Authority

Developers shall use the following criteria to determine the proper MDA authority for accepting residual safety risks.

Note: MDA acceptance of these risks does not imply that Test Range(s) will accept these risks for tests. Programs must work with the Test Ranges to ensure that they are willing to accept the risks as well.

The hazard/mishap severity categories are defined below. To provide a means to assess safety risks of damage to equipment, cognizant MDA Deputates and Elements shall add monetary loss criteria to severity categories 2, 3, and 4. MIL-STD-882 should be used as a guide for determining the monetary loss criteria, but may be tailored to fit the program.

a. *Catastrophic (Category 1).* Could result in death, permanent total disability, or loss of system. In the context of production operations, Category 1 Severity includes loss of capability to manufacture.

b. *Critical (Category 2).* Could result in permanent partial disability, injuries or occupational illness that may result in hospitalization of at least three personnel.

c. *Marginal (Category 3).* Could result in injury or occupational illness resulting in one or more lost workdays(s).

d. *Negligible (Category 4).* Could result in injury or illness not resulting in a lost workday.

The hazard/mishap probability levels are defined below. The developer shall define the expected size of the fleet or inventory and the expected lifetime of the item.

a. *Frequent (Level A).* Likely to occur often in the life of an item, with a probability of occurrence greater than 10^{-1} in that life (continuously experienced).

b. *Probable (Level B).* Will occur several times in the life of an item, with a probability of occurrence less than 10^{-1} but greater than 10^{-2} in that life (will occur frequently).

c. *Occasional (Level C).* Likely to occur some time in the life of an item, with a probability of occurrence less than 10^{-2} but greater than 10^{-3} in that life (will occur several times).

d. *Remote (Level D).* Unlikely but possible to occur in the life of an item, with a probability of occurrence less than 10^{-3} but greater than 10^{-6} in that life (will occur several times).

e. *Improbable (Level E).* So unlikely, it can be assumed occurrence may not be experienced, with a probability of occurrence less than 10^{-6} in that life (unlikely to occur, but possible).

TABLE 1. Safety Risk Acceptance Levels

HAZARD/MISHAP CATEGORY FREQUENCY	(1) CATASTROPHIC	(2) CRITICAL	(3) MARGINAL	(4) NEGLIGIBLE
(A) FREQUENT	1A	2A	3A	4A
(B) PROBABLE	1B	2B	3B	4B
(C) OCCASIONAL	1C	2C	3C	4C
(D) REMOTE	1D	2D	3D	4D
(E) IMPROBABLE	1E	2E	3E	4E

Hazard/Mishap Risk Index	Safety Risk Acceptance Authority
High: 1A, 1B, 1C, 2A, 2B,	MDA/D
Serious: 1D, 2C, 3A, 3B	MDA/DX
Medium: 1E, 2D, 2E, 3C, 3D, 3E, 4A, 4B	PD and/or RTO
Low: 4C, 4D, 4E	PD and/or RTO

3.14.3.5 Mishap Investigations

Developers shall investigate and report mishaps involving MDA programs in accordance with written MDA policy. Developers shall support mishap investigations as required.

3.14.4 Safety Design Criteria

The following design requirements supplement provision 3.2, Design and Development, and 3.3, Software and Firmware.

3.14.4.1 Unacceptable Conditions

The following conditions shall be considered unacceptable for systems and components. Positive action and verified implementation is required to reduce the mishap risk associated with these situations to a level acceptable to cognizant MDA Deputates and Elements.

a. Single component failure, common mode failure, human error, or a design feature that could cause a mishap of Catastrophic or Critical mishap severity categories.

b. Dual independent component failures, dual independent human errors, or a combination of a component failure and a human error involving safety critical command and control functions, which could cause a mishap of Catastrophic or Critical mishap severity categories.

c. Generation of hazardous radiation or energy, when no provisions have been made to protect personnel or sensitive subsystems from damage or adverse effects.

d. Packaging or handling procedures and characteristics that could cause a mishap of severity category 1, 2, or 3 for which no controls have been provided to protect personnel or sensitive equipment.

e. Hazard categories that are specified as unacceptable in the developer agreement.

3.14.4.2 Acceptable Conditions

The following conditions are considered acceptable and will require no further analysis once mitigating actions are implemented and verified.

a. For non-safety critical command and control functions: a system design that requires two or more independent human errors, or that requires two or more independent failures, or a combination of independent failure and human error to lead to a mishap.

b. For safety critical command and control functions: a system design that requires at least three independent failures, or at least three independent human errors, or a combination of least three independent failures and human errors to lead to a mishap.

c. System designs that positively prevent errors in assembly, installation, or connections that could result in a mishap.

d. System designs that positively prevent damage propagation from one component to another or prevent sufficient energy propagation to cause a mishap.

e. System design limitations on operation, interaction, or sequencing that preclude occurrence of a mishap.

f. System designs that provide an approved safety factor, or a fixed design allowance that limits, to an acceptable level, possibilities of structural failure or release of energy sufficient to cause a mishap.

g. System designs that control energy build-up that could potentially cause a mishap (e.g., fuses, relief valves, or electrical explosion proofing).

h. System designs where component failure can be temporarily tolerated because of residual strength or alternate operating paths, so that operations can continue with a reduced but acceptable safety margin.

i. System designs that positively alert the controlling personnel to a hazardous situation where the capability for operator reaction has been provided.

j. System designs that limit or control the use of hazardous materials.

3.14.4.3 Ignition System Safety Requirements

All solid propellant rocket motor ignition systems shall be developed in accordance with MIL-STD-1901A. “Munitions Rocket and Missile Motor Ignition System Design, Safety Criteria For”.

3.14.4.4 Fuze System Safety Requirements

All fuze systems shall be designed and tested in accordance with MIL-STD-1316E(1) “Fuze Design, Safety Criteria For”.

3.14.4.5 Hazardous Materials Transportation

All hazardous materials systems shall be designed, tested, classified, and/or transported in accordance with 49 CFR100-199 “Transportation”; TB-700-2/ NAVSEA Inst 8020.8/TO11A-1-47/DLAR 8220.1, “Explosives Hazard Classification Procedures”; and, as applicable, NAVSEAINST 9310.1B “Naval Lithium Battery Safety Program.”

3.14.4.6 Insensitive Munitions Design and Safety Tests

All munitions systems shall be designed to meet the requirements of and pass the tests in accordance with the following:

a. Public Law (United States Code), Title 10, Chapter 141, Section 2389; Armed Forces Miscellaneous Procurement Provisions: Ensuring Safety Regarding Insensitive Munitions.

b. DoD Directive 5000.1, The Defense Acquisition System, dated May 12, 2003.

c. CJCS Instruction 3170.01C, Joint Capabilities Integration and Development System, dated 24 June 2003.

d. CJCS Manual 3170.01, Operations of Joint Capabilities Integration and Development System, dated 24 June 2003.

e. MIL-STD-2105C, Section 5.2, Hazard Assessment Tests For Non-Nuclear Munitions dated 14 July 2003.

3.14.4.7 Ordnance Systems

All ordnance shall be developed in accordance with DOD 4145.26-M "DOD Contractor's Safety Manual for Ammunition and Explosives," and all electroexplosive devices in accordance with MIL-STD-1576 "Electroexplosive Subsystem Safety Requirements and Test Methods for Space Systems."

3.14.4.8 Missile and Space Vehicle Pressure Systems

All missile and space vehicle pressure systems and their associated ground support equipment shall be developed in accordance with MIL-STD-1522A General Requirements for Safe Design and Operation of Pressurized Missile and Space Systems, Jun 84 or AIAA S080 and S081.

3.14.4.9 Orbital Debris

All programs shall minimize orbital debris per US Government Orbital Debris Mitigation Standard Practices. In addition, programs shall give consideration to NASA Safety Standard 1740.14 Guidelines and Assessment Procedures for Limiting Orbital Debris.

3.14.4.10 Human Engineering

The developer should use MIL-STD-1472F "Human Engineering" as guidance to foster effective procedures, work patterns, and personnel safety and health, and to minimize factors, which degrade human performance or increase error. The developer shall ensure that design induced requirements for operator workload, accuracy, time constraint, mental processing, and communication do not exceed operator capabilities.

3.14.5 Occupational Safety and Health

The developer shall manage Occupational Safety and Health (OSH) and ensure compliance with applicable federal, state, interstate, and local laws and regulations, to mitigate OSH risks, as required by industry and DOD standards. The developer shall address OSH considerations in each phase of a system's life cycle. OSH factors shall be integrated into the systems engineering process (3.2) and risk management program (3.1.5). The developer shall conduct OSH risk assessments associated with:

a. Safety and Health.

b. Hazardous Materials Management.

c. Test safety.

The risk assessment criteria shall be consistent with Risk Acceptance section. The developer shall support Programmatic Environmental, Safety & Health Evaluations (PESHE).

3.14.5.1 Safety and Health

The developer shall identify and evaluate safety and health hazards, define risk levels, and establish a program that manages the probability and severity of all hazards associated with development, use, and disposal of the system. The developer shall utilize the system safety program to manage safety and

health risks encountered in the acquisition process of systems, subsystems, equipment, and facilities. These risks include conditions that create risks of death, injury, acute/chronic illness, disability, and/or reduced job performance of personnel who produce, test, operate, maintain, support, or dispose of the system.

3.14.5.2 Hazardous Materials Management

The developer shall establish and maintain a Hazardous Material Management Program (HMMP) to eliminate or reduce the use of hazardous materials in processes and products and the tracking, storing, handling, packaging, transporting, and disposing of such material. The developer shall evaluate and manage the selection, use, and disposal of hazardous materials consistent with ESOH regulatory requirements and program cost, schedule, and performance goals. A Health Hazard Assessment Report shall be prepared and submitted to the cognizant MDA Deputates and Elements.

3.14.5.3 Test Safety

The purpose of Test Safety is to ensure safety is considered (and safety responsibility assigned) in test and evaluation, to provide existing analysis reports and other safety data, and to respond to all safety requirements necessary for testing in-house, at developer facilities, and at government ranges, centers, or laboratories. These testing requirements supplement those contained in provisions 3.3.2.4 and 3.7.

The developer shall ensure the test and evaluation safety activities recommend actions, and assess actions taken, to reduce, correct, or control CATASTROPHIC and CRITICAL level hazards in the test and evaluation environment. MARGINAL or NEGLIGIBLE level hazards shall also be addressed as required by the cognizant MDA Deputates and Elements and MDA/DT. Specific test and evaluation safety activity tasks shall include the following:

a. *Test and Evaluation Planning*. Planning for test and evaluation safety from the beginning of, and throughout the contract period, shall incorporate the following:

   1) Test program milestones requiring completion of hazard analyses, risk assessments, or other safety studies.

   2) Schedule for analysis, evaluation, and approval of test plans, procedures, and other documents to ensure safety is covered during all testing.

   3) Preparation of, or input to, safety, operating, and test procedures.

   4) Coverage of test equipment, installation of test equipment, and instrumentation in hazard analyses prior to test commencement.

   5) Meeting specialized requirements designated by the cognizant MDA Deputates and Elements and MDA/DT, and informing the cognizant MDA Deputates and Elements, MDA/DT, and MDA/QS of any identified hazards that are unique to the test environment.

   6) Coordination and status reviews with the cognizant test site safety representatives to ensure test safety requirements are identified, monitored, and completed as scheduled.

b. *Safety Assessments*. The developer shall conduct safety assessments and hazard analyses in accordance with MIL-STD-882 to address test and evaluation specific safety concerns.

c. *Safety Reviews.* The developer shall provide assistance to the safety review teams to the extent necessary to support an independent safety review that will validate, from a safety perspective that the system is ready to test.

d. *Follow-up Actions.* The developer shall:

1) Analyze and document safety related test results.

2) Initiate follow-up action to ensure completion of the corrective efforts taken to reduce, correct, or control test and evaluation hazards.

e. *Reports.* The developer shall maintain a repository of test and evaluation hazard/action status reports.

3.14.6 Range Safety

The developer shall meet the Range Safety requirements for each and every range where they intend to test. Developers shall support Range Safety tailoring efforts as directed by the cognizant MDA Deputates and Elements and MDA/QS.

For MDA test operations that occur at a non-National range, a National Range and its associated range safety requirements shall be selected by the cognizant MDA Deputates and Elements and MDA/QS, and used to supplement the requirements of the non-National range.

A lead range shall be identified by cognizant MDA Deputates and Elements and MDA/QS for MDA test operations involving multiple ranges. That lead range will be responsible for assuring overall range safety for the mission. Special attention shall be given to operational handoffs between ranges as required for specific flight tests. Critical considerations include command codes and handover points, and interchange of real-time tracking and telemetry data. Command handovers shall be automated to minimize latency. These handoff processes and procedures shall be positively verified prior to launch of each the test vehicle(s).

The developer shall comply with applicable Range Safety requirements to assure that the general public, launch area personnel, foreign land masses, and launch area resources are provided a level of safety acceptable to Range Safety and that all aspects of prelaunch and launch operations adhere to public laws and national needs.

Developers should use the requirements of EWR 127-1 as guidance for specific range safety and system safety criteria and guidelines.

The developer shall comply with the Lightning Launch Commit Criteria as documented in Aerospace Report No. TR-99 (1413)-1 Natural and Triggered Lightning Launch Commit Criteria, dated 15 Jan 1999 (if modified, the most recent requirements apply), as required based on the natural and triggered lightning environment at the launch site.

3.14.6.1 Flight Termination Systems

3.14.6.1.1 Flight Termination System and Range Safety Tracking System Standards

The developer shall design, test, and deliver flight termination systems, global positioning and inertial measurement range safety tracking systems, transponder tracking systems, and telemetry systems as appropriate, that comply with the requirements of the following as jointly required and tailored by all affected Ranges:

a. RCC-319, Flight Termination Commonality Standard.

b. RCC-254-94, Non-Coherent Transponder Standards

c. RCC-106-01, Telemetry Standards

Any variance to the tailored Range Safety requirements shall require written approval from every affected party, including Range Safety.

In addition to these requirements, the following apply to multiple range and/or multiple vehicle operations:

a. Each launch vehicle requiring flight termination capability shall have flight termination design and procedures, which preclude the possibility of destroying the wrong vehicle during simultaneous flight or while a vehicle or vehicles, are flying and another or others are on the ground. Careful consideration shall be given to selection of command and channel check tones.

b. RCC-324, Global Positioning and Inertial Measurements Range Safety Tracking Systems Commonality Standard. In addition, MDA test operations shall require two independent non-common and adequate range tracking sources.

3.14.6.2 Flight Safety Analysis

The developer shall perform flight safety analysis in accordance with the requirements of the affected Ranges or jointly tailored RCC-321 “Common Risk Criteria for National Test Ranges: Inert Debris” as applicable.

3.14.7 Safety Critical Computing System Functions

The developer shall designate the required safety functions of the computing system as Safety Critical Computing System Functions (SCCSF). The SCCSF are defined as those computer functions in which an error can result in a hazard to the user, friendly forces, material, third parties, or the environment. Safety critical functions of computing systems include not only control functions where the computer exercises direct control over a system but those where the output is used to make safety critical decisions such as monitors of safety critical functions. Typical errors, which result in hazards, are incorrect, inadvertent, or improper functioning; functioning in an improper sequence; or failure to function when required.

The developer shall perform a hazard analysis of the risks associated with the specified functions of the computing system in accordance with MIL-STD-882. Some examples of SCCSF includes:

a. Any function, which controls or directly influences the pre-arming, arming, enabling, release, launch, firing, or detonation of a munitions or directed energy system.

b. Any function that determines, controls, or directly influences the flight path of a munitions or directed energy system.

c. Any function that controls or directly influences the movement of gun mounts, launchers, and other equipment, especially with respect to the pointing and firing safety of that equipment.

d. Any function that controls or directly influences the movement of munitions and/or hazardous materials.

e. Any function that monitors the state of the system for purposes of ensuring its safety.

f. Any function that senses hazards and/or displays information concerning the protection of the system.

g. Any function that controls or regulates energy sources in the system.

Additional functions not on the above list may be considered to be SCCSF depending on the software implementation in the system context.

3.14.8 Software Safety

The software system safety effort shall apply to all computer systems and subsystems that perform safety critical functions during assembly, handling, checkout, test, operation, maintenance, and disposal. In the

context of launch vehicle range safety, these systems and subsystems include auxiliary support equipment (such as cranes and ground transport), vehicle ground support equipment (such as fuel, oxidizer), and airborne systems.

In addition to developed safety critical computer systems and software, these requirements shall apply to all safety critical programmable logic controllers, firmware such as erasable programmable read only memory, Non-Developmental Item (NDI), Commercial-Off-The-Shelf (COTS), and Government-Off-The-Shelf (GOTS) products, and reused code.

Software identified as SCCSF shall be designed, developed, coded, tested, and maintained to meet the requirements of 3.14.8 through 3.14.18 consistent with the type, size, or complexity of the system.

3.14.8.1 Modular Code

Software design and code shall be modular. Modules shall have one entry and one exit point.

3.14.8.2 Number of Modules

Number of program modules containing safety critical functions shall be minimized where possible within the constraints of operational effectiveness, computer resources, and good software design practices.

3.14.8.3 Execution Path

Safety Critical Computing System Functions shall have one and only one possible path leading to their execution.

3.14.8.4 Halt Instructions

Halt, stop, or wait instructions shall not be used in code for safety critical functions. Wait instructions may be used where necessary to synchronize Input/ Output, and when appropriate handshake signals are not available.

3.14.8.5 Single Purpose Files

Files used to store safety critical data shall be unique and shall have a single purpose. Scratch files, those used for temporary storage of data during or between processes, shall not be used for storing or transferring safety critical information, data, or control functions.

3.14.8.6 Unnecessary Features

Operational and support software shall contain only those features and capabilities required by the system. The programs shall not contain undocumented or unnecessary features.

3.14.8.7 Indirect Addressing Methods

Indirect addressing methods shall be used only in well controlled applications. When used, the address shall be verified as being within acceptable limits prior to execution of safety critical operations. Data written to arrays in safety critical applications shall have the address boundary checked by the compiled code.

3.14.8.8 Uninterruptible Code

Sections of code, which have been defined as uninterruptible, shall have defined execution times monitored by an external timer if interrupts are used.

3.14.8.9 Safety Critical Files

Files used to store or transfer safety critical information shall be initialized to a known state before and after use. Data transfers and data stores shall be audited where practical to allow traceability of system functioning.

3.14.8.10 Unused Memory

All processor memory not used for or by the operational program shall be initialized to a pattern that will cause the system to revert to a safe state if executed. It shall not be filled with random numbers, halt, stop, wait or no-operation instructions. Data or code from previous overlays or loads shall not be allowed to remain. (Examples: If the processor architecture halts upon receipt of non-executable code, a watchdog timer shall be provided with an interrupt routine to revert the system to a safe state. If the processor flags non-executable code as an error, an error handling routine shall be developed to revert the system to a safe state and terminate processing.) Information shall be provided to the operator to alert him to the failure or fault observed and to inform him of the resultant safe state to which the system was reverted.

3.14.8.11 Overlays

Overlays of safety critical software shall all occupy the same amount of memory. Where less memory is required for a particular function, the remainder shall be filled with a pattern that will cause the system to revert to a safe state if executed. It shall not be filled with random numbers, halt, stop, no-op, or wait instructions or data or code from previous overlays.

3.14.8.12 Operating System Functions

If an operating system function is provided to accomplish a specific task, operational programs shall use that function and not bypass it or implement it in another fashion.

3.14.8.13 Compilers

The implementation of software compilers shall be validated to ensure that the compiled code is fully compatible with the target computing system and application (may be done once for a target computing system).

3.14.8.14 Flags and Variables

Flags and variable names shall be unique. Flags and variables shall have a single purpose and shall be defined and initialized prior to use.

3.14.8.15 Loop Entry Point

Use of loops shall be restricted to having one and only one entry point. Branches into loops shall not be used. Branches out of loops shall lead to a single exit point placed after the loop within the same module.

3.14.8.16 Software Maintenance Design

Software shall be annotated, designed, and documented for ease of analysis, maintenance, and testing of future changes to the software.

3.14.8.17 Variable Declaration

Software variables or constants used by a safety critical function will be declared/initialized at the lowest possible level.

3.14.8.18 Unused Executable Code

Operational program loads shall not contain unused executable code.

3.14.8.19 Unreferenced Variables

Operational program loads shall not contain unreferenced or unused variables or constants.

3.14.8.20 Assignment Statements

Safety Critical Computing System Functions and other safety critical software items shall not be used in one-to-one assignment statements unless the other variable is also designated as safety critical (e.g., shall not be redefined as another non-safety critical variable).

3.14.8.21 Conditional Statements

Conditional statements shall have all possible conditions satisfied and under full software control (i.e., there shall be no potential unresolved input to the conditional statement). Conditional statements shall be analyzed to ensure that the conditions are reasonable for the task and that all potential conditions are satisfied and not left to a default condition. All condition statements shall be annotated with their purpose and expected outcome for given conditions.

3.14.8.22 Strong Data Typing

Safety critical functions shall exhibit strong data typing. Safety critical functions shall not employ a logic "1" and "0" to denote the safe and armed (potentially hazardous) states. The armed and safe states for munitions shall be represented by at least a unique four-bit pattern. The safe state shall be a pattern that cannot, as a result of a one, two, or three bit error, represent the armed pattern. The armed pattern shall also not be the inverse of the safe pattern. If a pattern other than these two unique codes is detected, the software shall flag the error, revert to a safe state and notify the operator, if appropriate.

3.14.8.23 Timer Values Annotated

Values for timers shall be annotated in the code. Comments shall include a description of the timer function, its value, and the rationale or a reference to the documentation explaining the rationale for the timer value. These values shall be verified and examined for reasonableness for the intended function.

3.14.8.24 Critical Variable Identification

Safety critical variables shall be identified in such a manner that they can be readily distinguished from non-safety critical variables (e.g., all safety critical variables begin with a letter S).

3.14.8.25 Global Variables

Global variables shall not be used for safety critical functions.

3.14.8.26 Software Formal Test Coverage

Software testing shall be controlled by a formal test coverage analysis and procedure. Computer based tools shall be used to ensure coverage is as complete as possible. These requirements supplement those in 3.3. Software testing shall include:

a. GO-NO-GO path testing.

b. Hardware and software input failure mode testing.

c. Boundary, out-of-bounds, and boundary crossing test conditions.

d. Minimum and maximum input data rates in worst case configurations to determine the system's capabilities and responses to these conditions.

e. Input values of zero, zero crossing, and approaching zero from either direction or similar values for trigonometric functions.

f. Complete regression testing for safety critical computing system functions in which changes have been made.

g. Operator interface testing with the introduction of operator errors during safety critical operations to verify safe system response to these errors.

h. Duration stress testing. The stress test time shall be continued for at least the maximum expected operating time for the system. Testing shall be conducted under simulated operational environments. Additional stress duration testing should be conducted to identify potential critical functions (e.g., timing, data senescence, resource exhaustion) that are adversely affected as a result of operational duration. Software testing shall include throughput stress testing (e.g., CPU, data bus, memory, input/output) under peak loading conditions.

3.14.9 Software Maintenance Requirements

The developer shall implement the following software maintenance requirements when performing maintenance on safety critical computing system applications. The developer shall implement the requirements applicable to design and development phase as well as software design and coding phase when conducting maintenance of computing system and firmware.

a. *Safety Critical Function Changes.* Changes to safety critical computing system functions on deployed or fielded systems shall be issued as a complete package for the modified unit or module and shall not be patched.

b. *Safety Critical Firmware Changes.* When not implemented at the depot level or in manufacturers facilities under appropriate quality control, firmware changes shall be issued as a fully functional and tested circuit card. Design of the card and the installation procedures should minimize the potential for damage to the circuits due to mishandling, electrostatic discharge, or normal or abnormal storage environments, and shall be accompanied with the proper installation procedure.

c. *Software Change Medium.* When not implemented at the depot level or in manufacturers facilities under appropriate quality control, software changes shall be issued as a fully functional copy on the appropriate medium. The medium, its packaging and the procedures for loading the program should minimize the potential damage to the medium due to mishandling, electrostatic discharge, potential magnetic fields, or normal or abnormal storage environments, and shall be accompanied with the proper installation procedure.

d. *Modification Configuration Control.* All modifications and updates shall be subject to strict configuration control. The use of automated configuration management tools is encouraged.

e. *Version Identification.* Modified software or firmware shall be clearly identified with the version of the modification, including configuration control information. Both physical (e.g., external label) and electronic (i.e., internal digital identification) "fingerprinting" of the version shall be used.

3.14.10 Design and Development of Computer Systems

The following requirements supplement provisions 3.2 and 3.3 for safety critical applications and computing systems. The developer shall design and develop computer systems to meet the requirements detailed in the following sections.

3.14.10.1 General Design Requirements

Computer systems shall meet the following requirements:

a. Under maximum system loads, central processing unit (CPU) throughput shall not exceed 80 percent of its design value.

b. Computer system architecture shall be single failure/fault tolerant.

c. No single software fault/output shall cause a marginal hazard/mishap.

d. No single software fault/output shall cause a critical hazard/mishap.

e. No double software fault/output shall cause a catastrophic hazard/mishap.

3.14.10.2 Two Person Rule

The developer shall ensure that at least two people shall be thoroughly familiar with the design, code, testing, and operation of each safety critical software module in the system.

3.14.10.3 Program Patch Prohibition

The developer shall prohibit patches throughout the development process. All software changes shall be coded in the source language and compiled prior to entry into operational or test equipment.

3.14.10.4 Design Verification and Validation

The SSWG shall analyze the software throughout the design, development, and maintenance process to verify and validate that the safety design requirements have been correctly and completely implemented. Test results shall be analyzed to identify potential safety anomalies that may occur.

3.14.11 System Design Requirements for Computer Systems

Developers shall implement the following system design requirements in safety critical applications of computing systems that are developed for MDA.

3.14.11.1 Designed Safe States

The system shall have at least one safe state identified for each logistic and operational phase.

3.14.11.2 Standalone Computer

Where practical, safety critical functions should be performed on a standalone computer. If this is not practical, safety critical functions shall be isolated to the maximum extent practical from non-critical functions.

3.14.11.3 Ease of Maintenance

The system and its software shall be designed for ease of maintenance for future personnel not associated with the original design team. Computer system operation and maintenance documentation shall be developed to facilitate maintenance of the software. The developer shall maintain strict configuration control of the software during development and after deployment. The use of techniques for the decomposition of the software system for ease of maintenance is recommended.

3.14.11.4 Safe State Return

Developers shall design software to return hardware subsystems under the control of software to a designed safe state when unsafe conditions are detected. Conditions that can be safely overridden by the battle short shall be identified and analyses performed to verify their safe incorporation.

3.14.11.5 Restoration of Interlocks

Upon completion of tests and/or training wherein safety interlocks are removed, disabled or bypassed, the computer system shall verify restoration of those interlocks prior to resuming normal operation. While overridden, a display shall be made on the operator's or test conductor's console of the status of the interlocks, if applicable.

3.14.11.6 Input/Output Registers

The computer system input/output registers and ports shall not be used for both safety critical and non-critical functions unless the same safety design criteria are applied to the non-critical functions.

3.14.11.7 External Hardware Failures

Software shall be designed to detect failures in external hardware input or output hardware devices and revert to a safe state upon their occurrence. The design shall consider potential failure modes of the hardware involved.

3.14.11.8 Safety Kernel Failure

Developers shall design the system such that a failure of the safety kernel (when implemented) will be detected and the system returned to a designed safe state.

3.14.11.9 Circumvent Unsafe Conditions

System design shall not permit detected unsafe conditions to be circumvented. If a "battleshort" or "safety arc" condition is required in the system, it shall be designed such that it cannot be activated either inadvertently or without authorization.

3.14.11.10 Fallback and Recovery

Computer systems shall be designed to include fallback and recovery to a designed safe state of reduced system functional capability in the event of a failure of system components.

3.14.11.11 Simulators

If simulated items, simulators, and test sets are required, they shall be designed such that the identification of the devices is fail safe and that operational hardware cannot be inadvertently identified as a simulated item, simulator, or test set.

3.14.11.12 System Errors Log

Software shall make provisions for logging all system errors detected. The operator shall have the capability to review logged system errors. Errors in safety critical routines shall be highlighted and shall be brought to the operator's attention as soon as practical after their occurrence.

3.14.11.13 Positive Feedback Mechanisms

Software control of critical functions shall have feedback mechanisms that give positive indications of the function's occurrence.

3.14.11.14 Peak Load Conditions

System and software shall be designed to ensure that design safety requirements are not violated under peak load conditions.

3.14.11.15 Endurance

The developer shall explicitly identify the duration requirements for the system. The design shall be analyzed to determine compliance with duration assumptions. If duration assumptions are violated, an assessment of the impact shall be performed and the results reported.

The developer shall identify potential exposure to:

a. Exhaustion of finite resources over time, and ensure adequate detection and recovery mechanisms are in place to handle these. Examples include file handles, transmission control protocol ports, and counter overflow.

b. Performance degradation over time, and ensure adequate detection and recovery mechanisms are in place to handle these. Example of this is memory and disk fragmentation that can result in increased latency.

c. Cumulative effects over time, and ensure adequate detection and recovery mechanisms are in place to handle these. Examples include cumulative drift in clocks, cumulative jitter in scheduling operations, and cumulative rounding error in floating point and fixed-point operations.

3.14.11.16 Corruption of Computing Environment

Software design shall preclude an application from corrupting the underlying computing environment.

3.14.12 Power-Up System Initialization Requirements

The design for power subsystem, power control, and power-on initialization for safety critical applications of computing systems shall meet the following requirements.

3.14.12.1 Power-up Initialization

The system shall be designed to power-up in a safe state. An initialization test shall be incorporated in the design that verifies that the system is in a safe state and those safety critical circuits and components are tested to ensure their safe operation. The test shall also verify memory integrity and program load.

3.14.12.2 Power Faults

Systems shall be designed to ensure that they are in a safe state:

a. During power-up.

b. During intermittent faults or fluctuations in the power that could adversely affect the system.

c. In the event of power loss.

The system and/or software shall be designed to provide for a safe, orderly shutdown of the system due to either a fault or power-down, such that potentially unsafe states are not created.

3.14.12.3 Primary Computer Failure

The system shall be designed such that a failure of the primary control computer will be detected and the system returned to a safe state.

3.14.12.4 Maintenance Interlocks

Maintenance interlocks, safety interlocks, safing handles, and/or safing pins shall be provided to preclude hazards to personnel maintaining the computing system and its associated equipment. Where interlocks must be overridden to perform tests or maintenance, they shall be designed such that they cannot be inadvertently overridden, or left in the overridden state once the system is restored to operational use. The override of the interlocks shall not be controlled by a computing system.

3.14.12.5 System Level Check

Software shall be designed to perform a system level check at power-up to verify that the system is safe and functioning properly prior to application of power to safety critical functions including hardware controlled by the software. Periodic tests shall be performed by the software to monitor the safe state of the system.

3.14.13 Munitions/Directed Energy Systems Hardware Requirements

Computers, microprocessors, programming languages, and memories for safety critical applications in munitions or directed energy systems shall meet the requirements in the following paragraphs.

3.14.13.1 Central Processing Units Selection

The following guidelines apply to the selection of Central Processing Units (CPUs):

a. CPUs that process entire instructions or data words are preferred to those that multiplex instructions or data (e.g., an 8-bit CPU is preferred to a 4-bit CPU emulating an 8-bit machine).

b. CPUs with separate instruction and data memories and busses are preferred to those using a common data/instruction buss. Alternatively, memory protection hardware, either segment or page protection, separating program memory and data memory is acceptable.

c. CPUs, microprocessors, and computers that can be fully represented mathematically are preferred to those that cannot.

3.14.13.2 Minimum Clock Cycles

Analyses and measurements shall be conducted to determine the minimum number of clock cycles that must occur between functions on the buss to ensure that invalid information is not picked up by the CPU when CPUs do not comply with the above guidelines or those used at the limits of their design criteria (e.g., at or above maximum clock frequency). Analyses shall also be performed to ensure that interfacing devices are capable of providing valid data within the required time frame for CPU access.

3.14.13.3 Read-Only-Memories

Where Read-Only-Memories are used, positive measures shall be taken to ensure that the data cannot be corrupted or destroyed.

3.14.14 Self-Checking Design Requirements

For safety critical applications and computing systems the following self-checking requirements shall apply.

3.14.14.1 Watchdog Timers

Watchdog timers or similar devices shall be provided to ensure that the microprocessor or computer is operating properly. The timer reset shall be designed such that the software cannot enter an inner loop and reset the timer as part of that loop sequence. The design of the timer shall ensure that failure of the

primary CPU clock cannot compromise its function. The timer reset shall be designed such that the system is returned to a known safe state and the operator alerted (as applicable).

3.14.14.2 Memory Checks

The design shall include periodic checks of memory, instruction, and each data buss. The design of the test sequence shall ensure that single point or likely multiple failures are detected. Checksum of data transfers and Program Load Verification checks shall be performed at load time and periodically thereafter to ensure the integrity of safety critical code.

3.14.14.3 Fault Detection

Fault detection and isolation programs shall be written for safety critical subsystems of the computing system. The fault detection program shall be designed to detect potential safety critical failures prior to execution of the related safety critical function. Fault isolation programs shall be designed to isolate the fault to the lowest level practical and provide this information to the operator or maintainer.

3.14.14.4 Operational Checks

Operational checks of testable safety critical system elements shall be made immediately prior to performance of a related safety critical operation.

3.14.15 Safety Critical Computing System Functions and Data

Safety critical computing system functions and data shall meet the following requirements.

3.14.15.1 Safety Degradation

The system shall be designed such that systems and software shall prevent degradation of safety by other interfacing automata and software.

3.14.15.2 Unauthorized Interaction

Software shall be designed to prevent unauthorized system or subsystem interaction from initiating or sustaining a safety critical function sequence.

3.14.15.3 Unauthorized Access

System design shall prevent unauthorized or inadvertent access to or modification of the software (source or assembly) and object code. This includes preventing self-modification of the code.

3.14.15.4 Safety Kernel

Safety kernels shall be:

a. Resident in non-volatile read only memory (ROM) or in protected memory that cannot be overwritten by the computing system.

b. Designed and implemented in such a manner that it cannot be corrupted, misdirected, delayed, or inhibited by any other program in the system.

3.14.15.5 Inadvertent Jumps

The system shall detect inadvertent jumps within or into Safety Critical Computing System Functions, return the system to a safe state, and, if practical, perform diagnostics and fault isolation to determine the cause of the inadvertent jump. Any safety critical functions overwritten by the loaded data/program shall trigger a warning message to the operator.

3.14.15.6 Load Data Integrity

The executive program or operating system shall ensure the integrity of data or programs loaded into memory prior to their execution.

3.14.15.7 Operational Reconfiguration Integrity

The executive program or operating system shall ensure the integrity of the data and programs during operational reconfiguration.

3.14.16 Interface Design Requirements

Safety critical input/output interfaces designs shall comply with the requirements of this section.

3.14.16.1 Feedback Loops

Feedback loops from the system hardware shall be designed such that the software cannot cause a runaway condition due to the failure of a feedback sensor. Known component failure modes shall be considered in the design of the software and checks designed into the software to detect failures.

3.14.16.2 Interface Control

Safety critical computing system functions and their interfaces to safety critical hardware shall be controlled at all times, that is, the interface shall be monitored to ensure that erroneous or spurious data does not adversely affect the system, that interface failures are detected, and that the state of the interface is safe during power-up, power fluctuations and interruptions, or in the event of system errors or hardware failures.

3.14.16.3 Decision Statements

Decision statements in safety critical computing system functions shall not rely on inputs of all ones or all zeros, particularly when this information is obtained from external sensors.

3.14.16.4 Inter-CPU Communications

Inter-CPU communications shall successfully pass verification checks in both CPUs prior to the transfer of safety critical data. Periodic checks shall be performed to ensure the integrity of the interface. Detected errors shall be logged. If the interface fails several consecutive transfers, the operator shall be alerted and the transfer of safety critical data terminated until diagnostic checks can be performed.

3.14.16.5 Data Transfer Messages

Data transfer messages shall be of a predetermined format and content. Each transfer shall contain a word or character string indicating the message length (if variable), the type of data and content of the message. As a minimum, parity checks and checksums shall be used for verification of correct data transfer. Cyclic Redundancy Checks shall be used where practical. No information from data transfer messages shall be used prior to verification of correct data transfer.

3.14.16.6 External Functions

External functions requiring two or more safety critical signals from the software (e.g., arming of an Ignition Safety Device or Arm Fire Device and release of an air launched weapon) shall not receive all of the necessary signals from a single input/output register or buffer.

3.14.16.7 Input Reasonableness Checks

Limit and reasonableness checks, including time limits, dependencies, and reasonableness checks, shall be performed on all analog and digital inputs and outputs prior to safety critical functions execution based

on those values. No safety critical functions shall be executable based on safety critical analog or digital inputs that cannot be verified.

3.14.16.8 Full Scale Representations

Software shall be designed such that full scale and zero representations of the software are fully compatible with the scales of any digital-to-analog, analog-to-digital, digital-to-synchro, and/or synchro-to-digital converters.

3.14.17 User Interface to Safety Critical Computing Systems

User interface to safety critical computing systems shall meet the design requirements of this section.

3.14.17.1 Processing Cancellation

The software shall be designed such that the operator may cancel current processing with a single action and have the system revert to a designed safe state. The system shall be designed such that the operator may exit potentially unsafe states with a single action. This action shall revert the system to a known safe state (e.g., the operator shall be able to terminate missile launch processing with a single action, which shall safe the missile). The action may consist of pressing two keys, buttons, or switches at the same time. Where operator reaction time is not sufficient to prevent a mishap, the software shall revert the system to a known safe state, report the failure, and report the system status to the operator.

3.14.17.2 Hazardous Function Initiation

Two or more unique operator actions shall be required to initiate any potentially hazardous function or sequence of functions. The actions required shall be designed to minimize the potential for inadvertent actuation, and shall be checked for proper sequence.

3.14.17.3 Safety Critical Displays

Safety critical operator displays, legends, and other interface functions shall be clear, concise, and unambiguous and, where possible, use redundant display devices.

3.14.17.4 Operator Entry Errors

Software shall be capable of detecting improper operator entries, sequences of entries, or operations and prevent execution of safety critical functions as a result. It shall alert the operator to the erroneous entry or operation. Alerts shall indicate the error and corrective action. The software shall also provide positive confirmation of valid data entry or actions taken (i.e., the system shall provide visual and/or aural feedback to the operator such that the operator knows that the system has accepted the action and is processing it). The system shall also provide a real-time indication that it is functioning. Processing functions requiring several seconds or longer shall provide a status indicator to the operator during processing.

3.14.17.5 Safety Critical Alerts

Alerts shall be designed such that routine alerts are readily distinguished from safety critical alerts. The operator shall not be able to clear a safety critical alert without taking corrective action or performing subsequent actions required to complete the ongoing operation.

3.14.17.6 Unsafe Situation Alerts

Signals alerting the operator to unsafe situations shall be directed as straightforward as practical to the operator interface.

3.14.17.7 Unsafe State Alerts

If an operator interface is provided and a potentially unsafe state has been detected, the system shall alert the operator to the anomaly detected, the action taken, and the resulting system configuration and status.

3.14.18 Critical Timing and Interrupt Functions

The developer shall implement the following design requirements on safety critical timing functions and interrupts.

a. *Safety Critical Timing*. Safety critical timing functions shall be controlled by the computer and not rely on human input. Safety critical timing values shall not be modifiable by the operator from system consoles, unless specifically required by the system design. In these instances, the computer shall determine the reasonableness of timing values.

b. *Valid Interrupts*. The software shall be capable of discriminating between valid and invalid (e.g. spurious) external and/or internal interrupts. Invalid interrupts shall not be capable of creating hazardous conditions. Valid external and internal interrupts shall be defined in system specifications. Internal software interrupts are not a preferred design as they reduce the analyzability of the system.

c. *Recursive Loops*. Recursive and iterative loops shall have a maximum documented execution time. Reasonableness checks will be performed to prevent loops from exceeding the maximum execution time.

d. *Time Dependency*. The results of a program should not be dependent on the time taken to execute the program or the time at which execution is initiated. Safety critical routines in real-time programs shall ensure that the data used is still valid (e.g., by using senescence checks).

3.14.19 MDA Safety Integration

The following integration requirements shall be applicable to developers and suppliers.

3.14.19.1 Developer and Supplier Integrator Responsibility

Developers and suppliers designated as integrator for the safety functions shall:

a. Establish and maintain an Integrated System Safety Program Plan (ISSPP) defining the role of the integrator and the effort required from each supplier to help integrate system safety requirements for the total system. The ISSPP shall be submitted to the cognizant MDA Deputates and Elements for approval and MDA/QS for information. In addition to the other contractually imposed requirements the plan shall address and identify:

    1) Definition of where the control, authority, and responsibility transitions from developer to suppliers.

    2) Analyses, risk assessment, and verification data to be developed by each supplier with format and method to be utilized.

    3) Data each supplier is required to submit to the integrator and its scheduled delivery keyed to program milestones.

    4) Schedule and other information considered pertinent by the integrator.

    5) The method of development of system level (including software) requirements to be allocated to each of the supplier's as a part of the system specification, end-item specifications, and other interface requirement documentation.

6) Safety related data pertaining to non-developmental items (NDI) and Commercial- Off- The- Shelf (COTS) hardware and software.

7) Integrated safety analyses to be conducted and support required from suppliers.

8) Developers' roles in test range, nuclear safety, explosive, or other certification processes.

b. Initiate action through the cognizant MDA Deputates and Elements to ensure each supplier is required to be responsive to the ISSPP. Recommend contractual modification where the need exists.

c. When conducting safety risk assessments, analyze the integrated system design, operations, and specifically the interfaces between the products of each supplier and the end item. Data or analyses provided by suppliers shall be used in the conduct of this effort.

d. When performing a safety assessment, summarize the mishap risk presented by the operation of the integrated system. Data or analyses provided by suppliers shall be used in the conduct of this effort.

e. Provide assistance and guidance to suppliers regarding safety matters.

f. Resolve differences between suppliers in areas related to safety, especially during development of safety inputs to system and item specifications. Where the integrator cannot resolve problems, notify the cognizant MDA Deputates and Elements for resolution and action.

g. Initiate action through the cognizant MDA Deputates and Elements to ensure information required by a supplier (from the developer or other suppliers) to accomplish safety tasks, is provided in an agreed-to format.

h. Develop a method of exchanging safety information between developers. If necessary, schedule and conduct technical meetings between all suppliers to discuss, review, and integrate the safety effort. Use of the SSWG meetings should be included as required.

i. Implement an audit program to ensure the objectives and requirements of the system safety program are accomplished. Whenever the developer finds that a supplier has failed to meet contract requirements, the developer shall notify the cognizant MDA Deputates and Elements and MDA/QS in writing. The integrator for the safety effort will send a copy of the notification to the supplier.

j. Provide support to other MDA or external SSWG's.

3.14.19.2 Non-Integration Developers and Suppliers

Non-Integration developers and suppliers shall provide safety data and support (including participation in SSWG's) needed by other suppliers and the integrator to the extent specified in the contract.

3.14.20 Flowdown of Requirements from Developer to Supplier

The following requirements supplement the supplier management requirements contained in 3.13. Developers shall include the following requirements in all contracts/agreements with suppliers. The "chain of responsibility" for formally flowing down the system safety contractual requirements from the integrating developer to different levels of suppliers and vendors (who provide different applicable subsystems, equipment and/or parts) shall be identified.

a. All suppliers shall be required to maintain suitable documentation of safety analyses they have performed in formats, which will permit incorporation of their data into the overall analysis program.

b. Suppliers shall be required to develop system safety program plans to be included as annexes to the developer's SSPP.

c. Suppliers shall be required to provide information on software, component, and subassembly characteristics, including failure modes, failure rates, and possible hazards, which will permit the integrating developer to evaluate the items for their impact on system safety.

d. All suppliers shall participate in the SSWG, when required.

3.14.21 Safety Metrics

The developer shall collect safety metrics as described below and report them to the cognizant MDA Deputates and Elements and MDA/QS.

The developer shall populate the table below with numbers representing the top level mishaps/hazards associated with each severity and probability level. Include one chart for initial residual safety risk assessments and another chart for projected final safety risk assessments once proposed mitigations are verified. Include a listing of all top-level hazards/mishaps, their causal factors, and their current status (open-pending analysis, open-pending risk acceptance, closed-risk accepted).

HAZARD/MISHAP CATEGORY FREQUENCY	(1) CATASTROPHIC	(2) CRITICAL	(3) MARGINAL	(4) NEGLIGIBLE
(A) FREQUENT	<#>	<#>	<#>	<#>
(B) PROBABLE	<#>	<#>	<#>	<#>
(C) OCCASIONAL	<#>	<#>	<#>	<#>
(D) REMOTE	<#>	<#>	<#>	<#>
(E) IMPROBABLE	<#>	<#>	<#>	<#>

This Page Intentionally Left Blank

4.0 NOTES

4.1 Custodian

Requests for copies of this document should be submitted to:

Director
Quality, Safety, and Mission Assurance (QS)
Missile Defense Agency
7100 Defense Pentagon
Washington, DC 20301-7100

This Page Intentionally Left Blank

APPENDIX (A)

MAIP Development and Approval Process

APPENDIX (A)

This Page Intentionally Left Blank

APPENDIX (A)

MAIP Development and Approval Process

Step 1 – Start with MAP.

MDA Assurance Provisions, MDA-QS-001-MAP, 9 January 2004, or current revision.

Step 2 - Baseline Current Practices. Each Deputate, with QS support, shall perform a detailed mission assurance baseline analysis to catalogue current assurance processes and practices against MAP requirements. The analysis shall address all acquisition phases of the Deputate's programs, processes, products, and services. The baseline analysis shall include the prioritization of the assurance provisions, the ones the Deputates think are critical, and define where their own assurance needs are not met in their current baseline. These analyses should be addressed to the lowest appropriate level or tier of each Deputate.

The results of the analysis shall be catalogued and documented in a Mission Assurance Implementation Plan (MAIP). MDA/QS will provide tools and personnel to assist. The MAIP shall identify requirements performed and not performed and include rationale for exceptions or alternate approaches.

Step 3 - Generate MAIP. Out of the Step 2 base-lining process, each Deputate shall develop a MAIP, which describes how the MAP is executed. The MAP shall be used as the boilerplate in developing the MAIP. While a single MAIP for each Deputate is recommended, Deputates may develop separate MAIPs within the Deputate (e.g., by System or contractor) at their discretion. Separate MAIPs shall be sufficiently linked to the top level Deputate MAIP to allow coherent traceability. In the introductory portion of the MAIP, the developer should describe how the MAP provisions are/will be invoked (e.g., contract, work authorization, internal policy). The MAIP shall:

a. Refer to each MAP Section sequentially, describing what implementation methodology is being used to accomplish MAP provisions along with rationale for exceptions or alternate approaches. Where existing, equivalent, documented procedures are in place, a simple reference to the procedure is sufficient.

Note: In some cases, an overarching procedure (e.g., Configuration Management) may cover an entire MAP section (3.10) thus eliminating the need to spell out details at the sub-paragraph levels (3.10.1, 3.10.1.1, etc). Minor exceptions or clarifications, whether at the top level, or at sub-paragraph levels, should be addressed.

b. Identify the executing organization for each applicable MAP paragraph (e.g., organization, government, contractor).

c. Where applicable, provide compelling reason to phase the implementation (incrementally by block or contract), describe when in the acquisition process the MAP provisions are/will be implemented and the rationale for such phasing.

Step 4 – MDA/QS MAIP Review. MDA/QS shall work with the Deputate and review the MAIP, in parallel with its development. Any unresolved issues during this review process shall be submitted to Director for resolution.

Step 5 - MDA PCB Process. Each Deputate's MAIP shall be processed in accordance with the MDA PCB process. The MDA Director is the approving authority. Disapproval requires MAIP revision and reprocessing through the MAIP development, review, and MDA PCB processes.

APPENDIX (A)

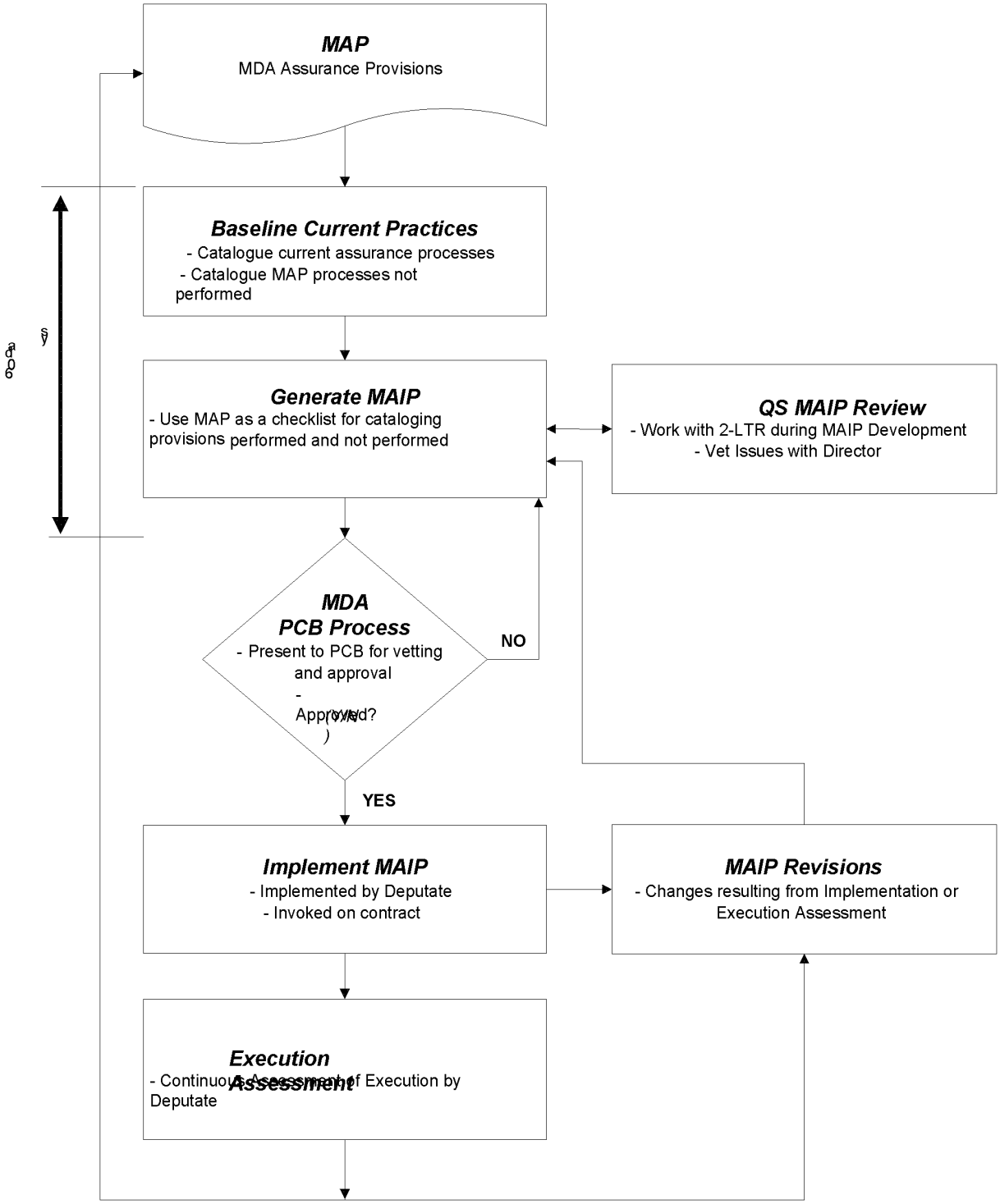
MAIP Development and Approval Process

Step 6 - Implement MAIP. Deputate directors are accountable for implementation of the approved MAIP, with review and support by QS and other related two-letters. Implementation may require contract revisions when authorized by the MDA Director.

Step 7 - MAIP Revisions. Changes resulting from implementation or execution assessment of the MAIP shall be reprocessed through the MAIP development, review, and MDA PCB processes.

Step 8 - Execution Assessment. Each Deputate shall continuously assess MAIP execution. Execution Assessment results in a continuous improvement process with results providing feedback to improve the MAIP/MAP or their implementation and execution.

MAIP Development and Approval Process Flow



This Page Intentionally Left Blank