

# AFRL-RH-WP-SR-2014-0001

### AN UPDATED VERSION OF THE U.S. AIR FORCE MULTI-ATTRIBUTE TASK BATTERY (AF-MATB)

William D. Miller, Jr. Kevin D. Schmidt

ASEE SMART

Justin R. Estepp Air Force Research Laboratory

> Margaret Bowers Iris Davis

**Ball Aerospace & Technologies Corp.** 

#### AUGUST 2014

#### SPECIAL REPORT

Distribution A: Approved for public release; distribution unlimited.

#### **STINFO COPY**

AIR FORCE RESEARCH LABORATORY 711TH HUMAN PERFORMANCE WING, HUMAN EFFECTIVENESS DIRECTORATE, WRIGHT-PATTERSON AIR FORCE BASE, OH 45433 AIR FORCE MATERIEL COMMAND UNITED STATES AIR FORCE

#### NOTICE AND SIGNATURE PAGE

Using Government drawings, specifications, or other data included in this document for any purpose other than Government procurement does not in any way obligate the U.S. Government. The fact that the Government formulated or supplied the drawings, specifications, or other data does not license the holder or any other person or corporation; or convey any rights or permission to manufacture, use, or sell any patented invention that may relate to them.

This report was cleared for public release by the 88<sup>th</sup> Air Base Wing Public Affairs Office and is available to the general public, including foreign nationals. Copies may be obtained from the Defense Technical Information Center (DTIC) (<u>http://www.dtic.mil</u>).

AFRL-RH-WP-SR-2014-0001 HAS BEEN REVIEWED AND IS APPROVED FOR PUBLICATION IN ACCORDANCE WITH ASSIGNED DISTRIBUTION STATEMENT.

//signed// KYLE L. TRAVER Work Unit Manager Applied Neuroscience Branch //signed// SCOTT M. GALSTER Chief, Applied Neuroscience Branch Warfighter Interface Division

//signed// WILLIAM E. RUSSELL Chief, Warfighter Interface Division Human Effectiveness Directorate 711 HPW Performance Wing

This report is published in the interest of scientific and technical information exchange, and its publication does not constitute the Government's approval or disapproval of its ideas or findings.

REPORT DOCUMENTATION PAGE			Form Approved OMB No. 0704-0188				
The public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Department of Defense, Washington Headquarters Services, Directorate for Information Operations and Reports (0704-0188), 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to any penalty for failing to comply with a collection of information it des not display a currently valid OMB control number. <b>PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS</b> .							
<b>1. REPORT DATE (</b> 29-08-14	DD-MM-YY)	2. REPO Spe	DRT TYPE ecial		<b>3. DATES CO</b> 01 MAY 2	DATES COVERED (From - To) 1 MAY 2013 to 12 FEBRUARY 2014	
4. TITLE AND SUBTITLE An Updated Version Of The U.S. Air Force Multi-Attribute Task Battery (AF-MATB)			5a. CONTRACT NUMBER				
				5b. GRANT NUMBER N/A			
						5c. PROGRAM ELEMENT NUMBER	
<b>6. AUTHOR(S)</b> William D. M	liller, Jr.**					5d. PROJECT NUMBER	
Kevin D. Sch	midt**					5e. TASK NUMBER	
Justin R. Estepp* Margaret Bowers** Iris Davis***				<b>5f. WORK UNIT NUMBER</b> 9999			
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)Science, Mathematics & Research for Transformation**, Ball Aerospace & Technologies Corp. ***American Society for Engineering Education, 1818 N Street NW, Suite 600Systems Engineering Solutions 2875 Presidential Dr., Suite 180 Fairborn, OH 45324-6269			8. PERFORMING ORGANIZATION REPORT NUMBER N/A				
9. SPONSORING/M Air Force Materi Air Force Resear	ONITORING AGEN iel Command* rch Laboratory	ICY NAME(S) AND A	ADDRESS(ES)			10. SPONSORING/MONITORING AGENCY ACRONYM(S) 711 HPW/RHCPA	
711 <sup>th</sup> Human Performance Wing Human Effectiveness Directorate Warfighter Interface Division Applied Neuroscience Branch			11. SPONSORING/MONITORING AGENCY REPORT NUMBER(S) AFRL-RH-WP-SR-2014-0001				
12. DISTRIBUTION/AVAILABILITY STATEMENT Distribution A: Approved for public release; distribution unlimited.							
13. SUPPLEMENTARY NOTES 88ABW Cleared 12/03/2014; 88ABW-2014-5667. Report contains color.							
<b>14. ABSTRACT</b> The U.S. Air Force-developed adaptation of the Multi-Attribute Task Battery, known as AF-MATB, was released in 2010 and has been widely used by both military and civilian researchers interested in assessing human performance in a controlled multitask environment. The most recent release of AF-MATB contains numerous improvements and additions, including but not limited to serial and digital port event triggering for real-time task state synchronization and new task modes including those based on seminal published works (e.g. Molloy & Parasuraman, 1996). This report serves as the manual for the updated software package and provides researchers with extensive documentation on how to configure and use, as well as analyze data generated by, the software.							
Workload, Strategic Behavior, MATB, Multitasking, Task Battery, Simulator, Multi-Attribute Task Battery, Automation							
16. SECURITY CLA a. REPORT Unclassified	SSIFICATION OF: b. ABSTRACT Unclassified	c. THIS PAGE Unclassified	17. LIMITATION OF ABSTRACT: SAR	18. NUMBER OI PAGES 237	- 19a. NAME 19b. TELE	E OF RESPONSIBLE PERSON Kyle Traver PHONE NUMBER (Include Area Code)	

Standard Form 298 (Rev. 8-98) Prescribed by ANSI Std. Z39-18 THIS PAGE INTENTIONALLY LEFT BLANK

Acknowledgements	xxiii
1.0 INTRODUCTION	1
2.0 SYSTEM REQUIREMENTS	2
3.0 AF-MATB SUBTASK DESCRIPTIONS	3
3.1. System Monitoring	3
3.1.1. Manual Mode	4
3.1.1.1. Gauges	4
3.1.1.2. Lights	6
3.1.2. Automated Mode	7
3.2. Resource Management	10
3.2.1. Manual Mode	11
3.2.2. Automated Mode	13
3.2.2.1. Automation Algorithm 1	13
3.2.2.2. Automation Algorithm 2	14
3.3. Tracking	18
3.4. Communications	20
3.5. Scheduling	22
4.0 ADDITIONAL AF-MATB FEATURES	27
4.1. Information Throughput Mode	27
4.2. Subtask Component Visibility	29
4.3. NASA-TLX Integration	30
4.4. Subtask Automation Indicators	33
4.5. Serial and Digital Port-Triggering	34
5.0 BEFORE USING AF-MATB	37
6.0 TASK COMMANDS	40
6.1. System Monitoring Commands	40
6.2. Resource Management Response Commands	42
6.3. Communications Response Commands	44
6.3.1. Frequency Entry Method – Arrow Keys	44
6.3.2. Frequency Entry Method – Number Pad	45
6.4. System Commands	46
6.5. Additional Keyboard Commands	57
6.5.1. Transition Logging	58
6.5.2. System Management	58
6.5.2.1. Manual Mode Commands	58
6.5.2.1.1. Malfunction Commands	58
6.5.2.1.2. Timeout Commands	58
6.5.2.2. Automated Mode Commands	59
6.5.2.2.1. Malfunction Commands	59
6.5.2.2.2. Timeout Commands	60
6.5.2.3. Other Commands	61
6.5.3. Resource Management	61
6.5.3.1. Manual Mode and Automation Algorithm 1 Commands	61
6.5.3.1.1. Malfunction Commands	61
6.5.3.1.2. Timeout Commands	62
6.5.3.2. Automation Algorithm 2 Commands	63

6.5.3.3. Other Commands	64
6.5.4. Tracking	64
6.5.5. Communications	65
6.5.6. Task Component Visibility Commands	65
7.0 RUNNING AF-MATB	66
7.1. Initial Execution	66
7.2. Loading Behaviors	66
7.2.1. Phase 1: Loading Config and Script Files	66
7.2.1.1. Loading Files From The AF-MATB	66
7.2.1.2. Loading Files Via The Configuration and Script Generator Utilities	70
7.2.2. Phase 2: Loading AF-MATB System Files	71
7.2.3. Phase 3: Establishing Port-Triggering Connections	72
7.3. Saving Behaviors	74
7.3.1. Performance Directory Types	74
7.3.1.1. Free Mode	75
7.3.1.2. Single Condition, Standard Trial	75
7.3.1.3. Multiple Condition, Standard Trial	77
7.3.1.4. Single Condition, System Automation Trial	78
7.3.1.5. Multiple Condition, System Automation Trial	79
7.3.1.6. Single Condition, Resource Automation Trial	80
7.3.1.7. Multiple Condition, Resource Automation Trial	81
7.3.1.8. Single Condition, Dual Automation Trial	82
7.3.1.9. Multiple Condition, Dual Automation Trial	83
7.3.1.10. Single Condition, IT Mode Trial	84
7.3.1.11. Multiple Condition, IT Mode Trial	85
7.3.2. Log Details and Descriptions	86
7.3.2.1. Correct Comm Log	86
7.3.2.2. Correct Gauge Log	87
7.3.2.3. Correct Light Log	87
7.3.2.4. Incorrect Comm Log	88
7.3.2.5. Incorrect Gauge Log	88
7.3.2.6. Incorrect Light Log	88
7.3.2.7. Master Event Log	88
7.3.2.8. Port Log	89
7.3.2.9. Resource Automation Log	89
7.3.2.10. Resource Management Log	89
7.3.2.11. System Automation Log	91
7.3.2.12. Tracking Log	91
7.3.2.13. Transition Log	93
7.3.3. Performance Summary Details and Descriptions	93
7.3.3.1. Standard Performance Summary	94
7.3.3.1.1. System Monitoring Section	94
7.3.3.1.2. Tracking Section	94
7.3.3.1.3. Resource Management Section	97
7.3.3.1.4. Communications Section	100
7.3.3.2. System Automation Performance Summary	101

7.3.3.3.	Resource Automation Performance Summary	101
7.3.3.4.	Information Throughput Summary	102
7.3.4. I	DataStructure Details and Descriptions	103
8.0 AF-MA	TB CONFIGURATION UTILITY	103
8.1. Utility	/ Introduction	103
8.2. Param	eter Groups	103
8.2.1. N	Vavigating The Group Dropdown	105
8.2.2. A	AF-MATB System Parameters Group	108
8.2.2.1.	Input Options	109
8.2.2.2.	Enable Pausing?	110
8.2.2.3.	Enable Manual Event Triggering?	110
8.2.2.4.	Enable Manual Pump Repair?	110
8.2.2.5.	Display Date and Time?	111
8.2.2.6.	Enable Scheduling?	111
8.2.2.7.	Enable Escape Key?	111
8.2.2.8.	Enable Home Kev?	112
8.2.2.9.	Timer Cycling Rate (Hz)	112
8.2.2.10	Initial Ramp-up Time Delay (Seconds)	112
8.2.2.11	NASA-TLX Ramp-Up Time Delay (Seconds)	113
8.2.2.12	Resource Management Automation Algorithm	113
8.2.2.13	Enable Subtask Automation Indicators?	114
8.2.2.14	Enable Information Throughput Mode?	114
8.2.3. S	system Monitoring Subtask Basic Parameters Group	115
8.2.3.1.	Gauge Speed Lower Limit (Pixels/Cycle)	115
8.2.3.2.	Gauge Speed Upper Limit (Pixels/Cycle)	116
8.2.3.3.	End of Range Delay Max (Cycles)	116
8.2.3.4.	Correct Fault Identification Pause (Cycles)	116
8.2.3.5.	Gauge Malfunction Timeout (Seconds)	117
8.2.3.6.	Indicator Light Malfunction Timeout (Seconds)	117
8.2.4. 7	Tracking Subtask Basic Parameters Group	118
8.2.4.1.	Tracking Gain	118
8.2.4.2.	Tracking RMS Interval (Seconds)	119
8.2.4.3.	Center Range (Pixels)	119
8.2.4.4.	Vertical Spacing for Crosshairs (Pixels)	119
8.2.4.5.	Horizontal Spacing for Crosshairs (Pixels)	120
8.2.4.6.	Horizontal Crosshair Count	120
8.2.4.7.	Tracking Range Outline Shape	120
8.2.4.8.	Circle Tracking Outline Radius (Pixels)	121
8.2.4.9.	Square Tracking Outline Width (Pixels)	121
8.2.5. C	Communications Subtask Parameters Group	122
8.2.5.1.	Callsign Label	123
8.2.5.2.	Comm Slot 1 Label	123
8.2.5.3.	Comm Slot 2 Label	123
8.2.5.4.	Comm Slot 3 Label	124
8.2.5.5.	Comm Slot 4 Label	124
8.2.5.6.	Enter Confirmation Highlight (Cycles)	124
	· · ·	

8.2.5.7. Ta	arget Communication Timeout (Seconds)	125
8.2.5.8. C	ommunication Event Duration (Seconds)	125
8.2.5.9. Fr	requency Entry Method	125
8.2.5.10. \$	Slot 1 & 2 Lower Limit	126
8.2.5.11. \$	lot 1 & 2 Upper Limit	126
8.2.5.12. \$	Slot 1 & 2 Frequency Increments	127
8.2.5.13. \$	blot 3 & 4 Lower Limit	127
8.2.5.14. \$	lot 3 & 4 Upper Limit	128
8.2.5.15. \$	Slot 3 & 4 Frequency Increments	129
8.2.6. Reso	burce Management Subtask Basic Parameters Group	130
8.2.6.1. Pu	Imp 1 & Pump 3 Flow Rate (Liters/Minute)	131
8.2.6.2. Pu	mp 2 & Pump 4 Flow Rate (Liters/Minute)	131
8.2.6.3. Pu	Imp 5 & Pump 6 Flow Rate (Liters/Minute)	131
8.2.6.4. Pu	mp 7 & Pump 8 Flow Rate (Liters/Minute)	132
8.2.6.5. Ta	ank A & Tank B Drop Rate (Liters/Minute)	132
8.2.6.6. Ta	ank A & Tank B Maximum (Liters)	132
8.2.6.7. Ta	ank C & Tank D Maximum (Liters)	133
8.2.6.8. Ta	ank A Starting Volume (Liters)	133
8.2.6.9. Ta	ank B Starting Volume (Liters)	133
8.2.6.10.	Cank C Starting Volume (Liters)	134
8.2.6.11.	Cank D Starting Volume (Liters)	134
8.2.6.12. F	Pump Failure Duration (Seconds)	134
8.2.7. Syst	em Monitoring Subtask Automation Parameters Group	135
8.2.7.1. A	utomation Fault Restore Time (Seconds)	135
8.2.7.2. A	utomation Failure Duration (Seconds)	135
8.2.8. Trac	king Subtask Difficulty Parameters Group	136
8.2.8.1. Lo	ow Difficulty Reticle Speed Lower Limit (Pixels/Cycle)	137
8.2.8.2. Lo	ow Difficulty Reticle Speed Upper Limit (Pixels/Cycle)	137
8.2.8.3. Lo	ow Difficulty Direction Change Lower Limit (Cycles)	138
8.2.8.4. Lo	ow Difficulty Direction Change Upper Limit (Cycles)	138
8.2.8.5. M	oderate Difficulty Reticle Speed Lower Limit (Pixels/Cycle)	138
8.2.8.6. M	oderate Difficulty Reticle Speed Upper Limit (Pixels/Cycle)	139
8.2.8.7. M	oderate Difficulty Direction Change Lower Limit (Cycles)	139
8.2.8.8. M	oderate Difficulty Direction Change Upper Limit (Cycles)	139
8.2.8.9. H	igh Difficulty Reticle Speed Lower Limit (Pixels/Cycle)	140
8.2.8.10. H	High Difficulty Reticle Speed Upper Limit (Pixels/Cycle)	140
8.2.8.11. H	High Difficulty Direction Change Lower Limit (Cycles)	140
8.2.8.12. H	High Difficulty Direction Change Upper Limit (Cycles)	141
8.2.9. Reso	ource Management Subtask Automation Parameters	142
8.2.9.1. Ta	ank Volume Update Interval (Seconds)	142
8.2.9.2. M	ain Tank Automation Lower Limit (Liters)	143
8.2.9.3. M	ain Tank Automation Upper Limit (Liters)	143
8.2.9.4. Su	apply Tank Automation Lower Limit (Liters)	144
8.2.9.5. Su	upply Tank Automation Upper Limit (Liters)	144
8.2.9.6. A	utomation Maximum Tank Difference (Liters)	145
8.2.9.7. R	esource Management RMS Interval (Seconds)	145

8.2.9.8. RMS Target Value (Liters)	146
8.2.9.9. Hide Automation Behavior	146
8.2.9.10. Automation Fault Duration (Seconds)	147
8.2.10. Subtask Visibility Parameters	148
8.2.10.1. Initially hide System Monitoring subtask?	148
8.2.10.2. Initially hide Tracking subtask?	148
8.2.10.3. Initially hide Communications subtask?	149
8.2.10.4. Initially hide Resource Management subtask?	149
8.2.10.5. Initially hide Scheduling window?	149
8.2.10.6. Initially hide Pump Status window?	150
8.2.11. Port Triggering Parameters	151
8.2.11.1. Event List	152
8.2.11.2. Enable Port Triggering?	153
8.2.11.3. Triggering Mechanism	153
8.2.11.4. Change Event Code	153
8.2.11.5. Serial Port Address	155
8.2.11.6. Digital Port Address	156
8.2.11.7. Digital Device Nickname	156
8.2.11.8. Load Separate Event Configuration	157
8.2.11.9. Remove All Event Codes	157
8.2.11.10. Set All Event Codes to Defaults	157
8.3. Main Function Buttons	158
8.3.1. Save Values	158
8.3.2. Load Config File	161
8.3.3. Set All Default Values	163
8.3.4. Save and Continue to Script Generator	163
8.3.5. Load and Continue to Script Generator	164
8.3.6. Save and Continue to AF-MATB	164
8.3.7. Load and Continue to AF-MATB	165
9.0 AF-MATB SCRIPT GENERATOR UTILITY	166
9.1. Utility Introduction	166
9.2. 6 Components of the Script Generator Utility	166
9.2.1. Variables Currently Loaded	167
9.2.1.1. Parameters List	167
9.2.1.2. Reset Script Generator Utility	168
9.2.2. Script Parameters	168
9.2.2.1. Condition Name	169
9.2.2.1.1. Entry Field	169
9.2.2.1.2. "Condition Name" Dropdown	169
9.2.2.1.3. Create New Condition Button	171
9.2.2.1.4. Delete Current Condition Button	171
9.2.2.2. Condition Length	172
9.2.2.3. Communications Subtask	172
9.2.2.4. Tracking Subtask	173
9.2.2.5. System Monitoring Subtask	173
9.2.2.6. Resource Management Subtask	175

9.2.2.7. Subtask Component Visibility	177
9.2.2.8. Script Parameter Errors	178
9.2.3. Conditions That Will Comprise This Script	179
9.2.4. Event Timeline Information	183
9.2.4.1. "Event" Lists and Timeline Modification	184
9.2.4.2. Timeline Selector and Review Timeline Button	188
9.2.5. Schedule Custom Events	196
9.2.6. Main Function Buttons	199
9.2.6.1. Save Script File	199
9.2.6.2. Load Config or Script File	202
9.2.6.3. Save & Continue to AF-MATB	206
9.2.6.4. Load & Continue to AF-MATB	207
9.3. Excel Script Details	208
References	
Glossary	

Figure 001 – The AF-MATB window once loading has been completed.	1
Figure 002 – The entire System Monitoring subtask in the AF-MATB	3
Figure 003 – The group of gauges used in the System Monitoring subtask	4
Figure 004 – The maximum and minimum points of normal operation for a gauge	4
Figure 005 – The minimum and maximum points of the indicator during a malfunction for	
the lower and upper portion of the gauge	5
Figure 006 – A gauge with a malfunction that was properly corrected	5
Figure 007 – The Lights portion of the System Monitoring subtask with both lights shown	
to be functioning normally.	6
Figure 008 – The Lights portion of the Systems Monitoring subtask with both lights	
malfunctioning	6
Figure 009 – The System Monitoring subtask under Automated Mode. Note the lack of	
"F5" and "F6" GUI buttons	7
Figure 010 – The System Monitoring subtask in Automated Mode, with the automation	
correctly detecting a Gauge 4 malfunction, as indicated by Light 2	8
Figure 011 – The System Monitoring subtask in Automated Mode, illustrating a Gauge 1	
automation failure	9
Figure 012 – The Resource Management and Pump Status windows of AF-MATB, with	
Pumps 1 through 6 on	10
Figure 013 – An example of Tank A and Tank B from two different trials illustrating the	
rendering of the researcher-specified target volume (red rectangle) and	
acceptable fuel range (surrounding blue rectangle) of the tanks	11
Figure 014 – The Resource Management and Pump Status windows illustrating a Pump 1	
failure	12
Figure 015 – The Resource Management and Pump Status windows illustrating a situation	
where the researcher has elected to mask the Resource Management's	
behavior while automated.	13
Figure 016 – The Resource Management and Pump Status windows illustrating a situation	
where the researcher has elected to mask the Resource Management's	
behavior while automated and a Pump Failure has occurred	14
Figure 017 – The Resource Management and Pump Status windows illustrating	
Automation Algorithm 2.	15
Figure 018 – An illustration of an Algorithm 2 automation failure for Tank A. Note that,	
despite the pumps indicating an "off" state, Tank A was overfilled	15
Figure 019 – An illustration of an Algorithm 2 automation failure for Tank B. Note that,	
despite the fact that the pumps should be on, as required by the low volume,	
they remain off	16
Figure 020 – An example of how an automation failure appears to the user under	
Automation Algorithm 2.	17
Figure 021 – The Resource Management algorithm successfully reset by a participant. Note	
the change in color on the <i>Reset</i> button and Tank volume indicators	18
Figure 022 – The Tracking subtask with a small box outline and 3 horizontal crosshairs	4.6
spaced 35 pixels apart	18
Figure $023 -$ The Tracking subtask with a large box outline and 5 horizontal crosshairs	
spaced approximately 50 pixels apart.	19
Figure 024 – The Tracking subtask with a large circle outline	19

Figure 025 – The Tracking subtask with a small circle outline	. 20
Figure 026 – The Communications subtask with the default maximum and minimum	
frequencies shown for the NAV channels and default minimum and	
maximum frequencies for the COM channels	. 21
Figure 027 – The Communications subtask where someone just locked in frequency 121.5	
on channel COM1, as indexed by the green <i>Enter</i> button	. 22
Figure 028 - The default appearance of the Scheduling window with no events scheduled	. 23
Figure 029 – A Scheduling window showing a 5-minute condition with the Tracking	
subtask operating at Low Difficulty and 8 CEs.	. 24
Figure 030 – A Scheduling window showing a 5-minute condition with the Tracking	
subtask operating at Moderate Difficulty and 16 CEs.	. 24
Figure 031 – A Scheduling window showing a 5-minute condition with the Tracking	
subtask operating at High Difficulty and 22 CEs	. 25
Figure 032 – A Scheduling window showing a 5-minute condition with the Tracking	
subtask operating in Automated Mode and 8 CEs.	. 25
Figure 033 – A Scheduling window showing three 2-minute conditions comprising one	
trial. The Tracking subtask is operating at Low Difficulty, in Automated	
Mode, and then at High Difficulty, with 8, 5, 8 CEs for three conditions,	
respectively	. 26
Figure 034 – A Scheduling window showing the start of a trial (Figure 034a) and the same	
trial 2 minutes later (Figure 034b)	. 26
Figure 035 – The appearance of the AF-MATB when IT Mode is enabled	. 28
Figure 036 – AF-MATB, rendered with all windows hidden	. 29
Figure 037 – The NASA-TLX launched during a trial	. 30
Figure 038 – Part 1 of the NASA-TLX.	. 31
Figure 039 – One workload pairing in the Sources of Workload (SoW), Part 2 of the NASA-TLX	. 32
Figure 040 – The AF-MATB with the System Monitoring, Tracking, and Resource	
Management subtasks operating in Automated Mode. Note the color of the	
name banners for these three subtasks has changed from the normal blue	
color to green	. 33
Figure 041 – National Instruments' Measurement & Automation Explorer illustrating the	
connection of a NI-USB 6008, nicknamed "Dev1."	. 35
Figure 042 – An example of how to rename DAQs to match previously defined AF-MATB	
parameters.	. 36
Figure 043 – The MCR Installer icon on a black desktop background.	. 37
Figure 044 – The Visual C++ Component Runtime installation window.	. 37
Figure 045 – The Mathworks Installer screen for the MCR	. 38
Figure 046 – Click "Install" to install the MCR so that the 3 included .exe files can be used	. 38
Figure 047 – The three executables and the AF-MATB System Files folder in a directory.	•
The installation is now complete	. 39
Figure $048 - 1$ he six buttons in the task window that the participant can click for the	4.1
System Monitoring subtask.	.41
Figure $049 - 1$ ne eight buttons in the task window that can be clicked for the Resource	10
Figure 050 The Deset button of the Deserves Management subtable	.45
rigure 050 – The <i>keset</i> button of the kesource Management subtask	. 43

Figure 052 – The appearance of the Communications subtask under the "Number Pad" entry method
Figure 053 – The AF-MATB window immediately after pressing <i>Space</i> , demonstrating the timer
Figure 053 – The AF-MATB window immediately after pressing <i>Space</i> , demonstrating the timer
Eigure $054$ The AE MATD window often pressing $E_{22}$
FIGURE UD4 = I DE AE-MATE WINDOW aller Dressing Esc. 41
Figure 055 – The space below the System Monitoring subtask, demonstrating what the
window looks like if the timer and date are disabled and the task is in a
"suspended" state
Figure 056 – The AF-MATB window after <i>Esc</i> was pressed again, placing the task back
into a "ready" state
Figure 057 – The task when paused
Figure 058 – The task window with file selection window after pressing <i>Home</i>
Figure 059 – AF-MATB determined new parameters were loaded using the <i>Home</i> key,
displaying the appropriate message
Figure 060 – AF-MATB determined a new script was loaded using the Home key,
displaying the appropriate message
Figure 061 – Failure to load a file
Figure 062 – Notification message indicating that the user loaded a Config File that was
corrupt or partially damaged 54
Figure 063 – Notification message indicating that the user loaded a Script File that was
missing one or more parameters55
Figure 064 – Notification message indicating that all parameters from a file were loaded in,
but that the script in this file, if present, may be damaged
Figure 065 – Notification message indicating that a Script File contained both missing or
damaged parameters as well as missing or damaged script information
Figure 066 – The Participant ID dialog box
Figure 067 – Dialog box asking if you would like to load a Config File or Script File
Figure $068 - Dialog box which asks which custom file you would like to load$
Figure 069 – Standard file selection utility in Windows which you can use to load custom
Eigune 070 When a file has been successfully leaded, this massage will be displayed 69
Figure $0/0 - w$ nen a me has been successfully loaded, this message will be displayed
Figure 0/1 – Dialog box nonlying the user that no script could be found
Figure $0/2$ – Loading was not successful due to a compt script of parameters messing
from the Config File that was loaded 60
Figure 074 Message notifying the user that the Script File loaded contained one or more
missing or damaged parameters 69
Figure $0.75$ – Message notifying the user that the Script File loaded contained missing or
damaged script information 70
Figure 076 – Message notifying the user that the Script File loaded contained missing or
damaged parameters, as well as missing or damaged script information 70
Figure 077 – Message notifying the user that their data was unable to be successfully
imported

Figure 078 – Message notifying the user that when attempting to import a Script File from	
the Script Generator Utility, the task determined that the imported Script File	
contained one or more damaged or missing parameters	. 71
Figure 079 – Message notifying the user that when attempting to import a Script File from	
the Script Generator Utility, the task determined that the imported Script File	
contained damaged or missing script information.	. 71
Figure 080 – Message notifying the user that information could not be successfully	
imported from one of the companion utilities.	. 71
Figure 081 – Standard directory selection GUI in Windows used to identify the location of	
the AF-MATB System Files folder	. 72
Figure 082 – Message notifying the user that the task is loading the libraries required to	
initialize a digital connection	72
Figure 083 – Message notifying the user of a successful serial port connection	73
Figure 084 – Message notifying the user of a successful digital port connection	73
Figure 085 – Message notifying the user of a failed serial port connection	73
Figure 086 – Message notifying the user of a total failure to establish a digital port	. 15
connection	73
Figure 087 Error code dialog produced by a misconfiguration of the task	. 73
Figure 089 "Saving In Progress" message notifying the user to please wait until the data	. 75
has finished saving	74
Figure 090 Deformence directory generated when the user does not lead any script into	. /4
rigule 089 – Performance directory generated when the user does not load any script into	75
Eigure 000 The root of a Single Condition Standard Trial norformance directory	. 13
Figure 090 – The root of a Single Condition, Standard That performance directory.	. 70
Figure 091 – Contents of a Condition directory for a Standard That performance	77
Eigune 002 The rest of a Multiple Condition Stondard Trial performance directory. Note	. //
Figure $092 - 1$ he root of a Multiple Condition, Standard Trial performance directory. Note	70
Eisen 002 Contents of a "Condition" directory for a Sectory Astronomy Trial	. /8
Figure 093 – Contents of a Condition directory for a System Automation I rial	70
Final Structure Contraction Structure Structur	. 79
Figure 094 – The root of a Multiple Condition, System Automation Trial performance	
directory. Note the additional System Automation Performance Summary	0.0
	. 80
Figure 095 – Contents of a "Condition" directory for a Resource Management Automation	~ .
Trial performance directory	. 81
Figure 096 – The root of a Multiple Condition, Resource Automation Trial performance	
directory. Note the additional Resource Automation Performance Summary	
Log	. 82
Figure 097 – Contents of a "Condition" directory for a Dual Automation Trial performance	
directory	. 83
Figure 098 – The root of a Multiple Condition, Dual Automation Trial performance	
directory. Note the additional Resource Automation and System Automation	
Performance Summary Logs.	. 84
Figure 099 – Contents of a "Condition" directory for an IT Mode Trial performance	
directory	. 85
Figure 100 – The root of a Multiple Condition, IT Mode Trial performance directory. Note	
the additional Information Throughput Summary Logs	. 86

Figure 101 – The appearance of the AF-MATB Configuration Utility when launched	104
Figure 102 – Ten parameter groups are available to select from the dropdown menu in the AF-MATB Configuration Utility.	105
Figure 103 – The appearance of the AF-MATB Configuration Utility after the "AF-MATB	
System Parameters" selection was selected from the Group Dropdown.	106
Figure 104 – The user made an invalid entry in the field of one parameter, and omitted a	
value for the field of another parameter. Note the appearance of the error	
dialog list left side of the window.	107
Figure 105 – Close-up of the error dialog list, illustrating the problems for that parameter	
group, as well as the acceptable values for that group.	107
Figure 106 – Options available in the AF-MATB System Parameters group in the AF-	
MATB Configuration Utility	108
Figure 107 - Input Options parameter in the AF-MATB System Parameters group	109
Figure 108 – Warning message displayed when users click "GUI Buttons Only" to the	
Input Options parameter	109
Figure 109 – Enable Pausing? parameter in the AF-MATB System Parameters group	110
Figure 110 – Enable Manual Event Triggering? parameter in the AF-MATB System	
Parameters group.	110
Figure 111 – Enable Manual Pump Repair? parameter in the AF-MATB System	
Parameters group.	110
Figure 112 – How the Enable Manual Pump Repair? parameter appears when the Enable IT	
Mode? parameter has been set to "Yes."	111
Figure 113 – Display Date and Time? parameter in the AF-MATB System Parameters	
group	111
Figure 114 – Enable Scheduling? parameter in the AF-MATB System Parameters group	111
Figure 115 – Enable Escape key? parameter in the AF-MATB System Parameters group	111
Figure 116 – Enable Home key? parameter in the AF-MATB System Parameters group	112
Figure 117 – Timer Cycling Rate (Hz) parameter in the AF-MATB System Parameters	
group	112
Figure 118 – Initial Ramp-up Time Delay (Seconds) parameter in the AF-MATB System	
Parameters group.	112
Figure 119 – NASA-TLX Ramp-Up Time Delay (Seconds) parameter in the AF-MATB	
System Parameters group.	113
Figure 120 – Resource Management Automation Algorithm parameter in the AF-MATB	
System Parameters group.	113
Figure 121 – How the Resource Management Automation Algorithm parameter appears	
when the Enable IT Mode? parameter has been set to "Yes."	113
Figure 122 – Enable Subtask Automation Indicators? parameter in the AF-MATB System	
Parameters group.	114
Figure 123 – Enable Information Throughput Mode? parameter in the AF-MATB System	
Parameters group.	114
Figure 124 – Warning message displayed when users click "Yes" to the Enable Information	
Throughput Mode? parameter.	114
Figure 125 – Options available in the System Monitoring Subtask Basic Parameters group	
in the AF-MATB Configuration Utility.	115

Figure 126 – Gauge Speed Lower Limit (Pixels/Cycle) parameter in the System Monitoring	
Subtask Basic Parameters group 1	115
Figure 127 – Gauge Speed Lower Limit (Pixels/Cycle) parameter in the System Monitoring	
Subtask Basic Parameters group 1	116
Figure 128 – End of Range Delay Max (Cycles) parameter in the System Monitoring	
Subtask Basic Parameters group 1	116
Figure 129 – Correct Fault Identification Pause (Cycles) parameter in the System	
Monitoring Subtask Basic Parameters group 1	116
Figure 130 – Gauge Malfunction Timeout (Seconds) parameter in the System Monitoring	
Subtask Basic Parameters group	117
Figure 131 – Indicator Light Malfunction Timeout (Seconds) in the System Monitoring	
Subtask Basic Parameters group	117
Figure 132 – The Tracking Subtask Basic Parameters group in the AF-MATB	
Configuration Utility	118
Figure 133 – Tracking Gain parameter in the Tracking Subtask Basic Parameters group 1	118
Figure 134 – Tracking RMS Interval (Seconds) parameter in the Tracking Subtask Basic	
Parameters group	119
Figure 135 – Center Range (Pixels) parameter in the Tracking Subtask Basic Parameters	
group	119
Figure 136 – Vertical Spacing for Crosshairs (Pixels) parameter in the Tracking Subtask	-
Basic Parameters group	119
Figure 137 – Horizontal Spacing for Crosshairs (Pixels) parameter in the Tracking Subtask	
Basic Parameters group.	120
Figure 138 – Horizontal Crosshair Count parameter in the Tracking Subtask Basic	
Parameters group	120
Figure 139 – Tracking Range Outline Shape parameter in the Tracking Subtask Basic	
Parameters group	120
Figure 140 – The Tracking Range Outline Shape parameter when the Enable IT Mode?	
parameter has been set to "Yes."	121
Figure 141 – Circle Tracking Outline Radius (Pixels) parameter in the Tracking Subtask	
Basic Parameters group	121
Figure 142 – How the Circle Tracking Outline Radius (Pixels) parameter appears when the	
Enable IT Mode? parameter has been set to "Yes."	121
Figure 143 – Square Tracking Outline Width (Pixels) parameter in the Tracking Subtask	
Basic Parameters group	121
Figure 144 – Options available in the Communications Subtask Parameters group in the	
AF-MATB Configuration Utility.	122
Figure 145 – Callsign Label parameter in the Communications Subtask Parameters group 1	123
Figure 146 – Comm Slot 1 Label parameter in the Communications Subtask Parameters	_
group	123
Figure 147 – Comm Slot 2 Label parameter in the Communications Subtask Parameters	
group	123
Figure 148 – Comm Slot 3 Label parameter in the Communications Subtask Parameters	
group	124
Figure 149 – Comm Slot 4 Label parameter in the Communications Subtask Parameters	
group1	124

Figure 150 – Enter Confirmation Highlight (Cycles) parameter in the Communications	
Subtask Parameters group 1	124
Figure 151 – Target Communication Timeout (Seconds) parameter in the Communications	105
Sublask Parameters group.	123
Figure 152 – Communications Event Duration (Seconds) parameter in the Communications Subtask Parameters group	125
Figure 152 Frequency Entry Method perometer in the Communications Subtesk	125
Parameters group	125
Figure 154 – How Frequency Entry Method parameter appears when the Input Options	-
parameter has been set to "GUI Buttons Only."	126
Figure 155 – Slot 1 & 2 Lower Limit parameter for the Communications Subtask	
Parameters group	126
Figure 156 – Slot 1 & 2 Upper Limit parameter for the Communications Subtask	
Parameters group.	126
Figure 157 – Slot 1 & 2 Frequency Increments parameter for the Communications Subtask	
Parameters group	127
Figure 158 – Notification of incongruent Lower Limit, Increment, and Upper Limit	
specifications for Comm Slots 1 & 2.	127
Figure 159 – Slot 3 & 4 Lower Limit parameter for the Communications Subtask	
Parameters group	128
Figure 160 – Slot 3 & 4 Upper I imit parameter for the Communications Subtask	120
Parameters group	170
Figure 161 Slot 2 & 4 Encourage and a neurometer for the Communications Subtask	120
Figure 101 – Slot 5 & 4 Frequency increments parameter for the Communications Subtask	100
Parameters group.	129
Figure 162 – Notification of incongruent Lower Limit, Increment, and Upper Limit	
specifications for Comm Slots 3 & 4.	129
Figure 163 – The Resource Management Subtask Basic Parameters group in the AF-	
MATB Configuration Utility	130
Figure 164 – Pump 1 & Pump 3 Flow Rate (Liters/Minute) parameter for the Resource	
Management Subtask Basic Parameters group	131
Figure 165 – Pump 2 & Pump 4 Flow Rate (Liters/Minute) parameter for the Resource	
Management Subtask Basic Parameters group.	131
Figure 166 – Pump 5 & Pump 6 Flow Rate (Liters/Minute) parameter for the Resource	
Management Subtask Basic Parameters group 1	131
Figure 167 – Pump 7 & Pump 8 Flow Rate (Liters/Minute) parameter for the Resource	
Management Subtask Basic Parameters group.	132
Figure 168 – Tank A & Tank B Drop Rate (Liters/Minute) parameter for the Resource	
Management Subtask Basic Parameters group.	132
Figure 169 – Tank A & Tank B Maximum parameter for the Resource Management	-
Subtask Basic Parameters group	132
Figure 170 – Tank C & Tank D Maximum parameter for the Resource Management	132
Subtask Pasia Darameters group	122
Figure 171 Tank A Starting Volume (Liters) parameter for the Descurse Management	155
Subtook Posic Deremeters group	122
Europe 172 Taula D. Stanting Values (Liters)	133
Figure 1/2 – Tank B Starting volume (Liters) parameter for the Resource Management	100
Subtask Basic Parameters group	133

Figure 173 – Tank C Starting Volume (Liters) parameter for the Resource Management	
Subtask Basic Parameters group134	4
Figure 174 – Tank D Starting Volume (Liters) parameter for the Resource Management	
Subtask Basic Parameters group	4
Figure 175 – Pump Failure Duration (Seconds) parameter for the Resource Management	
Subtask Basic Parameters group	4
Figure 176 – The System Monitoring Subtask Automation Parameters group in the AF-	
MATB Configuration Utility13	5
Figure 177 – The Automation Fault Restore Time (Seconds) parameter in the System	
Monitoring Subtask Automation Parameters group	5
Figure 178 – The Automation Failure Duration (Seconds) parameter in the System	
Monitoring Subtask Automation Parameters group	5
Figure 179 – The Tracking Subtask Difficulty Parameters group in the AF-MATB	
Configuration Utility	6
Figure 180 – Low Difficulty Reticle Speed Lower Limit (Pixels/Cycles) parameter in the	
Tracking Subtask Difficulty Parameters group.	7
Figure 181 – Low Difficulty Reticle Speed Upper Limit (Pixels/Cycles) parameter in the	
Tracking Subtask Difficulty Parameters group.	7
Figure 182 – Low Difficulty Direction Change Lower Limit (Cycles) parameter in the	
Tracking Subtask Difficulty Parameters group.	8
Figure 183 – Low Difficulty Direction Change Upper Limit (Cycles) parameter in the	
Tracking Subtask Difficulty Parameters group.	8
Figure 184 – Moderate Difficulty Reticle Speed Lower Limit (Pixels/Cycles) parameter in	
the Tracking Subtask Difficulty Parameters group.	8
Figure 185 – Moderate Difficulty Reticle Speed Upper Limit (Pixels/Cycles) parameter in	
the Tracking Subtask Difficulty Parameters group.	9
Figure 186 – Moderate Difficulty Direction Change Lower Limit (Cycles) parameter in the	
Tracking Subtask Difficulty Parameters group.	9
Figure 187 – Moderate Difficulty Direction Change Upper Limit (Cycles) parameter in the	
Tracking Subtask Difficulty Parameters group.	9
Figure 188 – High Difficulty Reticle Speed Lower Limit (Pixels/Cycles) parameter in the	
Tracking Subtask Difficulty Parameters group.	0
Figure 189 – High Difficulty Reticle Speed Upper Limit (Pixels/Cycles) parameter in the	
Tracking Subtask Difficulty Parameters group.	0
Figure 190 – High Difficulty Direction Change Lower Limit (Cycles) parameter in the	
Tracking Subtask Difficulty Parameters group.	0
Figure 191 – High Difficulty Direction Change Upper Limit (Cycles) parameter in the	
Tracking Subtask Difficulty Parameters group.	1
Figure 192 – The Resource Management Subtask Automation Parameters group in the AF-	
MATB Configuration Utility142	2
Figure 193 – Tank Volume Update Interval (Seconds) parameter in the Resource	
Management Subtask Automation Parameters group	2
Figure 194 – Main Tank Automation Lower Limit (Liters) parameter in the Resource	
Management Subtask Automation Parameters group	3
Figure 195 – Main Tank Automation Upper Limit (Liters) parameter in the Resource	
Management Subtask Automation Parameters group.	4

Figure 196 –	Supply Tank Automation Lower Limit (Liters) parameter in the Resource	
	Management Subtask Automation Parameters group.	144
Figure 197 –	Supply Tank Automation Upper Limit (Liters) parameter in the Resource	
	Management Subtask Automation Parameters group.	144
Figure 198 –	Autopilot Maximum Tank Difference (Liters) parameter in the Resource	
	Management Subtask Automation Parameters group.	145
Figure 199 –	Resource Management RMS Interval (Seconds) parameter in the Resource	
	Management Subtask Automation Parameters group.	145
Figure 200 –	RMS Target Value (Liters) parameter in the Resource Management Subtask	
<b>D</b> ' <b>0</b> 01	Automation Parameters group.	146
Figure 201 –	Hide Automation Behavior? parameter in the Resource Management Subtask	110
E	Automation Parameters group.	146
Figure 202 –	How the Hide Automation Benavior? parameter appears when the Resource	
	a "	116
Eiguro 202	2. Automation Eault Duration (Seconds) parameter in the Desource Management	140
Figure $203 =$	Subtask Automation Darameters group	1/7
Figure 204 _	Subtask Visibility Parameters window in the AF-MATB Configuration	14/
1 Iguie 204 –	Utility	148
Figure 205 –	Initially hide System Monitoring subtask? parameter in the Subtask Visibility	1-0
I Iguie 205	Parameters group	148
Figure 206 –	Initially hide Tracking subtask? parameter in the Subtask Visibility	110
1 iguie 200	Parameters group	149
Figure 207 –	Initially hide Communications subtask? parameter in the Subtask Visibility	,
8	Parameters group	149
Figure 208 –	Initially hide Communications subtask? parameter in the Subtask Visibility	
C	Parameters group.	149
Figure 209 –	Initially hide Scheduling window? parameter in the Subtask Visibility	
	Parameters group	149
Figure 210 –	Initially hide Pump Status window? parameter in the Subtask Visibility	
	Parameters group	150
Figure 211 –	The Port Triggering Parameters group in the AF-MATB Configuration	
	Utility	151
Figure 212 –	The Event List, used in the Port Triggering Parameters group to show the	
	event codes of every event in AF-MATB.	152
Figure 213 –	The Enable Port Triggering? parameter in the Port Triggering Parameters	
<b>D</b> ' 014	group	153
Figure 214 –	The Triggering Mechanism parameter in the Port Triggering Parameters	1 5 0
<b>D'</b> 015	group.	153
Figure 215 –	A user clicking on the first entry ("FI Gauge Fault (Normal)") in the Event	1 ~ 1
Eigung 016	List, with the current value rendered in the Change Event Code field.	154
rigure 216 –	A user emering in a new event code for the FI Gauge Fault (Normal) <sup>2</sup> entry.	
	Change Event Code button Note that the event code for this entry in the	
	Event List is still "1"	151
		134

Figure 217 – A user removing the event code for the "F1 Gauge Fault (Normal)" entry.	
They have cleared the event code in the entry field, but have not hit the	
Change Event Code button. Note that the event code for this entry in the	
Event List is still "1."	. 154
Figure 218 – A user confirming the new event code for the "F1 Gauge Fault (Normal)"	
entry. Note that the event code for this entry is now "1234."	. 154
Figure 219 – A user confirming the removal of an event code for the "F1 Gauge Fault	
(Normal)" entry. Note that the event code for this entry is now "NaN."	. 155
Figure 220 – Notification message explaining to the user that they have entered an invalid	
event code	155
Figure 221 – The Serial Port Address parameter in the Port Triggering Parameters group	155
Figure 222 – The Digital Port Address parameter in the Port Triggering Parameters group	156
Figure 222 The Digital Device Nickname parameter in the Port Triggering Parameters	. 150
aroup	156
Figure 224 Typical file selection GUI that is launched after pressing the Load Senarate	. 150
Fugure 224 – Typical file selection GOT that is fauthened after pressing the Lotal separate	157
Event Configuration buttons located on the right of the AE MATE	. 137
Configuration Utility window	150
Figure 226 The AE MATE Configuration Utility showing the user attempting to save a	. 130
Config File with a missing value for one of the peremeters in the AE MATE	
Config File with a missing value for one of the parameters in the AF-WATD	
System Parameters group. Note the text in the lower left of the window	150
Figure 227 The CLU that allows users to gove Config Files from this utility. Note the	. 139
Figure 227 – The GUI that allows users to save Config Files from this utility. Note the	
Config File known as Test1 already saved in this directory. Specifically,	
note the addition of a date-and-time stamp, as well as "_CONFIG" appended	1.00
to the file name. $\Box$	. 160
Figure 228 – The Configuration Utility after the user has successfully saved a Config File,	1.61
as notified by the status text.	. 161
Figure 229 – The AF-MATB Configuration Utility notifying the user that the loading	1.00
Figure 200 The AE MATE Construction of the status text.	. 162
Figure 230 – The AF-MATB Configuration Utility, notifying the user that the loading	1.40
process was not completed via the status text.	. 162
Figure 231 – The AF-MATB System Parameters group after the Set All Default Values	
button has been pressed. Note the status text notifying the user of	
confirmation that the action was completed	. 163
Figure 232 – The Configuration Utility notifying the user that the AF-MATB Script	
Generator Utility will launch momentarily via the status text.	. 164
Figure 233 – The Configuration Utility notifying the user that AF-MATB will launch	
momentarily via the status text	. 165
Figure 234 – The AF-MATB Script Generator Utility	. 166
Figure 235 – The Script Generator Utility, highlighting the "Variables Currently Loaded"	
component	. 167
Figure 236 – The Script Generator Utility, highlighting the "Script Parameters" component.	. 168
Figure 237 – The "Condition Name" portion of the "Script Parameters" component of the	
AF-MATB Script Generator Utility.	. 169
Figure 238 – An example demonstrating the user creating a condition named "Test1."	. 169

Figure 239 – The user click on the "Condition Name" dropdown to save the values entered into the "Script Parameters" component.	. 170
Figure 240 – The user has clicked on the condition originally labeled "Condition 1" and	, .
reads "Test1"	170
Figure 241 – Status text confirming to the user that the condition was saved	. 170
Figure 242 – The "Condition Name" dropdown, demonstrating 3 conditions that are	
currently stored in the utility. As illustrated by the blue highlighting and the	
current condition "Test1," the user is preparing to switch from "Test1" to	
"Test2."	. 170
Figure 243 – A section of the "Script Parameters" component after the user has used the	
"Condition Name" dropdown to switch conditions. Not the change in information in the "Seconda" "Targets" and "Distructors" fields from Figure	
242	171
Figure $244 - A$ user ready to create a new condition in the Script Generator Utility	171
Figure 245 – The "Script Parameters" component after the user has pressed the <i>Create New</i>	. 1 / 1
Condition button.	. 171
Figure 246 – The "Condition Length" portion of the "Script Parameters" component of the	
AF-MATB Script Generator Utility.	. 172
Figure 247 – The "Communications Subtask" portion of the "Script Parameters"	
component of the AF-MATB Script Generator Utility	. 172
Figure 248 – The "Condition Name" portion of the "Script Parameters" component of the	. – –
AF-MATB Script Generator Utility.	. 173
Figure 249 – The "System Monitoring Subtask" portion of the "Script Parameters"	
component of the AF-MATB Script Generator Utility when configured for Manual Mode	172
Figure 250 The "System Monitoring Subtask" portion of the "Script Parameters"	. 175
component of the AF-MATB Script Generator Utility when configured for	
Automated Mode.	. 174
Figure 251 – The "System Monitoring Subtask" portion of the "Script Parameters"	
component of the AF-MATB Script Generator Utility when IT Mode is	
enabled	. 174
Figure 252 – The "Resource Management Subtask" portion of the "Script Parameters"	
component of the AF-MATB Script Generator Utility when configured for	175
Manual Mode.	. 175
Figure 253 – The Script Generator Utility warning message when user schedules a large	175
Figure 254 – The "Resource Management Subtask" portion of the "Script Parameters"	. 175
component of the AF-MATB Script Generator Utility when configured for	
Automated Mode	. 176
Figure 255 – The "System Monitoring Subtask" portion of the "Script Parameters"	
component of the AF-MATB Script Generator Utility when IT Mode is	
enabled	. 176
Figure 256 – The "Subtask Component Visibility" portion of the "Script Parameters"	
component of the AF-MATB Script Generator Utility	. 177

Figure 257 – The "Script Parameter Errors" portion of the "Script Parameters" component of the AF-MATB Script Generator Utility	8
Figure 258 – The "Script Parameters" component, illustrating a case in which the user has exceeded the number of possible event occurrences for the Communications subtask, entered an unacceptable value for the Gauges component of the System Monitoring subtask, and omitted the Resource Management subtask's "Shut-Offs" field. Note the text labels for the fields have turned red to highlight the issue	8
Figure 259 – The "Script Parameter Errors" portion of the "Script Parameters" component. Note the verbose error descriptions designed to help the user understand the issue	8
Figure 260 – The Script Generator Utility, highlighting the "Conditions That Will Comprise This Script" component	9
Figure 261 – The Script Generator Utility after the user has pressed the <i>Schedule Current</i> <i>Condition</i> button	0
Figure 262 – The Script Generator Utility, with the user having entered in a third condition that has not yet been saved	1
Figure 263 – The "Conditions That Will Comprise This Script" component, illustrating a script with two different conditions	1
Figure 264 – The "Conditions That Will Comprise This Script" component, illustrating a script with five repeated conditions	2
Figure 265 – Script Generator Utility notification when the user attempts to generate a script with no conditions	2
Figure 266 – The user clicking on the second of four conditions in the condition list	3
listed conditions	3
component	4 1
Figure 200 – The "Event Timeline Information" component after the user has clicked on the "Event Time" item	5
Figure 271 – The user is about to change the event time of from 37.2507 to 10 seconds	5 6 7
Figure 275 – Value the user is attempting to update that event s obset time to matter in 18 Figure 274 – The Script Generator utility after the user enters an invalid value	7 7
Figure 276 – Another close-up of the "Script Parameter Errors" list after the user attempted to change an event onset time to an invalid value, in this case illustrating a different acceptable upper limit to account for an event's duration	7 7
Figure 277 – The "Event Timeline Information" component showing a partial timeline after the user has selected the "Show Selected Condition" button	8
Figure 278 – The "Event Timeline Information" component showing a partial timeline after the user has selected the "Show Selected Condition" button. In this figure, the user clicked on "Test1" in the condition list and then scrolled to the	

	bottom of the event information lists to show that the lists are only showing	
	the timeline for the first condition	188
Figure 279 -	- The event distribution for the Communications subtask. Note the different	
	colors used to indicate TCs and FCs.	190
Figure 280 -	- The event distribution for the Lights component of the System Monitoring	
-	subtask	190
Figure 281 -	- The event distribution of the Gauges component of the System Monitoring	
C	subtask	191
Figure 282 –	- The event distribution of the Pump Failures for the Resource Management	
C	subtask	191
Figure 283 –	- The event distribution of the Pump Shut-Offs for the Resource Management	
U	subtask	192
Figure 284 –	- The event distribution of the System Monitoring subtask when in Automated	
8	Mode	192
Figure 285 –	The event distribution of automation failures for the Resource Management	-
8	subtask when in Automated Mode using Automation Algorithm 2	193
Figure 286 –	The distribution approval dialog box.	193
Figure 287 –	- The event distribution of the Gauges component of the System Monitoring	170
8	subtask's after the user disapproved of the initial distribution	194
Figure 288 –	- The event distribution for the Communications subtask over a 30-minute trial	195
Figure 289 –	- The event distribution of the Pump Failures for the Resource Management	170
115010 209	subtask across two conditions. One condition shows a favorable distribution	
	while the other does not	195
Figure 290 -	- The Script Generator Utility highlighting the "Schedule Custom Events"	170
119410 200	component	196
Figure 291 -	- The user entering in information for a task pause before pressing the <i>Confirm</i>	170
115010 291	Selection button	196
Figure 292 -	- The user successfully entering information for a task nause	197
Figure 293 –	- The Script Generator Utility after the user unsuccessfully attempts to add a	177
1 iguie 295	custom event	197
Figure 294 -	- The user entering information for a NASA-TLX before pressing the <i>Confirm</i>	177
I iguie 29 i	Selection button	198
Figure 295 -	- The user successfully entering information for a NASA-TLX	198
Figure 296 –	- The Script Generator Utility highlighting the utility's main function buttons	170
I iguie 290	and status text notifying the user of the utility's current state	199
Figure 207	File saving GUI launched by the Scrint Generator Utility when the Sava	1))
11guie 277 -	Script File or Save & Continue to AF MATE buttons are pressed	200
Figure 208	The Script Generator Utility after the Save Script File and Save & Continue to	200
11guie 298 -	AE MATR buttons are pressed	201
Figure 200	The Script Generator Utility after the script has been successfully saved	201
Figure 299 -	File selection CIII lounched if the user presses the Load Config or Seriet File	201
Figure 500 -	or Load & Continue to AE MATP byttons	202
Figure 201	The Seriet Concreter Utility ofter the year has loaded a Config Eile	203
Figure 301 -	- The Script Generator Utility after the user has loaded a Config File	204
$r_1 \sigma_1 r_2 \rightarrow 0 / -$	The Comment Company on I Itility often a Comment File was a second file 1 - 1 - 1	205
$\Gamma$ iguit $302 - \Gamma$	- The Script Generator Utility after a Script File was successfully loaded	205

Figure 304 – The Script Generator Utility, notifying the user that AF-MATB is loading via	
the status text.	. 206
Figure 305 – The Script Generator Utility after the user attempts to use the Load &	
Continue to AF-MATB button to load a Config File.	. 207

#### Acknowledgements

We would like to thank Sean Weston, Major Dawn Russell, Dr. Scott Galster, and Dr. James Christensen for their time, guidance, and support during the development of the software and technical manual.

THIS PAGE INTENTIONALLY LEFT BLANK

#### 1.0 **INTRODUCTION**

In 2010, the AFRL's 711<sup>th</sup> Human Performance Wing's Human Effectiveness Directorate introduced an update to the original Multi-Attribute Task Battery (MATB) developed in by 1992 J. Raymond Comstock and Ruth J. Arnegard. This version, known as AF-MATB, provided a needed update the original task, allowing it to run on a new generation of computers as well as adding more functionality and experimental control than was previously available. Additions such as more detailed task information for performance analysis, greater task portability, and tools that allowed researchers to generate scripts almost instantly, all served to make life easier for the researcher.

Since the updated version's initial release in 2010, we have continued to improve the AF-MATB, starting with a redesign of the Configuration and Script Generator utilities by adding additional configurability to the task and more control over the scripts that were being generated. Next, we added serial and digital port-triggering to AF-MATB, allowing the task to interface with neurophysiological acquisition systems. This change will facilitate time-syncing between AF-MATB and acquired neurophysiological data and will aid the integration of state-based information into acquired data. Finally, some new modes of operation requested by our user base, such as the version of MATB used to investigate questions related to automation by Molloy & Parasuraman (1996), were added.

As in our previous release of AF-MATB, this release consists of six windows that comprise the total battery (see Figure 001). Each of the four subtasks is represented in a separate window; these subtasks include: System Monitoring, Communications, Resource Management, and Tracking. The last two windows, which contain Scheduling and the Pump Status information, are resources that the participant can use to improve performance during the task. Each of these windows will be discussed later in more detail.





Figure 001 - The AF-MATB window once loading has been completed.

#### 2.0 SYSTEM REQUIREMENTS

Minimum hardware requirements for the AF-MATB and its associated utilities include 2 GB of RAM, a 2 GHz dual-core processor, a keyboard, mouse, and joystick, and a 15-inch monitor (1280x1024 resolution). It is important that these minimum requirements be met to ensure the task functions properly. Executing the task on machines that do not meet these minimum requirements may produce errors or unstable task behavior, such as abnormally sluggish movement and/or response of the tracking reticle and fuel level and gauge indicators. There may also be periods of freezing in the software caused by audio playback. Further, the use of machines that do meet these requirements may affect the reliability of the event-times recorded.

As mentioned previously, a keyboard, mouse and joystick are all required for the task to be properly executed. A mouse is required to initiate the software, and can also be used, in the task, to respond to events. The keyboard is necessary to manually trigger events and AF-MATB System Commands, which will be discussed in depth later. Finally, the joystick is required for operation of the Tracking Task. In the event that a joystick is not present at the start of the task, the participant will be notified that no joystick has been detected, and the tracking reticle will float freely around the screen. Please be aware that the joystick must be connected to the computer prior to execution of the task. If the joystick is connected after the task has been executed, the joystick will not be detected.

The AF-MATB software package was designed for Windows computers running Windows 7 or higher using the MATLAB Compiler Runtime 7.8. This package, particularly the Script Generator utility, also requires Microsoft Office 2007 or later.

Please ensure that your DPI settings are set to default values. If your DPI is set to custom values, you may experience window distortion.

#### 3.0 AF-MATB SUBTASK DESCRIPTIONS

#### 3.1. System Monitoring

The System Monitoring subtask is located in the upper left section of the AF-MATB window. It consists of two components: Gauges and Lights.



Figure 002 – The entire System Monitoring subtask in the AF-MATB.

However, unlike previous versions, this subtask now has two different modes of operation; the Manual Mode, which is identical to the traditional behavior documented in Miller (2010) and Comstock and Arnegard (1992), and the Automated Mode, which behaves similar to the System Monitoring subtask outlined in Molloy and Parasuraman (1996).

For detailed information on the commands associated with performing this subtask please see: **6.1 System Monitoring Commands**.

#### 3.1.1. **Manual Mode**

#### 3.1.1.1. Gauges

The Gauges component (see Figure 003) consists of four gauges located in the System Monitoring window of AF-MATB. The participant is asked to observe the fluctuations of the gauge indicators to identify gauge malfunctions.



Figure 003 – The group of gauges used in the System Monitoring subtask.

The indicators of gauges that are operating normally will only fluctuate one tick mark above and below the center line. Fluctuations beyond that range are considered malfunctions. Gauge indicator will pause at the extremes of the operating range before changing direction.



Figure 004 – The maximum and minimum points of normal operation for a gauge.

The two images above (Figure 004) illustrate the maximum and minimum value of each gauge when operating normally, respectively. A gauge malfunction is defined as any time when the gauge indicator fluctuates outside of the previously defined normal operating range.



Figure 005 – The minimum and maximum points of the indicator during a malfunction for the lower and upper portion of the gauge.

A gauge malfunction is characterized by the gauge indicator's smooth transition out of the normal operating range and into either the upper or lower malfunctioning ranges. For the duration of the malfunction, the indicator will continuously alternate between the malfunctioning range and the normal range, as illustrated in Figure 005. The algorithm that governs the gauge behavior is designed such that a scheduled malfunction will occur in the direction that the indicator is already traveling. For instance, if the indicator is traveling up when the malfunction is scheduled to occur, the indicator will stay in the upper malfunctioning range. Conversely, if the indicator is traveling down when the malfunction is scheduled to occur, the indicator will stay in the lower malfunction will stay in the lower malfunction is down when the malfunction is above (Figure 005) illustrate the minimum and maximum points of the malfunctioning range that can occur in the lower portion and upper portion of the gauge, respectively.

If the malfunction is corrected within the time specified, the gauge indicator will automatically return to the center of the gauge, and stay there for a specified duration. Additionally, a yellow bar will appear at the bottom of the gauge, as illustrated (Figure 006). This serves as a cue to the participant that their response was correct and timely.



Figure 006 – A gauge with a malfunction that was properly corrected.

If the participant fails to respond before the malfunction terminates, the lack of response will be recorded as a timeout. If the participant responds after the specified timeframe, the response will be recorded as a false alarm. If a response is not made in the correct timeframe, the gauge indicators will automatically resume normal functioning with no indication that a malfunction occurred.

For detailed information on modifying the parameters associated with governing the behavior of the Gauges under Manual Mode, please see 8.2.3 System Monitoring Subtask Basic **Parameters Group.** 

#### 3.1.1.2. Lights

The Lights subtask consists of two indicator lights located in the System Monitoring window of AF-MATB.



Figure 007 – The Lights portion of the System Monitoring subtask with both lights shown to be functioning normally.

The above figure (Figure 007) illustrates the normal operation of the two lights. The light located on the left, designated Light 1, is green or "on" and the light located on the right, designated Light 2, is black or "off."

When Light 1 malfunctions, the green light turns off (i.e., turns black). When Light 2 malfunctions, the light turns on (i.e., turns red). The figure below (Figure 008) illustrates how each light looks when it malfunctions. Thus, if the participant were to encounter the situation below, responses would be required for each light.



Figure 008 – The Lights portion of the Systems Monitoring subtask with both lights malfunctioning.

If the appropriate response is recorded within the malfunction duration, the malfunction will be corrected, and the light will be restored to normal function.

If no response is recorded within the specified duration, or if a response is recorded after the timeframe, it will be recorded as a timed-out response or a timed-out response and a false alarm, respectively. In the event that a response is not made in the correct timeframe, the lights will automatically resume their normal functioning behavior.

For detailed information on modifying the parameters associated with governing the behavior of the Lights under Manual Mode, please see 8.2.3 System Monitoring Subtask Basic **Parameters Group.** 

#### **3.1.2.** Automated Mode



Figure 009 - The System Monitoring subtask under Automated Mode. Note the lack of "F5" and "F6" GUI buttons.

When the System Monitoring subtask is automated, the behavior of the Lights and Gauges components varies slightly. Instead of treating these two components as separate, when automation is engaged, these two components work together.

The behavior of the Gauges component remains unchanged from Manual Mode, with each of the four gauges oscillating between the normal operating limits. However, the Lights component under Automated Mode serves an entirely different purpose. In this case, as illustrated above (see Figure 009), the *F5* and *F6* GUI buttons have been removed.

In this mode, the Lights, instead of requiring a response, are designed to provide information about the state of the automation to the participant. The automation is designed to automatically detect when a gauge goes out of range and fix the gauge within a few seconds (see Figure 010). In this case, participants are not required to respond to gauge malfunction as Light 2 has indicated that it has detected the problem.



Figure 010 – The System Monitoring subtask in Automated Mode, with the automation correctly detecting a Gauge 4 malfunction, as indicated by Light 2.

However, there can be instances when the automation does not detect a malfunction, as illustrated in Figure 011, where Light 2 does not turn on, despite the fact that a gauge is outside of its normal operating range. In this case, it is the participant's responsibility to manually correct the gauge malfunction using the appropriate response commands.

As in Manual Mode, if the participant correctly identifies an automation failure, the appropriate response will be recorded and that gauge will indicate a correct identification (see Figure 006). If the participant incorrectly identifies an automation fault, or if the participant attempts to restore a gauge already detected by the automation, the response will be counted as a false alarm. Any failure to detect an automation failure within the specified duration will count as a miss.



Figure 011 – The System Monitoring subtask in Automated Mode, illustrating a Gauge 1 automation failure.

For detailed information on modifying the parameters associated with governing the behavior of the Gauges under Automated Mode, please see section **8.2.7 System Monitoring Subtask Automation Parameters Group**.

#### 3.2. Resource Management



Figure 012 – The Resource Management and Pump Status windows of AF-MATB, with Pumps 1 through 6 on.

As in previous versions, the basic goal of the Resource Management subtask is to maintain the two consumption tanks, Tank A and Tank B, at a target volume, or at least within some specified range, using the eight pumps (represented by the eight numbered squares) and four supply tanks. Details on how to instruct participants to best achieve this goal can be found in section **3.2.1 Manual Mode**.

Also, as in previous versions, the Pump Status window, considered the companion window to the Resource Management subtask, is still used to indicate which pumps are active by displaying the flow rate of each pump when active. Please note that this window is only meant to be used as a resource and requires no action on the part of the participant.

The first addition to this subtask in the new version of the software is the implementation of functional target volume and acceptable range indicators. Originally, the red lines and blue rectangles on the left and right sides of Tanks A and B respectively were just non-dynamic indicators of the center fill line of the tanks. However, these values are now dynamically rendered based on the researcher's specifications. As illustrated in Figure 013a-b, the target tank volume for this subtask is now indicated by the red line, while acceptable range of fuel is indicated by the surrounding blue rectangle.


Figure 013 – An example of Tank A and Tank B from two different trials illustrating the rendering of the researcher-specified target volume (red rectangle) and acceptable fuel range (surrounding blue rectangle) of the tanks.

Another new addition to this subtask is the inclusion of a second Automated Mode. While the original Automated Mode introduced in the previous version of AF-MATB has been preserved, we've implemented a second mode with a different set of governing behaviors designed to focus on some different participant interactions. For information on the Automated Modes of this subtask, please see section **3.2.2** Automated Mode.

Finally, while the scoring metrics introduced in the last release have been preserved, some additional metrics will now provide researchers with more accurate information regarding a participant's performance in this subtask. For more information on these metrics and how they're calculated, please see section **7.3.3.1.3 Resource Management Section**.

For detailed information on the commands associated with performing this subtask please see section **6.2 Resource Management Response Commands**.

### 3.2.1. Manual Mode

As previously stated, the goal of this subtask is to maintain the volumes of Tanks A and B at or near the specified target value. Due to the fact that Tanks A and B are constantly losing fuel, researchers must focus on teaching participants how to fill these tanks.

Tank A can be filled using three methods. First, fuel can be pumped through Tank C via Pump 1. However, Tank C has a limited capacity and must be filled via Pump 5, which pumps fuel from the supply tank to right of Tank C. Second, fuel can be pumped into Tank A directly from the unlimited tank via Pump 2. Finally, fuel can be pumped into Tank A from Tank B via Pump 8. This is a viable solution in the event that Tank B is over-filled while Tank A is under-filled or if Pumps 1, 2, or 5 are disabled for extended periods. However, it should be noted that this solution is only recommended as a last resort, because in most cases it may negatively affect one's Resource Management score.

Tank B can be filled using three methods. First, fuel can be pumped through Tank D via Pump 3. However, Tank D has a limited capacity and must be filled via Pump 6, which pumps fuel from the supply tank to right of Tank D. Second, fuel can be pumped into Tank B directly from the unlimited tank via Pump 4. Finally, fuel can be pumped into Tank B from Tank A via Pump 7.

This is a viable solution in the event that Tank A is over-filled while Tank B is under-filled or if Pumps 3, 4, or 6 are disabled for extended periods. However, as was previously the case, it this solution is only recommended as a last resort.

In this task, the pumps can be toggled on and off in two ways. They can be activated/deactivated by pressing the corresponding number on the keyboard (i.e., *I* for Pump 1), or by mouse-clicking the small square between two tanks. Squares that are black indicate that a given pump is "off," while squares that are green indicate that a given pump is "on."

Additionally, there are two types of malfunctions that can occur in the Resource Management subtask. The first type of fault is known as a Shut-off. With a Shut-off, a specific pump will turn off automatically with no notification that it has turned off. In essence, a Shut-off simulates a participant manually turning the pump off.

The second type of fault is a Pump Failure. With a Pump Failure, the affected pump turns red, indicating that it is disabled and that fuel is cannot flow through it. Participants will not be able to use that pump for a specified duration, unless the task is configured to allow the participant to repair that pump. If the subtask is configured to accept repair commands, or if the pump repairs itself after the specified duration, upon repair, the pump will turn black and will resume normal function. Please note that depending on how the subtask's flow rates are configured, the participant will only be able to successfully manage the tank volumes with a limited number of Pump Failures for any length of time.



Figure 014 – The Resource Management and Pump Status windows illustrating a Pump 1 failure.

For information on enabling the subtask to accept repair commands, see section **8.2.2.4 Enable Manual Pump Repair?**. For more information on the repair commands themselves, see section **6.5.3.1.2 Timeout Commands**. Finally, for configuring the duration of Pump Failures, as well as other parameters associated with the basic behavior of the Resource Management subtask when in Manual Mode, please refer to section **8.2.6 Resource Management Subtask Basic Parameters Group**.

#### **3.2.2.** Automated Mode

As in the previous release of AF-MATB, when Automated Mode is engaged, the participant's ability to manipulate the pumps is disabled. Both automation algorithms are designed to govern the volume of Tanks A and B, keeping them within the target range specified by the researcher, though the means by which the two algorithms do this and the failures that occur under automation vary under each algorithm.

For detailed information on configuring which algorithm is used by the Resource Management subtask's Automated Modes, please see section **8.2.2.12 Resource Management Automation Algorithm**. For more information on configuring specific parameters associated with Automated Mode behaviors, please see section **8.2.9 Resource Management Subtask Automation Parameters**.

### 3.2.2.1. Automation Algorithm 1

In this mode, the automation can take on two different appearances. First, the Resource Management window can look as if no Automated Mode has been enabled, giving no overt indication of any additional functionality. When configured in this manner, the Resource Management subtask appears just as it does in Figure 012, except that, as previously stated, the participant's control over the task has been disabled.

However, if the researcher prefers to have a more obvious indication that Automated Mode has been enabled, while also reducing the visual distraction associated with the automation activating and deactivating pumps, researchers can elect to enable the option illustrated in Figure 015. In this configuration, the pump states are masked in the Resource Management and Pump Status windows.



Figure 015 – The Resource Management and Pump Status windows illustrating a situation where the researcher has elected to mask the Resource Management's behavior while automated.

Malfunctions of this automated mode are visually and functionally identical to the Pump Failures discussed in section **3.2.1 Manual Mode**, and illustrated in Figure 014 when the subtask's behavior is not masked. For an example of how Pump Failures appear when the subtask's

behavior is masked, see Figure 016. Please note that the automation is only capable of coping with a limited number of Pump Failures, just as is the case for participants who are manually managing the tank volumes.



Figure 016 – The Resource Management and Pump Status windows illustrating a situation where the researcher has elected to mask the Resource Management's behavior while automated and a Pump Failure has occurred.

#### **3.2.2.2.** Automation Algorithm 2

The main goal of this algorithm was to allow the researcher to manipulate the reliability of the automation, just as was the case in the System Monitoring subtask. When engaged (see Figure 017), this algorithm is also able to maintain the volumes of Tanks A and B within the researcherdefined automation ranges, but in this case, the automation may experience failures in managing the volumes of either tank. For example, the automation may incorrectly indicate to the participant that some pumps are inactive (see Figure 018), or the automation may fail to activate certain pumps, despite low tank volumes (see Figure 019). As such, participants must not only monitor the states of the pumps, but also the volumes of Tanks A and B to determine whether the automation has experienced a failure.



Figure 017 – The Resource Management and Pump Status windows illustrating Automation Algorithm 2.



Figure 018 – An illustration of an Algorithm 2 automation failure for Tank A. Note that, despite the pumps indicating an "off" state, Tank A was overfilled.



Figure 019 – An illustration of an Algorithm 2 automation failure for Tank B. Note that, despite the fact that the pumps should be on, as required by the low volume, they remain off.

Once that determination has been made, Tank A's volume will travel in the appropriate direction, and once it has crossed the appropriate threshold, the clock will start on the amount of time the participant has to respond to the fault. This allows the duration of automation failures to be somewhat independent of the refresh rate of the task. See sections **8.2.9.1Tank Volume Update Interval (Seconds)** and **8.2.9.10 Automation Fault Duration (Seconds)** for more information.

It should be noted that, in the moments before the software triggers a scheduled failure, the volume of fuel in Tanks A and B is used as a factor in determining the direction of the failure. For example, if the volume in Tank A is closer to the upper end of the accepted range, the software will trigger a failure that causes the tank to overfill. Likewise, if the tank's volume is closer to the lower end of the accepted range, the failure will cause the tank volume to fall below the accepted range. Once the volume has crossed outside of the target range, the failure will persist until the user either resets the automation or the failure has timed-out.



Figure 020 – An example of how an automation failure appears to the user under Automation Algorithm 2.

The change of volume that occurs in a given tank during a failure will exhibit a roughly parabolic curve. The properties of the curve, as well as when the rebound occurs, are determined by the duration of the automation failure and the refresh rate of the Resource Management subtask.

For example, Figure 020 details an automation failure that was scheduled while Tank A was at 2481 liters. As the failure occurred when the volume of Tank A was closest to the lower limit, this triggered a loss state. However, because of the refresh rate of the task and rate at which Tank A was losing fuel, the problem did not actually manifest until four seconds later, when the volume of the tank displayed 2281 liters. From this point, the participant has a researcher-specified ten seconds to identify and correct the failure. After six seconds, Tank A's volume begins to rebound, and at ten seconds, the volume of Tank A is back above 2350 liters, the minimum automation operating range.

Participants that recognize this automation failure must simply reset the automation via the *Reset* button or associated command (see section **6.2 Resource Management Response Commands** for more information), forcing the algorithm to update its information and restore normal function. It should be noted though that regardless of whether a participant does or does not identify an automation failure, the volume of Tank A will continue to follow the same curve outline in Figure 020.

Any reset command by the participant, either a correct identification or false alarm, will temporarily turn the *Reset* button green, confirming to the participant that the command was received. A correct identification of an automation failure is characterized by a change in color of the Tank A and Tank B level indicators from green to red (see Figure 021). These indicators

will remain red until the automation has successfully returned Tank A and Tank B's volumes back to the researcher-defined automation range.



Figure 021 – The Resource Management algorithm successfully reset by a participant. Note the change in color on the *Reset* button and Tank volume indicators.

#### 3.3. Tracking

Located in the middle upper portion of the Task Window, is the Tracking subtask. This subtask is the only subtask that does not have dual input. Using a joystick, the participant's goal is to take the green, circular reticle and steer it as close to the center yellow crosshair as possible. The Tracking subtask has 3 levels of difficulty, Low Difficulty, Moderate Difficulty, and High Difficulty, each of which can be triggered via a keyboard command (see section **6.5.4 Tracking**) or through a script (see section **9.2.2.4 Tracking Subtask**). Each level of difficulty is marked by more frequent changes in direction and faster movement of the reticle. For more information on configuring the levels of difficulty in this subtask, please see section **8.2.8 Tracking Subtask Difficulty Parameters Group**.



Figure 022 - The Tracking subtask with a small box outline and 3 horizontal crosshairs spaced 35 pixels apart.

18

Distribution A: Approved for public release; distribution unlimited. 88 ABW Cleared 12/03/2014; 88ABW-2014-5667.

This subtask also includes an Automated Mode. When this mode is enabled the reticle will move towards the center crosshair and maintain that position until the autopilot has been disengaged. During this time, no input from the joystick is required. This mode can also be activated via a keyboard command or through a script.

One new feature to this subtask is that now both the number and spacing of the horizontal crosshairs in the task window are now configurable. As illustrated in Figure 022, this configuration has 3 horizontal crosshairs spaced approximately 35 pixels apart, while demonstrates a configuration with 5 horizontal crosshairs spaced 50 pixels apart.

Another new appearance-related feature that has been updated is the center outline shape and size. Originally, this task supported only a single outline shape. The new version, however, supports a number of different size boxes (see Figure 022 and Figure 023) and circles (see Figure 024 and Figure 025) that can be customized to researcher specifications.



Figure 023 – The Tracking subtask with a large box outline and 5 horizontal crosshairs spaced approximately 50 pixels apart.



**Figure 024** – The Tracking subtask with a large circle outline.



Figure 025 – The Tracking subtask with a small circle outline.

Finally, it should be noted that in previous versions of the task, the outline served no functional purpose. However, this outline now serves as a scoring boundary we use for generating some new metrics implemented we've implemented. Please see section **7.3.3.1.2 Tracking Section** for more information on these metrics and how they're calculated.

For more information on the parameters associated with governing the general behavior of the Tracking subtask, please see section **8.2.4 Tracking Subtask Basic Parameters Group**.

#### **3.4.** Communications

The Communications subtask is an aural task located in the lower left portion of the Task Window. During this task, audio transmissions are issued to specific callsigns with instructions to change the radio channel and frequency. The two types of Communications Events (CEs) in this task are True Communications events (TCs) and False Communication events (FCs). TCs are addressed to the participant's designated callsign, which is identified by the blue "Callsign" label in the Communications window (see Figure 026). As illustrated in Figure 026, the participant's callsign is designated as "NGT504." FCs are addressed to any callsign other than the one designated in the Communications window.



Figure 026 – The Communications subtask with the default maximum and minimum frequencies shown for the NAV channels and default minimum and maximum frequencies for the COM channels.

Participants are allotted a designated amount of time to respond to TCs, and a lack of response during that timeframe will count as a timeout. In order to successfully respond to a TC, the participant must follow the instructions issued and change the specified channel to the correct frequency before the communication transmission times-out. The four channels in the Communications window are the column on the left, labeled as "NAV1," "NAV2," "COM1," and "COM2" by default. The frequencies of each of those corresponding channels are located to the right of each channel.

An example of a TC is as follows: "NGT504, NGT504, set first communication to one two one point five." "First communication" refers to the channel "COM1," and "one two one point five" refers to the frequency, 121.5. In order to respond properly, the arrow indicators need to be set to the correct channel, in this case, "COM1." Depending on the input method selected specified by the researcher, the participant can use the corresponding arrow keys on the keyboard, or the participant can click on the corresponding arrow in the Task Window. Next, the correct frequency should be selected using the arrow keys on the keyboard or Task Window, again depending on how the response options are configured by the researcher. Finally, while the arrows are still on the appropriate channel, the participant needs to lock-in that frequency either by pressing the *Enter* key on the keyboard or clicking the *Enter* button on the Task Window with the mouse. Once a participant locks-in a frequency, the *Enter* button will briefly change from blue to green, allowing the participant to verify that his/her response was successfully

recorded (see Figure 027). Please note that the arrows indicate the information that will be locked-in when *Enter* is pressed.



Figure 027 – The Communications subtask where someone just locked in frequency 121.5 on channel COM1, as indexed by the green *Enter* button.

For detailed information on modifying the parameters associated with governing the behavior of the Communications subtask, please see section **8.2.5 Communications Subtask Parameters Group**.

### 3.5. Scheduling

The Scheduling window is located in the upper right portion of the Task Window. This window allows the participant to see eight minutes into the future for the Tracking and Communications subtasks. Ideally, this information can be used by participants to anticipate future task demands in order to improve performance. When disabled, or when a script is not loaded, the Scheduling window will not display information regarding upcoming events; the Scheduling window will simply appear with two thin yellow lines, as illustrated in Figure 028. This window only provides information and does not require any action by the participant.

For detailed information on modifying the parameters associated with governing the behavior of the Scheduling window, please see section **8.2.2.6 Enable Scheduling**?



Figure 028 – The default appearance of the Scheduling window with no events scheduled.

The line on the left represents the pending difficulty level(s) for the Tracking subtask, and the line on the right represents pending communication transmissions in the Communications subtask. For the Tracking subtask, events appear as blocks of color (green, red, yellow, or blue). For the Communications subtask, upcoming transmissions are represented as small red rectangles, regardless of whether they were TCs or FCs.

The advancement of time in the Scheduling Window is indicated by the progression of the event indicators toward the top of the window. Once an event indicator reaches the top, it will disappear.

In the first example (see Figure 029), the thick, green line on the left indicates that the Tracking subtask will operate at Low Difficulty for the next five minutes. The red rectangles on the right indicate that eight communication transmissions are scheduled to take place within the next five minutes.



Figure 029 – A Scheduling window showing a 5-minute condition with the Tracking subtask operating at Low Difficulty and 8 CEs.

The second example (see Figure 030) illustrates a 5-minute condition with the Tracking subtask operating at Moderate Difficulty (represented by the thick, yellow line) and 16 transmissions.



Figure 030 – A Scheduling window showing a 5-minute condition with the Tracking subtask operating at Moderate Difficulty and 16 CEs.

24 Distribution A: Approved for public release; distribution unlimited. 88 ABW Cleared 12/03/2014; 88ABW-2014-5667. The third example (see Figure 031) illustrates a 5-minute condition with the Tracking subtask operating at a High Difficulty (represented by the thick, red line) and 22 transmissions. Please note that the thicker red rectangles in the Communications timeline indicate that some transmissions will be received almost consecutively.



Figure 031 – A Scheduling window showing a 5-minute condition with the Tracking subtask operating at High Difficulty and 22 CEs.

In the last basic example (see Figure 032), the Tracking subtask is operating in Automate Mode, as indicated by the thin, blue line on the Tracking timeline.



Figure 032 – A Scheduling window showing a 5-minute condition with the Tracking subtask operating in Automated Mode and 8 CEs.

25 Distribution A: Approved for public release; distribution unlimited. 88 ABW Cleared 12/03/2014; 88ABW-2014-5667. In addition to being able to display individual conditions, the Scheduling window can also display scripts that contain multiple conditions. For example, in Figure 033, a six minute script was loaded, composed of three 2-minute conditions. In the first two minutes, the Tracking subtask is operating at Low Difficulty, in the second 2 minutes, (minutes 2:00 - 4:00), the subtask operated in Automated Mode, and finally, in minutes 4:00 - 6:00, the subtask operated at High Difficulty. In the second example (see Figure 034) a script was loaded that contained two conditions, each 5 minutes long. This example shows the Scheduling window at the start of the trial, and then 2 minutes later.



**Figure 033** – A Scheduling window showing three 2-minute conditions comprising one trial. The Tracking subtask is operating at Low Difficulty, in Automated Mode, and then at High Difficulty, with 8, 5, 8 CEs for three conditions, respectively.



Figure 034 – A Scheduling window showing the start of a trial (Figure 034a) and the same trial 2 minutes later (Figure 034b).

26 Distribution A: Approved for public release; distribution unlimited. 88 ABW Cleared 12/03/2014; 88ABW-2014-5667.

#### 4.0 **ADDITIONAL AF-MATB FEATURES**

This section contains information about specific additions to the AF-MATB not necessarily related to any particular subtask. These features were implemented as a result of requests by our user base over the last several years.

#### 4.1. **Information Throughput Mode**

While the individual performance measures generated by the AF-MATB software can provide researchers with a clear picture of the performance of a participant, a combined performance metric can be very useful for those looking to evaluate human performance and multitasking strategy. As a result, a specific Information Throughput (IT) Mode was designed to customize the task to fit the Human-MATB Interaction component of the quantitative Human Operator Informatic Model (HOIM) (Phillips, Repperger, Kinsler, Bharwani, & Kender, 2007; Phillips et al., 2013; Phillips & Walters, 2013; Phillips, Walters, & McKinley, 2013; Walters, 2012). This customization allows for the computation of a single, uniform metric that describes overall system complexity.

In order for the AF-MATB to fit the model, a number of modifications to the task were required. First, Automated Modes for the System Monitoring, Resource Management, and Tracking subtasks have been disabled through both manual commands and via script instructions. In addition, the HOIM frames the Tracking subtask as a "time-on-target" task, which requires that the subtask be configured to use a circle for the Tracking subtask's outline. As such, when IT Mode is enabled, the ability to configure the Tracking subtask's outline is disabled, as well as the ability to configure the size of the outline. Next, the HOIM also frames the Resource Management subtask as a "time-on-target" task, meaning this subtask must operate under a special configuration. Specifically, when IT Mode is enabled, all but two pumps (Pumps 2 and 4) will be rendered as Pump Failures. Additionally, this configuration also disables the ability to repair Pump Failures. Finally, because of the revised nature of the subtask, the acceptable fuel range indicators discussed in section 3.2 Resource Management and illustrated in Figure 013 have been enhanced with the addition of red lines that run across each tank to help the illustrate the target to the participant. Please refer to Figure 035 for the appearance of AF-MATB when IT Mode is enabled.

For detailed information on configuring this mode, please see section 8.2.2.14 Enable **Information Throughput Mode?.** 



Figure 035 – The appearance of the AF-MATB when IT Mode is enabled.

### 4.2. Subtask Component Visibility

This capability allows the researcher to hide any one of the six AF-MATB windows (see Figure 036). Researchers can hide AF-MATB windows by one of two ways; either via Config File or Script File. Using Config Files, researchers can initially load the task so that any window can be hidden. Using Script Files, researchers can use multiple conditions to dynamically change the task by hiding one or more windows over time, something that might greatly aid those interested in workload transitions. For more information on using this feature in the Configuration Utility, please see section **8.2.10 Subtask Visibility Parameters**. For more information on using this feature in the Script Generator Utility, please see section **9.2.2.7 Subtask Component Visibility**.

A.	AF_MATB_V300	- 🗆 🗙
		۲ ۲
SYSTEM MONITORING	TRACKING	SCHEDULING
00.00.00 08/17/2013		
COMMINICATIONS	RESOURCE MANAGEMENT	סוואס פייאייני
COMMONICATIONS		FOME STATOS

Figure 036 – AF-MATB, rendered with all windows hidden.

### 4.3. NASA-TLX Integration

This version of the AF-MATB software now includes an electronic version of the NASA-Task Load Index (NASA-TLX), an inventory designed to provide a subjective measure of workload assessment (Hart & Staveland, 1988).

When scheduled, the task will move to a "paused" state and the TLX will appear (see Figure 037). Once the inventory is launched, participants will first need to complete Part 1 of the inventory (see Figure 038) by clicking on the continuum of each dimension to report their experience. Additionally, researchers have the option to instruct participants to complete the Sources of Workload section (Part 2, see Figure 039) or skip over it by selecting the appropriate radio button at the bottom of Part 1. Once the inventory has been completed, a countdown timer (see Figure 053) will appear, allowing participants to prepare for the task to resume.

For more information on configuring the duration of the timer for this feature, please see section **8.2.2.11 NASA-TLX Ramp-Up Time Delay (Seconds)**. For information on scheduling scripts to launch the NASA-TLX during trials, see section **9.2.5 Schedule Custom Events**.



Figure 037 – The NASA-TLX launched during a trial.



Figure 038 – Part 1 of the NASA-TLX.



Figure 039 - One workload pairing in the Sources of Workload (SoW), Part 2 of the NASA-TLX

#### 4.4. Subtask Automation Indicators

One of the issues with the implementation of automation in previous versions of MATB and AF-MATB was the lack of a consistent indication when one or more subtasks were in Automated Mode. As result, a simple customization is now available to convey the operating state of each subtask to the participant via the subtask's banner color, allowing the researcher to select one of five colors (red, green, blue, yellow, and orange) when a subtask is in Automated Mode.



**Figure 040** – The AF-MATB with the System Monitoring, Tracking, and Resource Management subtasks operating in Automated Mode. Note the color of the name banners for these three subtasks has changed from the normal blue color to green.

For detailed information on the parameter that governs this feature, please see section **8.2.2.13 Enable Subtask Automation Indicators?**.

### 4.5. Serial and Digital Port-Triggering

To assist researchers in coordinating AF-MATB data with psychophysiological or neurobehavioral data, which requires precision, the new version of the software now supports the ability to send serial and/or digital event triggers to acquisition software. For more details on the configuration options available to researchers, see section **8.2.11 Port Triggering Parameters**.

This feature supports serial triggers using a COM port, as well as digital triggers using a National Instruments' NI USB-6008 DAQ. For researchers who will use digital triggering, there are some additional software requirements. First, for best results, researchers should use the version of AF-MATB corresponding to the OS of the task computer (either x86 or x64). Next, researchers must follow the setup guide provided by National Instruments, installing the NIDAQmx driver software and National Instruments' Measurement & Automation Explorer (MAX) software. Once installed, open MAX and plug in a DAQ into the task computer. Confirm that the appropriate device is listed under the "Devices and Interfaces" branch of the "My System" configuration tree (see Figure 041). Researchers should note the nickname provided to the device by MAX, such as "Dev1," "Dev2," etc. This is critical, as the AF-MATB software requires the researcher to specify the appropriate device nickname to properly interface with the hardware (see section 8.2.11.7 Digital Device Nickname). Devices may be renamed by right-clicking on the device and clicking rename (see Figure 042). Finally, researchers must be aware that they must create a harness by wiring and/or soldering to connect the DAQ to the acquisition system. This requires sufficient knowledge of either electrical and/or biomedical engineering, so please take this into consideration when electing to use digital triggering.

Once port-triggering has been configured and loaded into the task, the task will attempt to initialize communication with either the digital or serial device. In the event that the task is not able to successfully initialize communication, the task will display a series of error codes. For information on translating these error codes, please refer to the error documentation provided with the NIDAQ drivers and MAX software. This documentation is typically located in the file C:\Program Files\National Instruments\Shared\Errors\English\MeasurementsUV-errors.txt.

Please understand that regardless of whether serial or digital triggering is enabled, each setup varies based on the operating system, drivers, hardware, and other software used. As such, we cannot explicitly support any one setup, and cannot provide technical assistance in the event that users are unable to establish communication between the task and any acquisition software. Instead, this feature was included solely as a convenience for those researchers who feel that this feature would greatly add to their research and have the time to invest in building a harness and troubleshooting the setup.



Figure 041 - National Instruments' Measurement & Automation Explorer illustrating the connection of a NI-USB 6008, nicknamed "Dev1."



Figure 042 – An example of how to rename DAQs to match previously defined AF-MATB parameters.

#### 5.0 BEFORE USING AF-MATB

In order to run AF-MATB using the executable, an acceptable MATLAB Compiler Runtime (MCR; see Figure 043) needs to be installed. The version of the installer included in the AF-MATB package is MCR 7.13. Please be aware, your installation may vary slightly depending on your operating system and version of the task (x86 or x64).

To install the MCR, run the included InstallShield Wizard (see Figure 043).



Figure 043 – The MCR Installer icon on a black desktop background.

Using InstallShield Wizard, users will select their language preference and then install the Visual C++ Runtime Components, called VCREDIST\_X86.exe.

InstallShield Wizard				
MATLAB(R) Compiler Runtime 7.13 requires that the following requirements be installed on your computer prior to installing this application. Click OK to begin installing these requirements:				
Status Requirement				
Pending VCREDIST_X86				
😵 Install Cancel				

**Figure 044** – The Visual C++ Component Runtime installation window.

Next, these runtime components (called VCREDIST\_X86.exe) will extract and install. After this, the MathWorks Installer screen will appear (see Figure 044).

MATLAB(R) Compiler Runtime 7.13 - InstallShield Wizard				
MATLAB Compiler Runtime	The InstallShield(R) Wizard will install MATLAB(R) Compiler Runtime 7.13 on your computer. To continue, dick Next. MATLAB and Simulink are registered trademarks of The MathWorks, Inc. Please see www.mathworks.com/trademarks for a list of other trademarks owned by The MathWorks, Inc. Other product or brand names are trademarks or registered tradmarks of their respective owners. WARNING: This program is protected by copyright law and international treaties. Copyright 1984-2010, The MathWorks, Inc.			
A The MathWorks				
	< Back Next > Cancel			

Figure 045 – The Mathworks Installer screen for the MCR

Users should click "Next" and select their Name and Organization. Finally, they will choose a location for the installation files, and then click "Install" (see Figure 045).

MATLAB(R) Compiler F	Runtime 7.13 - InstallShield Wizard
Ready to Install the Program The wizard is ready to begin installation	n.
Click Install to begin the installation. If you want to review or change any o exit the wizard.	of your installation settings, click Back. Click Cancel to
InstallShield	< Back Install Cancel

Figure 046 – Click "Install" to install the MCR so that the 3 included .exe files can be used.

Once the installation is complete, users will be able to run all three of the included executables that are part of the AF-MATB Package (see Figure 046).

After installing the MCR, the AF-MATB Executables are ready to use. Task executables should reside within the same directory as the AF-MATB System Files directory (Figure 047). If this is not the case, the user will still be able to select the location of this directory prior to entering the participant's ID and script information. Please note that the System Files folder should not be modified in any way in order to preserve proper functioning of the task.



Figure 047 – The three executables and the AF-MATB System Files folder in a directory. The installation is now complete.

## 6.0 TASK COMMANDS

As in the previous version of AF-MATB, this version also supports participant response via the keyboard, clickable buttons, or both, depending on the researcher's configuration of the task. This section and the following figure sets (

Figure 048, Figure 049, and Figure 051) serve to illustrate the keys and/or buttons on the task that the participant can press in order to respond to or control various aspects of the AF-MATB.

## 6.1. System Monitoring Commands

It should be noted that any key or button pressed for any gauge or light not malfunctioning will result in a false alarm. Additionally, participants should be aware that the F5 and F6 commands are only available when the System Monitoring subtask is in Manual Mode.

### F1

In the event that a malfunction occurs in the first gauge, pressing the F1 key or button will correct the malfunction.



# F2

In the event that a malfunction occurs in the second gauge, pressing the F2 key or button will correct the malfunction.



# **F3**

In the event that a malfunction occurs in the third gauge, pressing the F3 key or button will correct the malfunction.



F4

In the event that a malfunction occurs in the fourth gauge, pressing the F4 key or button will correct the malfunction.



# F5

In the event a malfunction occurs in the first light, pressing the F5 key or button will correct the malfunction.



#### **F6**

In the event a malfunction occurs in the second light, pressing the F6 key or button will correct the malfunction.



Figure 048 – The six buttons in the task window that the participant can clicked for the System Monitoring subtask.

### 6.2. Resource Management Response Commands

All pumps in the Resource Management subtask can be toggled on or off using either the associated keyboard command or by clicking on the pump with the mouse. However, pumps that are experiencing failures (red pumps) cannot be activated until the failure has been repaired, either by the participant or by the system. Additionally, pumps will not activate when the participant attempts to pump fuel away from an empty tank. This information applies to all pumps in the Resource Management subtask.

1

The direction of flow from this pump, as denoted by the arrow next to that pump, indicates that fuel is flowing from Tank C to Tank A at the rate indicated by the Pump Status Window.



2

The direction of flow from this pump is from an unlabeled supply tank to Tank A at the rate indicated by the Pump Status Window.



3

The direction of flow from this pump is from Tank D to Tank B at the rate indicated by the Pump Status Window.



4

The direction of flow from this is from an unlabeled supply tank to Tank B at the rate indicated by the Pump Status Window.



#### 5

The direction of flow from this pump is from an unlabeled supply tank to Tank C at the rate indicated by the Pump Status Window.



#### 6

The direction of flow from this pump is from an unlabeled tank to Tank D at the rate indicated by the Pump Status Window.



#### 7

The direction of flow from this pump is from Tank A to Tank C at the rate indicated by the Pump Status Window.



### 8

The direction of flow from this pump is flowing from Tank B to Tank A at the rate indicated by the Pump Status Window.



#### Figure 049 – The eight buttons in the task window that can be click for the Resource Management subtask.

# 0

The command used for resetting the Resource Management automation when operating as discussed in section **3.2.2.2 Automation Algorithm 2**. Participants can execute this command by pressing the *0* key in the number row, the *Reset* button (see Figure 050), or the *Joy3* button on the joystick.



Figure 050 – The *Reset* button of the Resource Management subtask.

#### 6.3. **Communications Response Commands**

The traditional method of response for the Communications task required participants to use the arrow keys to select the correct radio channel and frequency. However, given that this can be a time-intensive process, an alternative to this method was developed, allowing participants to respond by typing the correct frequency using the number pad, which should allow for more streamlined responses and more authentic response times. For information on configuring the entry method used for this subtask, please refer to section 8.2.5.9 Frequency Entry Method.

#### **Frequency Entry Method – Arrow Keys** 6.3.1.

#### $\rightarrow$ (Right Arrow)

This key will increase the frequency of the selected channel by the defined increment. If the maximum frequency for that channel is reached, then the frequency will wrap around and start over from the minimum frequency.



#### $\leftarrow$ (Left Arrow)

This key will decrease the frequency of the selected channel by the defined increment. If the minimum frequency for that channel is reached, then the frequency will wrap around and start over from the maximum frequency.



### ↑ (Up Arrow)

This key allows participants to select a channel. Pressing this key will move the arrows towards the top communication channel. If the first communication channel is selected and this arrow is pressed, the arrows will wrap around to the bottom and the fourth channel will then be selected.



# (Down Arrow)

This key allows participants to select a channel. Pressing this key will move the arrows towards the bottom communication channel. If the bottom communication channel is selected and this key is pressed, the arrows will wrap around to the top and the first channel will then be selected.



Enter

The *Enter* key locks in the frequency of the currently selected channel. Whichever channel is currently selected, as indicated by the location of the arrows, will be the channel and frequency that are recorded. The *Enter* button in the task will turn green for a defined duration, confirming that the frequency has been locked-in.



Figure 051 – The five objects in the task window that can be selected to perform actions for the Communications subtask.

## 6.3.2. Frequency Entry Method – Number Pad

COMMUNICATIONS					
Callsign NGT504					
	NAV1	000.0			
	NAV2	000.0			
	COM1	000.0			
	сом2	000.0			
	Ent	er			

Figure 052 – The appearance of the Communications subtask under the "Number Pad" entry method.

The number pad entry method exhibits the same wrap-around and movement behaviors found when using the *Up Arrow*, *Down Arrow*, and *Enter* keys in the previous entry method. The primary difference to this entry method is that the *Left Arrow* and *Right Arrow* keys are disabled here. Instead, participants input the frequency using the number pad on the keyboard.

Participants may respond to events with or without the use of the decimal. For example, if a TC event instructed participants to enter "109.7" at "NAV1", the participant just needs to key in those four numbers on the number pad. In the event that the participant makes a mistake, they do not need to delete the entry, but rather just retype the numbers until the correct frequency is populated.

When the participant is ready to lock-in their frequency, they can press either the *Enter* key or GUI button, resetting that field to "000.0." Please note that while this entry method supports use of the mouse to click on the *Enter* button in the task window, there are no available mouse commands for selecting a specific channel or manipulating frequencies.

## 6.4. System Commands

This section details the basic commands required to start and stop the task, as well as a few other features designed to make task configuration testing, participant practice, and demonstrations easier. For detailed information on configuring the functionality of the *Esc*, *Pause / Break*, and *Home* commands, please see sections 8.2.2.7 Enable Escape Key?, 8.2.2.2 Enable Pausing?, and 8.2.2.8 Enable Home Key?, respectively.

### Space Bar

Pressing the *Space Bar* will trigger a countdown to start the task. The command works only when the task is in a "ready" state.



Figure 053 – The AF-MATB window immediately after pressing Space, demonstrating the timer.

The above figure (Figure 053) illustrates the countdown timer after pressing the *Space Bar*. When the timer expires, it will disappear and the task will begin normal functioning. In order to provide the researcher with more control over the task, the duration of this time is subject to the researcher's specification (please see section **8.2.2.10 Initial Ramp-up Time Delay (Seconds**)).
#### Esc (Escape)

If this command is enabled, pressing *Esc* on the keyboard once will stop the task. Pressing *Esc* After *Esc* is pressed, the task is in a "suspended" state, as indicated by the word "Stopped" located beneath the gauges (see Figure 054. This state is characterized by a suspension of all commands other than *Esc* and *Home* (detailed below). Once stopped, the current trial/experiment cannot be resumed. Please note that that the data and timer must be visible for "Stopped" to be displayed. Figure 055 illustrates the absence of the date and timer, as well as other indicators like "Paused" or "Stopped."







Figure 055 – The space below the System Monitoring subtask, demonstrating what the window looks like if the timer and date are disabled and the task is in a "suspended" state.

Pressing *Esc* a second time will reset the entire task and restore it to a "ready" state. At this point, *Space Bar* functionality is restored, and the participant can start the task. When in the

"ready" state, the timer at the bottom of the System Monitoring window will return to "00:00:00" in yellow if the date and timer were previously displayed.

Please note that all performance data generated in the interrupted trial will be erased once the task has transitioned back to a "ready" state. Therefore, in the event that these data are desired, it should be copied before the task is returned to a "ready" state.



Figure 056 - The AF-MATB window after Esc was pressed again, placing the task back into a "ready" state.

#### Pause | Break

If this command is enabled, pressing the *Pause* key once will suspend the task and change the color of the timer, if visible, to green. Pressing *Pause* again will instantly resume the task, restoring the timer to its normal, yellow color. Using this command will not transition the task out of a "ready" state, as is the case with *Esc*.



Figure 057 – The task when paused.

#### Home

If enabled, pressing the *Home* key at any point will prompt a file selection screen and allow the user to load a new configuration or script file (see Figure 058). Loading either file will clear all previously generated performance logs and generate new logs as specified by the new file. As such, the researcher should be aware that this feature is meant only for demonstration purposes or to test the development of Config or Script Files, and should not meant to be used if any performance data is desired across multiple trials.



Figure 058 – The task window with file selection window after pressing Home.

Researchers do not need to specify the type of file they are loading, as the task is able to determine the file type. For examples of the messages generated when either file type is loaded into the task, see Figure 059 and Figure 060. Once the participant or researcher acknowledges the confirmation message, the task will transition to the "ready" state.

If the loading process is interrupted or if the loading screen is closed, the user will be notified (see Figure 061). When this occurs, the task will remain in a "suspended" state until the user presses *Esc* to restore the task to a "ready" state. Finally, if a researcher attempts to load a corrupt or damaged Config File (see Figure 062) or Script File (see Figure 063, Figure 064, & Figure 065), the user will be notified. For more information on loading behavior, see section **7.2.1 Phase 1: Loading Config and Script Files**.



Figure 059 – AF-MATB determined new parameters were loaded using the *Home* key, displaying the appropriate message.



Figure 060 – AF-MATB determined a new script was loaded using the *Home* key, displaying the appropriate message.



Figure 061 – Failure to load a file.



Figure 062 – Notification message indicating that the user loaded a Config File that was corrupt or partially damaged.



Figure 063 – Notification message indicating that the user loaded a Script File that was missing one or more parameters.



Figure 064 - Notification message indicating that all parameters from a file were loaded in, but that the script in this file, if present, may be damaged.



Figure 065 – Notification message indicating that a Script File contained both missing or damaged parameters as well as missing or damaged script information.

# 6.5. Additional Keyboard Commands

The commands in this section were designed to provide the researcher or participant with the ability to demonstrate task behaviors instantly, without the use of a script. This functionality is particularly beneficial in training situations, where interactive demonstrations can help to decrease the time that participants understand the task.

The following sections outline all of the commands available to participants and/or researchers as they relate to the onset and/or offset of task behaviors. There are several things that the user should be aware of regarding this section. First, it is not advisable to use any commands in this section (except for those in section **6.5.1 Transition Logging**) when collecting performance data, as these commands can affect performance. Next, please be aware that unless otherwise noted, the functionality in this section is subject to the task configuration. Additionally, task behaviors triggered by these commands, such as a Gauge 1 malfunction, triggered by these commands do not have an associated timeout, unless otherwise noted,. Thus, Gauge 1 will continue to exhibit a malfunction until a timeout or repair signal is received from either the participant or the script. Finally, unless otherwise noted, all commands will be logged in the Master Event Log.

## 6.5.1. Transition Logging

# Q

This keyboard command allows participants to report when they perceive that a trial has transitioned from one condition to another. If a script is loaded, the information from this command will be logged in the Transition Event Log (discussed in section **7.3.2.13 Transition Log**). If no script is loaded, this command will be logged in the Master Event Log only. This command is the only command in this section that is not subject to the task configuration, as it is always enabled.

#### 6.5.2. System Management 6.5.2.1. Manual Mode Commands 6.5.2.1.1. Malfunction Commands

#### Shift + F1

Pressing this key combination allows the participant or researcher to manually trigger a malfunction in Gauge 1.

## Shift + F2

Pressing this key combination allows the participant or researcher to manually trigger a malfunction in Gauge2.

## Shift + F3

Pressing this key combination allows the participant or researcher to manually trigger a malfunction in Gauge 3.

#### Shift + F4

Pressing this key combination allows the participant or researcher to manually trigger a malfunction in Gauge 4.

# Shift + F5

Pressing this key combination allows the participant or researcher to manually trigger a malfunction in Light 1 (Green Light).

# Shift + F6

Pressing this key combination allows the participant or researcher to manually trigger a malfunction in Light 2 (Red Light).

# 6.5.2.1.2. Timeout Commands

#### Shift + F7

Pressing this key combination allows the participant or researcher to simulate a timeout of a Gauge 1 malfunction. After pressing this key, Gauge 1 will resume normal functioning.

#### Shift + F8

Pressing this key combination allows the participant or researcher to simulate a timeout of a Gauge 2 malfunction. After pressing this key, Gauge 2 will resume normal functioning.

#### Shift + F9

Pressing this key combination allows the participant or researcher to simulate a timeout of a Gauge 3 malfunction. After pressing this key, Gauge 3 will resume normal functioning.

#### F10

Pressing this key allows the participant or researcher to simulate a timeout of a Gauge 4 malfunction. After pressing this key, Gauge 4 will resume normal functioning.

IMPORTANT: Due to the nature of the Windows Operating System, pressing F10 is similar to pressing Alt, and the focus on the window will move away from the actual AF-MATB GUI and be redirected towards the Menu Bar at the top of the screen. At this point, the task may appear unresponsive to other keyboard commands. Simply press F10, Alt, or click on the task window again to move focus back to AF-MATB.

#### *Shift* + *F11*

Pressing this key combination allows the participant or researcher to simulate a timeout of a Light 1 malfunction. After pressing this key, Light 1 will resume normal functioning.

#### *Shift* + *F12*

Pressing this key combination allows the participant or researcher to simulate a timeout of a Light 2 malfunction. After pressing this key, Light 2 will resume normal functioning.

#### 6.5.2.2. Automated Mode Commands

Due to the nature of the System Monitoring subtask when in Automated Mode, the malfunction commands described in this section cannot be executed simultaneously. As a result, researchers who have executed a malfunction command must execute the associated timeout command before issuing another malfunction command.

#### 6.5.2.2.1. Malfunction Commands

#### Shift + F1

Pressing this key combination allows the participant or researcher to manually trigger a Gauge 1 malfunction that will be corrected by the automation.

#### Shift + F2

Pressing this key combination allows the participant or researcher to manually trigger a Gauge 2 malfunction that will be corrected by the automation.

#### Shift + F3

Pressing this key combination allows the participant or researcher to manually trigger a Gauge 3 malfunction that will be corrected by the automation.

#### Shift + F4

Pressing this key combination allows the participant or researcher to manually trigger a Gauge 4 malfunction that will be corrected by the automation.

#### Shift + F5

Pressing this key combination allows the participant or researcher to manually trigger a Gauge 1 malfunction that the automation will fail to catch.

## Shift + F6

Pressing this key combination allows the participant or researcher to manually trigger a Gauge 2 malfunction that the automation will fail to catch.

## Shift + F7

Pressing this key combination allows the participant or researcher to manually trigger a Gauge 3 malfunction that the automation will fail to catch.

## Shift + F8

Pressing this key combination allows the participant or researcher to manually trigger a Gauge 4 malfunction that the automation will fail to catch.

# 6.5.2.2.2. Timeout Commands

#### Ctrl + 1

Pressing this key combination allows the participant or researcher to simulate the automation successfully returning Gauge 1 to normal operation.

## Ctrl + 2

Pressing this key combination allows the participant or researcher to simulate the automation successfully returning Gauge 2 to normal operation.

# Ctrl + 3

Pressing this key combination allows the participant or researcher to simulate the automation successfully returning Gauge 3 to normal operation.

#### Ctrl + 4

Pressing this key combination allows the participant or researcher to simulate the automation successfully returning Gauge 4 to normal operation.

#### Shift + F9

Pressing this key combination allows the participant or researcher to simulate a timeout of a Gauge 1 malfunction the automation failed to correct.

#### F10

Pressing this key combination allows the participant or researcher to simulate a timeout of a Gauge 2 malfunction the automation failed to correct.

IMPORTANT: Due to the nature of the Windows Operating System, pressing F10 is similar to pressing Alt, and the focus on the window will move away from the actual AF-MATB GUI and be redirected towards the Menu Bar at the top of the screen. At this point, the task may appear unresponsive to other keyboard commands. Simply press F10, Alt, or click on the task window again to move focus back to AF-MATB.

#### *Shift* + *F11*

Pressing this key combination allows the participant or researcher to simulate a timeout of a Gauge 3 malfunction the automation failed to correct.

# Shift + F12

Pressing this key combination allows the participant or researcher to simulate a timeout of a Gauge 4 malfunction the automation failed to correct.

## 6.5.2.3. Other Commands

# Shift + S

Pressing this key combination allows the participant or researcher to manually activate the System Monitoring subtasks Automated Mode.

## Alt + S

Pressing this key combination allows the participant or researcher to manually activate the System Monitoring subtasks Manual Mode.

# 6.5.3. Resource Management

## 6.5.3.1. Manual Mode and Automation Algorithm 1 Commands 6.5.3.1.1. Malfunction Commands

## Shift + 1

Pressing this key combination allows the participant or researcher to manually trigger a failure of Pump 1.

# *Shift* + 2

Pressing this key combination allows the participant or researcher to manually trigger a failure of Pump 2.

# Shift + 3

Pressing this key combination allows the participant or researcher to manually trigger a failure of Pump 3.

#### *Shift* + *4*

Pressing this key combination allows the participant or researcher to manually trigger a failure of Pump 4.

#### *Shift* + 5

Pressing this key combination allows the participant or researcher to manually trigger a failure of Pump 5.

#### Shift + 6

Pressing this key combination allows the participant or researcher to manually trigger a failure of Pump 6.

#### *Shift* + 7

Pressing this key combination allows the participant or researcher to manually trigger a failure of Pump 7.

## *Shift* + 8

Pressing this key combination allows the participant or researcher to manually trigger a failure of Pump 8.

## 6.5.3.1.2. Timeout Commands

The researcher can use these commands to simulate timeouts of Pump Failures or the participant can use these same commands to repair pumps if that functionality has been enabled. For more information on this feature, please see sections **3.2.1 Manual Mode** and **8.2.2.4 Enable Manual Pump Repair?**.

## Alt + 1

Pressing this key combination allows the participant or researcher to simulate a timeout of a Pump 1 Failure.

## Alt + 2

Pressing this key combination allows the participant or researcher to simulate a timeout of a Pump 2 Failure.

## Alt + 3

Pressing this key combination allows the participant or researcher to simulate a timeout of a Pump 3 Failure.

#### Alt + 4

Pressing this key combination allows the participant or researcher to simulate a timeout of a Pump 4 Failure.

#### Alt + 5

Pressing this key combination allows the participant or researcher to simulate a timeout of a Pump 5 Failure.

#### Alt + 6

Pressing this key combination allows the participant or researcher to simulate a timeout of a Pump 6 Failure.

# Alt + 7

Pressing this key combination allows the participant or researcher to simulate a timeout of a Pump 7 Failure.

#### Alt + 8

Pressing this key combination allows the participant or researcher to simulate a timeout of a Pump 8 Failure.

#### 6.5.3.2. Automation Algorithm 2 Commands

In Section **3.2.2.2 Automation Algorithm 2**, the nature of an automation failure when operating in this mode was discussed in depth. Failures in this mode will always last for the specified duration, whether or not they were successfully identified by the participant. As a result, keyboard commands in this section are unique in that there are no timeout commands. Additionally, malfunction commands in this section, like those outlined in section **6.5.2.2 Automated Mode Commands**, follow similar restrictions in that they cannot be issued simultaneously. The participant or researcher must wait to until the previous malfunction has timed-out before executing another malfunction command.

## Shift + 1

Pressing this key combination allows the participant or researcher to manually trigger an automation malfunction with Tank A.

# *Shift* + 2

Pressing this key combination allows the participant or researcher to manually trigger an automation malfunction with Tank A.

## *Shift* + *3*

Pressing this key combination allows the participant or researcher to manually trigger an automation malfunction with Tank A.

#### *Shift* + *4*

Pressing this key combination allows the participant or researcher to manually trigger an automation malfunction with Tank A.

#### *Shift* + 5

Pressing this key combination allows the participant or researcher to manually trigger an automation malfunction with Tank B.

#### Shift + 6

Pressing this key combination allows the participant or researcher to manually trigger an automation malfunction with Tank B.

#### *Shift* + 7

Pressing this key combination allows the participant or researcher to manually trigger an automation malfunction with Tank A.

#### *Shift* + 8

Pressing this key combination allows the participant or researcher to manually trigger an automation malfunction with Tank B.

## 6.5.3.3. Other Commands

#### Shift + F

Pressing this key combination allows the participant or researcher to manually engage the Resource Management subtask's Automated Mode. Please note that the algorithm that governs this automation is controlled by the task's configuration and is not configurable from within the task.

## Alt + F

Pressing this key combination allows the participant or researcher to manually disengage the Resource Management subtask's Automated Mode.

## Shift + A

Pressing this key combination allows the participant or researcher to toggle the Resource Management Automation's "Hide" function (see Figure 015 for an example). The command is only available when Resource Management is in Automated Mode and the subtask is configured for Automation Algorithm 1.

# 6.5.4. Tracking

## Shift + T

Pressing this key combination allows the participant or researcher to manually engage the Tracking subtask's Automated Mode.

#### Alt + T

Pressing this key combination allows the participant or researcher to manually disengage the Tracking subtask's Automated Mode.

#### Shift + L

Pressing this key combination allows the participant or researcher to manually set the Tracking subtask to Low Difficulty.

#### Shift + M

Pressing this key combination allows the participant or researcher to manually set the Tracking subtask to Moderate Difficulty.

#### Shift + H

Pressing this key combination allows the participant or researcher to manually set the Tracking subtask to High Difficulty.

#### Shift + I

Pressing this key combination allows the participant or researcher to invert the Y-Axis of the joystick. By default, pulling back on the joystick will move the Tracking reticle up the Y-Axis, and pushing forward on the joystick will move the Tracking reticle down the Y-Axis.

#### 6.5.5. **Communications**

## Shift + C

Pressing this key combination allows the participant or researcher to play an example True Communication (TC) event.

## Shift + D

Pressing this key combination allows the participant or researcher to play an example False Communication (FC) event.

#### 6.5.6. **Task Component Visibility Commands**

## Ctrl + Q

Pressing this key combination allows the participant or researcher to toggle the visibility of the System Monitoring subtask.

#### Ctrl + W

Pressing this key combination allows the participant or researcher to toggle the visibility of the Tracking subtask.

## Ctrl + E

Pressing this key combination allows the participant or researcher to toggle the visibility of the Scheduling window.

#### Ctrl + A

Pressing this key combination allows the participant or researcher to toggle the visibility of the Communications subtask.

#### Ctrl + S

Pressing this key combination allows the participant or researcher to toggle the visibility of the Resource Management subtask.

#### Ctrl + D

Pressing this key combination allows the participant or researcher to toggle the visibility of the Pump Status window.

#### 7.0 RUNNING AF-MATB

## 7.1. Initial Execution



Figure 066 – The Participant ID dialog box.

When AF-MATB is initialized, the first thing the user must enter is a Participant ID (see Figure 066). This information allows AF-MATB to generate a directory containing all performance files associated with this participant. Please be sure that your Participant ID adheres to Microsoft Windows' rules regarding illegal characters in filenames. Any Participant ID that contains any illegal characters will be rejected and the user will be required to enter a new ID. Press *OK* to submit your Participant ID.

Please note that the user may want to include trial information here as well. For example, instead of using "P01" as the ID for Participant 01, you may want to put use "P01\_T01," indicating that this is the first trial for P01. This will make identification of performance files easier. Also, be aware that AF-MATB automatically appends a date/time stamp to all files generated in order to ensure that regardless of identical file or trial names, no files generated from one trial could overwrite another.

# 7.2. Loading Behaviors

As with previous versions of AF-MATB, this version also supports the ability to load files into the task, or import them via the Configuration or Script Generator utilities. For information on importing files directly from either the Configuration or Script Generator utilities, please see sections **8.3.6 Save and Continue to AF-MATB** and **9.2.6.3 Save & Continue to AF-MATB**, respectively. Loading behaviors in this version can be divided into three phases; loading Config and Script Files (see section **7.2.1 Phase 1: Loading Config and Script Files**), loading the AF-MATB System Files (see section **7.2.2 Phase 2: Loading AF-MATB System Files**) and establishing port-triggering connections (see section **7.2.3 Phase 3: Establishing Port-Triggering Connections**).

#### 7.2.1. Phase 1: Loading Config and Script Files 7.2.1.1. Loading Files From The AF-MATB

After entering a valid Participant ID, users will be presented with a dialog box if they would like to load a file into the task (see Figure 067). Selecting *No* will automatically start Phase 2 (see section **7.2.2 Phase 2: Loading AF-MATB System Files**). Once this phase is completed, the task will render in its default state, also referred to as Free Mode. This state bypasses the port-triggering connection phase (see section **7.2.3 Phase 3: Establishing Port-Triggering Connections**) as this feature is not enabled by default. Free Mode is the preferred means for demonstrating the task to participants for the first time, because the task can run for as long as

necessary. The parameters that make up the task's configuration in Free Mode are the same parameters that are initially rendered when the Configuration Utility is opened. For more information on the Configuration Utility, please see section 8.0 AF-MATB **CONFIGURATION UTILITY.** 



Figure 067 – Dialog box asking if you would like to load a Config File or Script File.

If the user does elect to load a custom file by selecting Yes, a dialog box will launch asking the user to specify what type of file they would like to load (see Figure 068).



Figure 068 – Dialog box which asks which custom file you would like to load.

Here, users can specify whether they would like to load a Config File or Script File. In order to ensure system stability, this option differs from those in previous versions of AF-MATB, as a script can no longer be loaded independently from the parameters it was constructed with, in order to ensure system stability. Once the user selects either option, a typical Windows file selection GUI launches, where users can navigate to select the appropriate Config File or Script File (see Figure 069)

2	Please Select A Configuration Or Sc	ript File	×
Look in:	Scripts	← 🗈 💣 📰 ◄	
Recent places Desktop Libraries Computer	Name AF-MATB System Files Test_01-Aug-2013_10-52-16 Test_01-Aug-2013_10-55-33 Test_01-Aug-2013_11-00-22 Test_01-Aug-2013_11-02-30	Date modified 6/28/2013 4:09 PM 8/1/2013 10:52 AM 8/1/2013 10:58 AM 8/1/2013 11:00 AM 8/1/2013 11:02 AM	Type File folder File folder File folder File folder
	<		>
	File name: I   Files of type: MATLAB Binary Files (*.mat)	• •	Open Cancel

Figure 069 – Standard file selection utility in Windows which you can use to load custom parameters or scripts.

Upon successfully completing the loading process, the user is notified (see Figure 070), and the loading process will continue to Phase 2. If a user selects the *Script File* button and accidentally selects only a file containing task parameters, the user will be notified of this mistake and have a chance to reload the appropriate file (see Figure 071). If a user selects the *Config File* button and accidentally selects a file containing a script, the parameters will be loaded but the script will not be loaded. In the event that the user cancels the loading process by closing the file selection GUI, the user will be notified as illustrated in Figure 072.



Figure 070 – When a file has been successfully loaded, this message will be displayed.



 $Figure \ 071-{\rm Dialog} \ {\rm box} \ {\rm notifying} \ the \ {\rm user} \ that \ {\rm no} \ {\rm script} \ {\rm could} \ {\rm be} \ {\rm found}.$ 



Figure 072 – Loading was not successful due to a corrupt script or parameters file.

With the new loading system, even corrupt or damaged files, or those that have been constructed by hand by the researcher, can be handled by the task to some degree. For example, if the user elects to load a Config File that is damaged or contains missing parameters, the task will attempt to use default values for the missing variables and continue loading. In this instance, the user would be notified (see Figure 073), in case they would like to terminate the task and attempt to reload an intact file.



Figure 073 – Message notifying the user that there was one or more parameters missing from the Config File that was loaded.

The task will also attempt to cope with Script Files that contain either damaged or missing parameters, script information, or both. Each of these situations will prompt an error message explaining the problem, and then continue to load based on the information that is available. Alternatively, the user will be prompted to select a new file. Please refer to the following figures (see Figure 074, Figure 075, & Figure 076) for examples of the notification messages for each of these situations.



Figure 074 – Message notifying the user that the Script File loaded contained one or more missing or damaged parameters.



Figure 075 – Message notifying the user that the Script File loaded contained missing or damaged script information.



Figure 076 – Message notifying the user that the Script File loaded contained missing or damaged parameters, as well as missing or damaged script information.

**7.2.1.2.** Loading Files Via The Configuration and Script Generator Utilities While the task's loading function is the preferred way to load Config or Script files, this process can be slightly cumbersome, particularly when a researcher needs to test and/or modify these files. As a result, the Configuration and Script Generator utilities can launch the task quickly by bypassing the loading process and importing the information, allowing researchers to test their custom files more quickly. Instead this section will focus on an overview of the task's behaviors after custom files are imported.

When loading files from either the Configuration or Script Generator utilities, a successful loading message discussed previously (see Figure 070) will be displayed. When a Config File is imported from the Configuration Utility that is either damaged or missing parameters, the user will be notified that there has been a failure to import the data (see Figure 077).



Figure 077 – Message notifying the user that their data was unable to be successfully imported.

When a Script File is imported from the Script Generator Utility, as previously described, the user will be notified when the Script File has either damaged or missing parameters (see Figure 078), damaged or missing script information (see Figure 079), or both (see Figure 077).



Figure 078 – Message notifying the user that when attempting to import a Script File from the Script Generator Utility, the task determined that the imported Script File contained one or more damaged or missing parameters.



Figure 079 – Message notifying the user that when attempting to import a Script File from the Script Generator Utility, the task determined that the imported Script File contained damaged or missing script information.

Once the user is notified of any file importing problems, they will have the option to choose whether or not to use the task's file loading function (see Figure 080). From here, any behaviors regarding loading or transitioning Phase 2 that were discussed in the previous section apply.



Figure 080 - Message notifying the user that information could not be successfully imported from one of the companion utilities.

#### 7.2.2. Phase 2: Loading AF-MATB System Files

After completing Phase 1, the task will need to locate the critical files needed to render the task. Recall in section **5.0 BEFORE USING AF-MATB**, one of the new features in this version is the ability to automatically load the System Files when the System Files folder and task executables are located in the same directory. If this is not the case, the user will need to identify the location of the AF-MATB System Files folder using a standard Microsoft Windows directory selection GUI (see Figure 081). Simply click on the System Files folder and press *OK*. The user should see "AF-MATB System Files" in the "Folder:" field in the GUI.

Browse For Folder	×			
Please select the AF-MATB System Files Directory.				
	_			
Dell	^			
Program Files (x86)				
Dearch				
4 퉬 Users				
4 鷆 William D Miller Jr				
Downloads				
My Documents				
AF-MATB				
AF-MATB System Files				
🌗 AF-MATB Audio				
🛛 🔜 AlienFX				
Car 🗸				
Eolder: AF-MATB System Files				
Make New Folder OK Cancel				

Figure 081 – Standard directory selection GUI in Windows used to identify the location of the AF-MATB System Files folder.

## 7.2.3. Phase 3: Establishing Port-Triggering Connections

As outlined in section **4.5 Serial and Digital Port-Triggering**, this version of the task can send information about the task to external acquisition systems. Once Phases 1 and 2 of the loading process have been completed, Phase 3 will start (if configured by the researcher), and the task will attempt to initialize and test the port-triggering setup. Since initializing the connection for digital triggering may take a few moments, users will be notified that the loading process starts (see Figure 082).

🛃 NIDA	-		х
Loading NIDA	Q Li	brary	

Figure 082 – Message notifying the user that the task is loading the libraries required to initialize a digital connection.

Additionally, the user will be notified once initialization for either the serial (see Figure 083) and/or digital (see Figure 084) triggering is successful, or if either the serial (see Figure 085) and/or digital (see Figure 086) triggering fails. Please note that unlike serial triggering, digital triggering initialization can fail due to misconfiguration of the task or missing drivers/DLLs. For example, Figure 086 shows the message users receive in the event of a complete failure to initialize a digital triggering. However, in the event that the task is misconfigured, such as if the task is configured to talk to an invalid Device ID, users will receive a message like that shown in Figure 087, which detail a series of error codes that can be used by the researcher to troubleshoot the specific issue. For more information on these error codes and how to troubleshoot, see sections **4.5 Serial and Digital Port-Triggering** and **8.2.11 Port Triggering Parameters**.



Figure 083 – Message notifying the user of a successful serial port connection.



Figure 084 – Message notifying the user of a successful digital port connection.



Figure 085 – Message notifying the user of a failed serial port connection.



Figure 086 – Message notifying the user of a total failure to establish a digital port connection.



Figure 087 – Error code dialog produced by a misconfiguration of the task

#### 7.3. Saving Behaviors

The parameters defined in the Config or Script Files dictate the wide variety of performance directories that can be generated. Each directory contains a slightly different complement of log files containing saved performance data. The following section (section **7.3.1 Performance Directory Types**) will discuss the creation of these directories, followed by a more in-depth explanation of the details of each particular log file (section **7.3.2 Log Details and Descriptions**) the performance summaries (section **7.3.3 Performance Summary Details and Descriptions**), and the data file (section **7.3.4 DataStructure Details and Descriptions**) generated by the task.

Although the performance data in the log files can be accessed before a participant has completed a trial, doing so may interrupt the logging process and result in missing data. Therefore, it is suggested that researchers wait until the task window and "Saving In Progress" window (see Figure 088) close before accessing any data.



Figure 088 - "Saving In Progress" message, notifying the user to please wait until the data has finished saving.

#### 7.3.1. Performance Directory Types

Any time the user opens the task and enters a Participant ID, a folder with the participant's ID and a date/timestamp is created and saved in the same directory that the task's executables are located. As previously discussed, this practice ensures that even when entering the same ID, no data generated by this task can be overwritten. The updated performance file section now generates specific output files based on the configuration of the task. The following sections outline these output directories.

#### 7.3.1.1. Free Mode

When the user elects not to load any script into the task, the task will operate in Free Mode. In Free Mode, performance data will only be saved to the Master Event Log (see Figure 089).



Figure 089 – Performance directory generated when the user does not load any script into the task.

#### 7.3.1.2. Single Condition, Standard Trial

This performance directory is generated when the user loads a script that contains one condition using the default configuration. Specifically, this particular directory is generated when IT Mode is disengaged, System Monitoring Automation is disengaged, and the Resource Management subtask is operating under Algorithm 1.

All of the performance directories are comprised of two sections. The first section is the root of the performance directory (see Figure 090), which contains the Master Event Log, Port Log, and Transition Log, and the DataStructure file. The second section is the individual "Condition" directories. In this example, due to the fact that the script was comprised of only one condition, only one "Condition" directory was created. Each of these directories contains a number of log files specific to that condition (see Figure 091). The logs included are: the Communication Log, Correct Gauge Response Log, Correct Light Response Log, Incorrect Communication Log, Incorrect Gauge Response Log, Incorrect Light Response Log, Resource Management Values Log, Standard Performance Summary, and Tracking Coordinate Log.

🌆 l 💽 🗿 = l		Test1_05-Aug-2	013_14-30-4	0	- 1	×
File Home	Share V	/iew				^ 😮
Copy Paste	i <b>ut</b> C <b>opy path</b> aste shortcut d	Move to	New New	Properties • Open • History	Select all Select nor Invert select	ne ection
	- 	B > Test1 05-Δυα-2013 14-30	-40 4	C. Search Test1 05-A	ug-2013 14-30	-40 0
☆ Favorites	^ Na	me	10 1	Date n	nodified	Туре
Desktop		Condition_1		8/5/20	13 2:30 PM	File folder
🐌 Downloads	7	Test1_05-Aug-2013_14-30-40_[	DataStructure	8/5/20	13 2:32 PM	Microsoft
🖳 Recent places	s  🖹	Test1_Master_Event_Log.res		8/5/20	13 2:32 PM	RES File
🝊 SkyDrive	X	Test1_Port_Log.res		8/5/20	13 2:30 PM	RES File
	X	Test1_Transition_Log.res		8/5/20	13 2:32 PM	RES File
🥽 Libraries						
Documents						
🌙 Music						
Pictures						
Videos						
🤣 Homegroup						
🖳 Computer	~ <					>
5 items   1 item se	elected Stat	te: 跳 Shared				:==

Figure 090 – The root of a Single Condition, Standard Trial performance directory.

👪 l 🕞 👪 = l	Conditio	n_1		-	×
File Home Sha	ire View				^ 🕜
Copy Paste	aath hortcut Move to → X Delete → E Copy to → E Rename	New folder	Properties ▼	Select all Select no Invert select	ne ection
Clipboard	Organize	New	Open	Select	
🗲 ラ 🔻 🚹 «	Test1_05-Aug-2013_14-30 → Condition_	1 v č	Search Condition	1	Q
🔆 Favorites	^ Name		Date m	nodified	Туре
🛄 Desktop	X Test1_Correct_Comm_Log_1.res		8/5/20	13 2:30 PM	RES File
〕 Downloads	K Test1_Correct_Gauge_Log_1.res		8/5/20	13 2:30 PM	RES File
🖳 Recent places	K Test1_Correct_Light_Log_1.res		8/5/20	13 2:30 PM	RES File
🐔 SkyDrive	Test1_Incorrect_Comm_Log_1.res	s	8/5/20	13 2:30 PM	RES File
	Test1_Incorrect_Gauge_Log_1.res	;	8/5/20	13 2:30 PM	RES File
🥽 Libraries	Test1_Incorrect_Light_Log_1.res		8/5/20	13 2:30 PM	RES File
Documents	X Test1_Resource_Management_Lo	og_1.res	8/5/20	13 2:32 PM	RES File
🌙 Music	X Test1_Standard_Performance_Su	mmary_1.res	8/5/20	13 2:32 PM	RES File
Pictures	K Test1_Tracking_Log_1.res		8/5/20	13 2:32 PM	RES File
Videos					
🔞 Homegroup					
👰 Computer	v <				>
9 items 🔰 State: 跳 Sha	red				:==

Figure 091 – Contents of a "Condition" directory for a Standard Trial performance directory.

# 7.3.1.3. Multiple Condition, Standard Trial

Trials with multiple conditions in a script automatically generate additional "Condition" directories that contain performance related specifically to each condition. The contents of these additional directories are identical to those shown in Figure 091, except that the number at the end of each log will reflect the number of each condition in the directory. For example, log files contained in the "Condition\_1" directory all have a filename that ends in "\_1.res," while all log files contained in the "Condition\_2" directory have filenames that end in "\_2.res," etc. The only other difference of note in this type of directory is the additional Standard Performance Summary Log generated in the root of the performance directory as shown in Figure 092. This log contains a summary of a participant's performance across all conditions in a trial.

🔉 I 🕞 🕼 = I	Test2_05-Aug-	2013_14-31-0	5	- 0	×
File Home Share	View				^ 🕐
Copy Paste	h Move to $\checkmark$ Delete $\checkmark$ ntcut Copy to $\checkmark$ 🛱 Rename	New folder	Properties	Select all	ne ection
Clipboard	Organize	New	Open	Select	
⋲ 🌛 🔻 🕇 🚺 « Al	-MATB > Test2_05-Aug-2013_14-3	1-05 🗸 🗸	C Search Test2_05-A	lug-2013_14-31	-05 🔎
☆ Favorites	Name		Date n	nodified	Туре
🔤 Desktop	퉬 Condition_1		8/5/20	)13 2:31 PM	File folder
🐌 Downloads	퉬 Condition_2		8/5/20	)13 2:31 PM	File folder
🖳 Recent places	🛃 Test2_05-Aug-2013_14-31-05_	DataStructure	8/5/20	)13 2:33 PM	Microsoft
🝊 SkyDrive	X Test2_Master_Event_Log.res		8/5/20	)13 2:33 PM	RES File
	Test2_Port_Log.res		8/5/20	)13 2:31 PM	RES File
📜 Libraries	Test2_Standard_Performance_	Summary.res	8/5/20	)13 2:33 PM	RES File
Documents	X Test2_Transition_Log.res		8/5/20	)13 2:33 PM	RES File
👌 Music					
Pictures					
💾 Videos					
🤣 Homegroup					
👰 Computer 🗸 🗸	<				>
7 items State: 🎎 Shared					:==

Figure 092 – The root of a Multiple Condition, Standard Trial performance directory. Note the additional Performance Summary log.

#### 7.3.1.4. Single Condition, System Automation Trial

This performance directory is generated when users load a script that contains a condition in which the System Monitoring subtask's Automated Mode is engaged. In this mode, two additional files are found in each "Condition" directory designed to provide all of the necessary details regarding a participant's performance while the System Monitoring subtask was automated. These files, as shown in Figure 093, are the System Automation Log and System Automation Performance Summary. The root of the System Monitoring Automation Trial performance directory is identical to the root shown in Figure 090.

👪 l ⊋ 🚯 = l	Condition_1		- 1	×
File Home Share	View			^ <b>(</b> )
Copy Paste	h ortcut Copy to - Rename New folder Properties	Open ▼ Edit History	Select all Select nor Invert sele	ne ection
Clipboard	Organize New Ope	n	Select	
🗲 🏵 🔻 🕇 퉬 « S	sAuto1_05-Aug-2013_1 → Condition_1 v C Search	Condition_1		م ر
☆ Favorites	Name	Date mo	odified	Туре
E Desktop	₭ SysAuto1_Correct_Comm_Log_1.res	8/5/201	3 2:31 PM	RES File
🚺 Downloads	✗ SysAuto1_Correct_Gauge_Log_1.res	8/5/201	3 2:31 PM	RES File
🖳 Recent places	SysAuto1_Correct_Light_Log_1.res	8/5/201	3 2:31 PM	RES File
🐔 SkyDrive	SysAuto1_Incorrect_Comm_Log_1.res	8/5/201	3 2:31 PM	RES File
	X SysAuto1_Incorrect_Gauge_Log_1.res	8/5/201	3 2:31 PM	RES File
🛜 Libraries	🔣 SysAuto1_Incorrect_Light_Log_1.res	8/5/201	3 2:31 PM	RES File
Documents	X SysAuto1_Resource_Management_Log_1.res	8/5/201	3 2:33 PM	RES File
J Music	X SysAuto1_Standard_Performance_Summary_1.res	8/5/201	3 2:33 PM	RES File
Pictures	🔣 SysAuto1_System_Automation_Log_1.res	8/5/201	3 2:31 PM	RES File
😸 Videos	K SysAuto1_System_Automation_Performance_Summary_1.res	8/5/201	3 2:33 PM	RES File
	K SysAuto1_Tracking_Log_1.res	8/5/201	3 2:33 PM	RES File
輚 Homegroup				
👰 Computer 🗸 🗸	<			>
11 items 🛛 State: 👫 Shar	ed			:

Figure 093 – Contents of a "Condition" directory for a System Automation Trial performance directory.

# 7.3.1.5. Multiple Condition, System Automation Trial

This directory is created after loading a script with multiple conditions in which one or more conditions are operating in Automated Mode. Please be aware that if any condition in the trial has the System Monitoring subtask operating in Automated Mode, the System Automation Log and System Automation Performance Log will be generated for all "Condition" directories in that trial. Automation logs that occur in "Condition" directories where the System Monitoring subtask was not operating in Automated Mode can be ignored.

As in the previous section discussing multiple-condition trials, the only change in the root performance directory is the addition of a Summary Performance Log which details a participant's performance across trials. However, in this case, an additional System Automation Performance Summary Log is also generated (see Figure 094)



Figure 094 – The root of a Multiple Condition, System Automation Trial performance directory. Note the additional System Automation Performance Summary Log.

#### 7.3.1.6. Single Condition, Resource Automation Trial

This performance directory is generated when a user loads a script that contains a condition that has the Resource Management subtask using Automation Algorithm 2. Similar to the System Automation Trials, Resource Management Automation Trials also generate two other files in addition to the basic files shown in Figure 091. These files are the Resource Automation Log and the Resource Automation Performance Summary (see Figure 095). The purpose of these files is identical to that of their System Monitoring counterparts; log all relevant participant behavior, as well as synthesize their performance in a succinct output.

Please note that any script configured to use Automation Algorithm 2 for its Automated Mode, even if not engaged for that condition or even that trial, will generate these additional log files. Just like the System Automation Trials, they can be ignored, except when a condition engaged the Resource Management subtask's Automated Mode using this specific algorithm. Automation Algorithm 1 will not generate these logs.

🔉 I 🕞 🚯 = I	Condition_1		- 🗆 🗙
File Home Sha	e View		^ 😮
Copy Paste	ath nortcut Copy to * Delete * New folder Properties	Copen - Sele Edit : Sele History : Inve	ct all ct none rt selection
Clipboard	Organize New Ope	n S	elect
🐑 🕘 🔻 🕇 퉬 «	ResAuto1_05-Aug-2013_1 ▶ Condition_1 v C Search	Condition_1	Q
🔶 Favorites	Name	Date modified	Туре
🔤 Desktop	ResAuto1_Correct_Comm_Log_1.res	8/5/2013 2:33 P	M RES File
〕 Downloads	ResAuto1_Correct_Gauge_Log_1.res	8/5/2013 2:33 P	M RES File
📃 Recent places	ResAuto1_Correct_Light_Log_1.res	8/5/2013 2:33 P	M RES File
🐔 SkyDrive	ResAuto1_Incorrect_Comm_Log_1.res	8/5/2013 2:33 P	M RES File
	ResAuto1_Incorrect_Gauge_Log_1.res	8/5/2013 2:33 P	M RES File
📄 Libraries	ResAuto1_Incorrect_Light_Log_1.res	8/5/2013 2:33 P	M RES File
Documents	ResAuto1_Resource_Automation_Log_1.res	8/5/2013 2:33 P	M RES File
🌙 Music	ResAuto1_Resource_Automation_Performance_Summary_1.r	es 8/5/2013 2:35 P	M RES File
Pictures	ResAuto1_Resource_Management_Log_1.res	8/5/2013 2:35 P	M RES File
🛃 Videos	ResAuto1_Standard_Performance_Summary_1.res	8/5/2013 2:35 P	M RES File
	ResAuto1_Tracking_Log_1.res	8/5/2013 2:35 P	M RES File
輚 Homegroup			
🖳 Computer	<		>
11 items 🛛 State: 🐉 Shi	red		:==

Figure 095 – Contents of a "Condition" directory for a Resource Management Automation Trial performance directory.

#### 7.3.1.7. Multiple Condition, Resource Automation Trial

As in the other Multiple Condition Trials previously discussed, this directory is constructed when a participant loads a script that contains more than one condition and is configured to use Automation Algorithm 2 for the Resource Management subtask. This trial generates the typical aggregate Standard Performance Summary Log, and in this case, an aggregate Resource Automation Performance Summary Log (see Figure 096).

👪 l ⊋ 👪 = l		ResAuto2_05	-Aug-2013_14	-42-4(	)	-	□ ×
File Home	Share	View					^ 🕜
Copy Paste	ut opy path aste shortcut	t Move to ▼ X Dele	ame New folder	]∙ ]∙ Pr	operties	Select all	one lection
Clipboard	1	Organize	New		Open	Select	
€ 🦻 ד ↑ 🌗	« AF	▶ ResAuto2_05-Aug-2013_	14-42-40	~ C	Search ResAuto2	_05-Aug-2013_	14 <b>,0</b>
🔆 Favorites	^ N	lame	*		Date	modified	Туре
Desktop		Condition_1			8/5/2	013 2:44 PM	File folder
〕 Downloads		Condition_2			8/5/2	013 2:44 PM	File folder
🖳 Recent places	5	ResAuto2_05-Aug-2013_1	14-42-40_DataStru	icture	8/5/2	013 2:45 PM	Microsoft
🐔 SkyDrive		KesAuto2_Master_Event_	Log.res		8/5/2	013 2:45 PM	RES File
	í.	🔣 ResAuto2_Port_Log.res			8/5/2	013 2:44 PM	RES File
🥽 Libraries	ĺ.	🔣 ResAuto2_Resource_Auto	omation_Performa	ance_Su	immary.res 8/5/2	013 2:45 PM	RES File
Documents	l.	🔣 ResAuto2_Standard_Perf	ormance_Summai	ry.res	8/5/2	013 2:45 PM	RES File
🎝 Music	ĺ.	KesAuto2_Transition_Log	j.res		8/5/2	013 2:45 PM	RES File
Pictures							
📑 Videos							
🤣 Homegroup							
👰 Computer	~ <						>
8 items 👘 State: 跳	Shared						:==

Figure 096 - The root of a Multiple Condition, Resource Automation Trial performance directory. Note the additional Resource Automation Performance Summary Log.

#### **Single Condition, Dual Automation Trial** 7.3.1.8.

This performance directory is generated when a user loads a script with the Resource Management subtask configured to use Automation Algorithm 2 when automation is engaged and the System Monitoring subtask is configured to operate in Automated Mode. As illustrated in Figure 097, this "Condition directory will contain both sets of unique logs that were previously discussed (see Figure 093 and Figure 095). As usual, no changes to the root of the performance directory are made.
🌆 l 🕞 📓 = l		Conditi	ion_1			— t	×
File Home Sh	are V	ïew					^ 🕐
Copy Paste	path shortcut	Move to V Delete V De	New folder New	Properties	pen ▼ dit istory	Select all Select nor Invert sele	ne ection
( → ↑ ) → ∧ ) →	SysResAu	ıto1_05-Aug-201 ▶ Condition	n_1 ∨	🖒 Search Co	ndition_1		Q
☆ Favorites	^ Na	me			Date mo	dified	Туре
E Desktop	X	SysResAuto1_Correct_Comm_L	.og_1.res		8/5/2013	3 2:34 PM	RES File
鷆 Downloads	X	SysResAuto1_Correct_Gauge_L	og_1.res		8/5/2013	3 2:34 PM	RES File
📃 Recent places	X	SysResAuto1_Correct_Light_Lo	g_1.res		8/5/2013	3 2:34 PM	RES File
🐔 SkyDrive	X	SysResAuto1_Incorrect_Comm	_Log_1.res		8/5/2013	3 2:34 PM	RES File
	X	SysResAuto1_Incorrect_Gauge_	Log_1.res		8/5/2013	3 2:34 PM	RES File
詞 Libraries	X	SysResAuto1_Incorrect_Light_L	.og_1.res		8/5/2013	3 2:34 PM	RES File
Documents	X	SysResAuto1_Resource_Autom	ation_Log_1.res		8/5/2013	3 2:34 PM	RES File
👌 Music	X	SysResAuto1_Resource_Autom	ation_Performa	nce_Summary	8/5/2013	3 2:35 PM	RES File
Pictures	X	SysResAuto1_Resource_Manag	ement_Log_1.re	25	8/5/2013	3 2:35 PM	RES File
📑 Videos	X	SysResAuto1_Standard_Perform	nance_Summan	y_1.res	8/5/2013	2:35 PM	RES File
	X	SysResAuto1_System_Automat	ion_Log_1.res		8/5/2013	3 2:34 PM	RES File
輚 Homegroup	X	SysResAuto1_System_Automat	ion_Performan	ce_Summary_1	8/5/2013	2:35 PM	RES File
	X	SysResAuto1_Tracking_Log_1.r	es		8/5/2013	3 2:35 PM	RES File
👰 Computer	~ <						>
13 items 👘 State: 跳 SI	hared						:== 🖿

Figure 097 – Contents of a "Condition" directory for a Dual Automation Trial performance directory.

# 7.3.1.9. Multiple Condition, Dual Automation Trial

This performance directory is generated when a user loads a script that meets the following criteria. First, the script must contain multiple conditions. Second, the Resource Management subtask must be configured to use Automation Algorithm 2 when automation is engaged. Finally, the System Monitoring subtask must operate in Automated Mode for at least cone condition. In the event that all of these things are true, each "Condition" directory will be identical to those in the Single Condition, Dual Automation Trial (see Figure 097). Finally, the root of the performance directory will contain three separate aggregated performance summary files: the Resource Automation Performance Summary Log, the Standard Performance Summary Log, and the System Automation Performance Summary Log (see Figure 098).

🏭 I 💽 👪 = I		SysRes	Auto2_05-Au	ig-2013_14-3	34-24	4		-	
File Home	Share	View							^ 🕜
Copy Paste	ut opy path iste shortc	Wove to ▼ ™ Copy to ▼	➤ Delete ▾ Rename	New folder	Proj	perties ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓	pen ⇒ dit istory	Select all	ne lection
Clipboard		Orga	anize	New		Open		Select	
🔄 🕘 👻 🚹	« AF	▶ SysResAuto2_	05-Aug-2013_14	v	Ç	Search Sys	ResAuto	2_05-Aug-20	13 🔎
🔆 Favorites	^	Name	^				Date m	nodified	Туре
📰 Desktop		Condition_1					8/5/20	13 2:34 PM	File folder
🝌 Downloads		퉬 Condition_2					8/5/20	13 2:34 PM	File folder
📃 Recent places		SysResAuto2_0	5-Aug-2013_14-	34-24_DataStru	cture		8/5/20	13 2:35 PM	Microsoft
🐔 SkyDrive		🔣 SysResAuto2_N	Master_Event_Lo	g.res			8/5/20	13 2:35 PM	RES File
		🔀 SysResAuto2_P	ort_Log.res				8/5/20	13 2:34 PM	RES File
🥽 Libraries		🛣 SysResAuto2_R	esource_Autom	ation_Performa	ince_S	Summary	8/5/20	13 2:35 PM	RES File
Documents		🔣 SysResAuto2_S	tandard_Perforn	nance_Summar	y.res		8/5/20	13 2:35 PM	RES File
🌙 Music		🔣 SysResAuto2_S	ystem_Automat	ion_Performan	ce_Su	mmary.res	8/5/20	13 2:35 PM	RES File
Pictures		🛣 SysResAuto2_T	ransition_Log.re	es			8/5/20	13 2:35 PM	RES File
🛃 Videos									
🤣 Homegroup									
🜉 Computer	~ <								>
9 items 🛛 State: 🞎	Shared								:==

Figure 098 – The root of a Multiple Condition, Dual Automation Trial performance directory. Note the additional Resource Automation and System Automation Performance Summary Logs.

## 7.3.1.10. Single Condition, IT Mode Trial

This performance directory is generated when the user has configured the task with the IT Mode enabled. Note that this task does not allow the Resource Management subtask to be configured using Automation Algorithm 2, and Automated Mode for the System Monitoring subtask has been disabled. As a result, the only difference between the performance directory in this configuration and the Single Condition, Standard Trial is the inclusion of an Information Throughput Summary Log in the "Condition" directory (see Figure 099).

👪 l 🕞 👪 = l	Condition_1		- 🗆 🗙
File Home Shar	View		^ <b>(</b> )
Copy Paste 2 Paste st	th ortcut Move to ~ X Delete ~ Copy to ~ Rename New folder	Properties	Select all Select none
Clipboard	Organize New	Open	Select
🔄 🔿 🔻 🕇 🕌 « I	[1_05-Aug-2013_14-35-24	C Search Condition	.1 <b>,</b> 2
☆ Favorites	Name	Date n	nodified Type
🛄 Desktop	IT1_Correct_Comm_Log_1.res	8/5/20	13 2:35 PM RES File
〕 Downloads	IT1_Correct_Gauge_Log_1.res	8/5/20	13 2:35 PM RES File
🕮 Recent places	IT1_Correct_Light_Log_1.res	8/5/20	13 2:35 PM RES File
🐔 SkyDrive	IT1_Incorrect_Comm_Log_1.res	8/5/20	13 2:35 PM RES File
	IT1_Incorrect_Gauge_Log_1.res	8/5/20	13 2:35 PM RES File
🥽 Libraries	IT1_Incorrect_Light_Log_1.res	8/5/20	13 2:35 PM RES File
Documents	X IT1_Information_Throughput_Summary_1.res	8/5/20	13 2:36 PM RES File
J Music	IT1_Resource_Management_Log_1.res	8/5/20	13 2:36 PM RES File
Pictures	🔣 IT1_Standard_Performance_Summary_1.res	8/5/20	13 2:36 PM RES File
Videos	IT1_Tracking_Log_1.res	8/5/20	13 2:36 PM RES File
🔣 Homegroup			
👰 Computer 🔍	<		>
10 items 🛛 State: 跳 Sha	ed		:==

Figure 099 – Contents of a "Condition" directory for an IT Mode Trial performance directory.

# 7.3.1.11. Multiple Condition, IT Mode Trial

The last type of performance directory that can be generated occurs when the user has configured the task IT Mode enabled and has loaded a script that contains multiple conditions. In this configuration, as in previous ones, the multi-condition "Condition" directories do not differ from the single condition directories (see Figure 099). In this configuration, an aggregated Information Throughput Summary accompanies the Standard Performance Summary (see Figure 100) found in the root of the performance directory.

🏨 l 💽 👪 = l		IT2_05-Aug-20	13_14-35-49	)	_ (	×
File Home	Share	View				^ 🕐
Copy Paste	Cut Copy path Paste shortcut	Move to → X Delete → Copy to → A Rename	New folder	Properties ▼ History	Select all	ne ection
Clipboar	d	Organize	New	Open	Select	
€ ∋ - ↑ [	📔 « AF-MA	TB → IT2_05-Aug-2013_14-35-49	9 🗸	C Search IT2_05-Au	g-2013_14-35-4	م 9
🔆 Favorites	^ N	ame		Date r	nodified	Туре
📃 Desktop		Condition_1		8/5/20	)13 2:35 PM	File folder
🗼 Downloads		Condition_2		8/5/20	)13 2:35 PM	File folder
🖳 Recent place	es 🛛 🛃	IT2_05-Aug-2013_14-35-49_Dat	taStructure	8/5/20	)13 2:37 PM	Microsoft
🐔 SkyDrive		IT2_Information_Throughput_S	Summary.res	8/5/20	)13 2:37 PM	RES File
		IT2_Master_Event_Log.res		8/5/20	)13 2:36 PM	RES File
🥃 Libraries		IT2_Port_Log.res		8/5/20	)13 2:35 PM	RES File
Documents	2	IT2_Standard_Performance_Sur	mmary.res	8/5/20	)13 2:37 PM	RES File
🌙 Music	2	IT2_Transition_Log.res		8/5/20	)13 2:37 PM	RES File
Pictures						
📑 Videos						
🔞 Homegroup						
👰 Computer	~ <					>
8 items 🔰 State: 🎎	Shared					:==

Figure 100 – The root of a Multiple Condition, IT Mode Trial performance directory. Note the additional Information Throughput Summary Logs.

# 7.3.2. Log Details and Descriptions

All log files generated by AF-MATB are generated as .res files. These files are formatted as tabdelimited text files, meaning any text reader, including Notepad, WordPad, Textpad++, or Microsoft Excel can read them. They are designed to provide extensive information about the individual components of the task.

The following sections detail the purpose of all logs generated by the task, as well as all of the information that may be contained in them. Unless otherwise stated, the information contained in any given log applied only to the "Condition" directory in which it resides.

# 7.3.2.1. Correct Comm Log

This log contains a record of any Communications Event (CE) that occurred, as well as the participant's response to that CE. Due to the complexity and open-ended nature of responses for this task, a scoring hierarchy must be used to explain participant actions. For example, in conditions with a high event density, participants may respond to CEs out-of-order. As a result, this log contains each participant's response matched to the CE that was most likely to have prompted it. For details on the hierarchy of this log see section **7.3.3.1.4 Communications Section**.

Fields contained in this log include:

- a. <u>Event Time:</u> Time when any CE was played by the task. Remember that a CE can include either a True Communication (TC) event or a False Communication (FC) event.
- b. Correct Channel: The channel a participant was instructed to set their radio to. Values for this field range from 1 to 4, corresponding to "NAV1," "NAV2," "COM1," and "COM2."
- c. Correct Frequency: The frequency a participant was instructed to set their radio to.
- d. Target Signal: Whether or not the CE was considered a TC or FC. Values for this field are either 0, corresponding to FCs, or 1, corresponding to TCs.
- e. <u>Timeout:</u> According to the researcher's specifications, the time at which the CE is scheduled to time-out after accounting for the duration of the event.
- f. <u>Response Time:</u> The time at which a participant responded to the event. The time recorded is dependent on the scoring algorithm's determination that a certain response was prompted by a particular CE.
- g. <u>Response Channel:</u> The channel that was used in response to a CE. The Response Channel is dependent on the scoring algorithm's determination that a certain response was prompted by a particular CE. Values for this field range from 1 to 4, just as in the Correct Channel field.
- h. Response Frequency: The frequency that was used in response to a CE. The Response Frequency is dependent on the scoring algorithm's determination that a certain response was prompted by a particular CE.

#### 7.3.2.2. **Correct Gauge Log**

This log contains a record of the onset of the Gauge events in the System Monitoring subtask, as well as the participant's correct responses to those events.

Fields contained in this log include:

- a. Event Time: Onset time of any Gauge event.
- b. Gauge Number: Gauge targeted for that event. Values range from 1 to 4, corresponding to the four gauges of the System Monitoring subtask.
- c. Response Time: The participant's response to that Gauge event.
- d. Event Timeout: According to the researcher's specifications, the time at which the Gauge event was scheduled to time-out.

#### 7.3.2.3. **Correct Light Log**

This log contains a record of the onset of the Light events in the System Monitoring subtask, as well as the participant's correct responses to those events.

Fields contained in this log include:

- a. Event Time: Onset time of any Light event.
- b. Gauge Number: Light targeted for that event. Values are either 1 or 2.
- c. Response Time: The participant's response to that Light event.
- d. Event Timeout: According to the researcher's specifications, the time at which the Light event was scheduled to time-out.

## 7.3.2.4. Incorrect Comm Log

This log contains a record of any participant response that could not be attributed to a CE. Responses contained in this log include Unexplained True, Unexplained False, and No Event responses. For information on these responses, please see **7.3.3.1.4 Communications Section**.

Fields contained in this log include:

- a. <u>Response Time:</u> When the participant response occurred.
- b. <u>Channel:</u> The channel locked-in by the participant.
- c. <u>Frequency:</u> The frequency locked-in by the participant.

#### 7.3.2.5. Incorrect Gauge Log

This log contains a record of all false alarms by the participant for the Gauges portion of the System Monitoring subtask.

Fields contained in this log include:

- a. <u>Response Time:</u> When the participant response occurred.
- b. <u>Gauge Number</u>: Which gauge the participant incorrectly identified was out of range.

#### 7.3.2.6. Incorrect Light Log

This log contains a record of all false alarms by the participant for the Lights portion of the System Monitoring subtask.

Fields contained in this log include:

- a. <u>Response Time:</u> When the participant response occurred.
- b. <u>Light Number</u>: Which light the participant incorrectly identified was out of range.

#### 7.3.2.7. Master Event Log

This log contains all of the actions performed by the user as well as any action the task. Recall that this file is the only file that is generated regardless of whether a script has been loaded into the task, that this log is located in the root performance directory, and that this log contains information for all conditions in a given trial.

Fields contained in this log include:

- a. <u>Event Time:</u> Time at which any participant or task action occurred.
- b. <u>Task or Item:</u> Describes what subtask or specific item (Gauge 1, Pump 1, etc.) of a subtask was affected by the action.
- c. <u>Action:</u> Describes the exact details of the action.

## 7.3.2.8. Port Log

This log contains a detailed list of all port-triggering commands issued by the task. This log was designed as a secondary source of information that can be used to check the validity of port information received by data acquisition systems. Recall that this log is located in the root performance directory and that this log contains information for all conditions in a given trial.

Fields contained in this log include:

- a. <u>Event Time:</u> According to the task, when the port-triggering command was executed.
- b. <u>Description</u>: The details of the port-triggering command.
- c. <u>Up/Down or Serial?</u>: This field is used to indicate whether a command in the Port Log was a serial port-triggering command, or was either the up-signal or down-signal for a digital port-triggering command.

## 7.3.2.9. Resource Automation Log

This log contains an overview of all event-related actions for the Resource Management subtask when in Automated Mode and configured to use Automation Algorithm 2.

Fields contained in this log include:

- a. Event Time: Time at which either a participant action or task action occurred.
- b. <u>Event Code:</u> Number used to note the specific state of each event. Numbers in this field range from 1-5 and represent the following actions:
  - i. 1: Automation failure was triggered
  - ii. 2: Automation failure is now visible to the participant
  - iii. 3: Participant identified automation failure
  - iv. 4: Automation restored to normal due to timeout.
  - v. 5: Participant incorrectly identified an automation failure.
- c. <u>Event Description</u>: Describes the exact nature of the <u>Event Code</u> field, as well as provides details as to what specific automation behavior was triggered.

## 7.3.2.10. Resource Management Log

This log contains all of the values for the Resource Management subtask's Tank A and Tank B volumes over the course of a condition. The rate at which these data are sampled and saved to this log is approximately equal to the specific cycling rate of the task. Also note that this log contains three key pieces of information required to calculate performance metrics for the Resource Management subtask. For information on the cycling rate of the task and configuring the parameters that influence the Resource Management subtask's scoring, please see sections **8.2.2.9 Timer Cycling Rate (Hz)** and **8.2.9.7 Resource Management RMS Interval (Seconds)**, respectively.

Fields contained in this log include:

- a. <u>RMS Target Value is:</u> One of the three key pieces of information needed to calculate performance metrics for the Resource Management subtask. This field represents the target volume for Tank A and Tank B.
- b. <u>Main Tank Automation Lower Limit is:</u> The second key piece of information needed to calculate performance metrics for this subtask. This field represents the minimum acceptable volume that for Tank A or Tank B to be considered within range.

- c. Main Tank Automation Upper Limit is: The third key piece of information needed to calculate performance metrics for this subtask. This field represents the maximum acceptable volume that for Tank A or Tank B to be considered within range.
- d. Event Time: Time at which the volumes were sampled.
- e. Tank A Value: Value of Tank A in liters.
- f. Tank B Value: Value of Tank B in liters.
- g. Tank A Difference: Used in calculating performance metrics for this subtask. This value is defined as

 $Tank \ A \ Difference = (Tank \ A \ Value - RMS \ Target \ Value)$ 

h. Tank B Difference: Used in calculating performance metrics for this subtask. This value is defined as

Tank B Difference = (Tank B Value - RMS Target Value)

i. Tank A Deviation: Used in calculating performance metrics for this subtask. This value is defined as

 $Tank \ A \ Deviation = Tank \ A \ Difference^2$ 

i. Tank B Deviation: Used in calculating performance metrics for this subtask. This value is defined as

$$Tank B Deviation = Tank B Difference^{2}$$

k. Euclidian Distance From Target: By treating Tank A and Tank B as two dimensions, the Euclidian distance of both tank volumes from their target value can be computed. This value is defined as

*Euc. Dist From Target* =  $\sqrt{(Tank \ A \ Deviation + Tank \ B \ Deviation))}$ 

- 1. Tank A In Range?: Boolean value that indicates whether Tank A is within the Lower and Upper Limits specified at the top of the file. Values are 0 or 1, indicating whether the tank's volume is out of range or within range, respectively.
- m. Tank B In Range?: Boolean value that indicates whether Tank B is within the Lower and Upper Limits specified at the top of the file. Values are 0 or 1, indicating whether the tank's volume is out of range or within range, respectively.
- n. Both In Range?: Boolean value that indicates whether Tank A and Tank B are within the Lower and Upper Limits specified at the top of the file. Values are 0 or 1, indicating whether one or both tanks' volumes are out of range, or whether they're both within range, respectively.

## 7.3.2.11. System Automation Log

This log contains an overview of all event-related actions for the System Monitoring subtask when in Automated Mode.

Fields contained in this log include:

- a. <u>Event Time:</u> Time at which either a participant action or task action occurred.
- b. <u>Event Code:</u> Numbers ranging from 1-6 are used to note the specific state of each event:
  - i. 1: Automation fails to detect a gauge malfunction.
  - ii. 2: Participant identified automation failure.
  - iii. 3: Automation failure restored to normal due to timeout.
  - iv. 4: Participant incorrectly identified an automation failure.
  - v. 5: Automation detects a gauge malfunction, which requires no action.
  - vi. 6: Automation repairs the gauge malfunction, which requires no action
- c. <u>Event Description</u>: Describes the exact nature of the <u>Event Code</u> field.

# 7.3.2.12. Tracking Log

This log contains all of the values for the Tracking subtask's reticle position over the course of a condition. Please note that the rate at which data are sampled and saved to this log is approximately equal to the specific cycling rate of the task. For information on configuring the cycling rate of the task please see section **8.2.2.9 Timer Cycling Rate (Hz)**.

The key pieces of information required to calculate performance metrics for the Tracking subtask will vary depending on how the researcher has configured the subtask's outline. As a result, unless otherwise noted, all fields contained in this log will be available regardless of configuration. For more information on configuring the parameters that influence the Tracking subtask's scoring, please see section **8.2.4 Tracking Subtask Basic Parameters Group**.

Fields contained in this log can include:

- a. <u>RMS Center Coordinate is:</u> A key piece of information needed to calculate performance metrics for the Tracking subtask. This field represents the X- and Y-axis position values of the reticle in order for it to appear like it is in the center of the subtask window. This is the position that is used for all metric calculations, and differs from the <u>Circle Center</u>, which will be discussed later.
- b. <u>Tracking Box X-Axis Lower Limit is:</u> A key piece of information needed to calculate performance metrics for the Tracking subtask when configured with the square outline. This field represents the X-axis value of the left-most border of the outline.
- c. <u>Tracking Box X-Axis Upper Limit is:</u> A key piece of information needed to calculate performance metrics for the Tracking subtask when configured with the square outline. This field represents the X-axis value of the right-most border of the outline.
- d. <u>Tracking Box Y-Axis Lower Limit is:</u> A key piece of information needed to calculate performance metrics for the Tracking subtask when configured with the square outline. This field represents the Y-axis value of the bottom border of the outline. Please note that this is because MATLAB specifies figure origins as the bottom-left corner of the figure.
- e. <u>Tracking Box Y-Axis Upper Limit is:</u> A key piece of information needed to calculate performance metrics for the Tracking subtask when configured with the square outline.

This field represents the Y-axis value of the top border of the outline. Please note that this is because MATLAB specifies figure origins as the bottom-left corner of the figure.

- f. <u>Circle Center:</u> This field represents the X- and Y-axis position of the center crosshair in the Tracking subtask and also the center of the circle when the Tracking subtask's outline is configured for a circle. This point is used only to generate the circle and is not directly used for any performance metric calculation.
- g. <u>Tracking Circle Radius</u>: Measured in pixels, this field represents the radius of the circle and is a key piece of information needed to calculate performance metrics for this subtask when the Tracking subtask outline is configured for a circle.
- h. <u>Theta:</u> This field represents the range of angles used to construct a circle. This information is not configurable and is not needed to calculate performance metrics.
- i. <u>Circle X-Axis Positions:</u> This field represents the formula used to calculate the X-axis positions of all points in the circle.
- j. <u>Circle Y-Axis Positions:</u> This field represents the formula used to calculate the Y-axis positions of all points in the circle.
- k. <u>Event Time:</u> Time at which the reticle position was sampled.
- 1. <u>X Coordinates:</u> X-axis position of the reticle.
- m. <u>Y Coordinates:</u> Y-axis position of the reticle.
- n. <u>X-Axis Difference:</u> Used in calculating performance metrics for this subtask. This value is defined as

X-Axis Difference = (X-Axis Reticle Position - X-Axis RMS Center Coordinate)

o. <u>Y-Axis Difference:</u> Used in calculating performance metrics for this subtask. This value is defined as

Y-Axis Difference = (Y-Axis Reticle Position - Y-Axis RMS Center Coordinate)

p. <u>X-Axis Deviation</u>: Used in calculating performance metrics for this subtask. This value is defined as

X-Axis Deviation = 
$$X$$
-Axis Difference<sup>2</sup>

q. <u>Y-Axis Deviation:</u> Used in calculating performance metrics for this subtask. This value is defined as

r. <u>Euclidian Distance From Target:</u> By treating Tank A and Tank B as two dimensions, the Euclidian distance of both tank volumes from their target value can be computed. This value is defined as

Euc. Dist. From Target = 
$$\sqrt{(X - Axis Deviation + Y - Axis Deviation)}$$

s. <u>X-Axis In Range?</u>: Boolean value used in calculating performance metrics that indicates whether the reticle is within the X-axis range specified at the top of the file. Values are 0

or 1, indicating whether the X-axis position is out of range or within range, respectively. This field is used only when the Tracking subtask's outline is configured for a square.

- t. <u>Y-Axis In Range?</u>: Boolean value used in calculating performance metrics that indicates whether the reticle is within the Y-axis range specified at the top of the file. Values are 0 or 1, indicating whether the Y-axis position is out of range or within range, respectively. This field is used only when the Tracking subtask's outline is configured for a square.
- u. <u>Both In Range?</u>: Boolean value that indicates whether the X-axis and Y-axis values of the reticle are within the range specified at the top of the file. Values are 0 or 1, indicating whether one or both axes are out of range, or whether they're both within range, respectively. This field is used when the Tracking subtask's outline is configured for either a square or circle.

# 7.3.2.13. Transition Log

This log contains a record of all of the transitions that occur in a trial, as well as the participant's indication of when those transitions occurred. Recall that this log is located in the root performance directory and contains information for all conditions in a given trial.

Fields contained in this log include:

- a. Event Time: Time at which either a participant action or task action occurred.
- b. <u>Perceived or Actual Transition</u>: This field denotes whether the logged event was either a task action, such as the transition between one condition and another, or a participant action, such as the participant reporting that a transition was perceived.
- c. <u>Previous Condition</u>: This field contains the name of the previous condition in a transition as it was named in the AF-MATB Script Generator Utility. For more information on condition names, please see section **9.2.2.1 Condition Name**.
- d. <u>Current Condition</u>: This field contains the name of the current condition in a transition as it was named in the AF-MATB Script Generator Utility.

# 7.3.3. Performance Summary Details and Descriptions

In addition to the log files generated by AF-MATB, a number of performance summary files are also generated depending on the task's configuration. These files, also formatted as .res files, allow researchers to quickly see a participant's performance without having to access the log files.

The following sections will detail each of the summaries generated by the task. Please note that as previously discussed in section **7.3.1 Performance Directory Types**, performance summaries are generated for each condition in a trial. For trials that contain multiple conditions, an aggregate summary that details participant performance is also generated.

## 7.3.3.1. Standard Performance Summary

The Standard Performance Summary provides participant performance from all four of the AF-MATB subtasks and is the only one of the four summaries generated regardless of the task configuration. The following sections describe the metrics used for each subtask and how they're calculated.

## 7.3.3.1.1. System Monitoring Section

Performance metrics for the System Monitoring subtask include the following:

- a. <u>Event Occurrences:</u> Number of malfunctions scheduled by the script.
- b. <u>Correct Responses:</u> Number of malfunctions correctly identified by the participant.
- c. Event Timeouts: Number of malfunctions the participant failed to correctly identify
- d. <u>False Alarms:</u> Number of participant responses that were not tied to an actual malfunction scheduled by the script.
- e. <u>Mean Correct Response Time:</u> Average time it took a participant to correctly identify a scheduled malfunction.
- f. <u>StDev Correct Response Time:</u> Standard deviation of response times to scheduled malfunctions that were answered correctly.

These metrics are calculated on three different levels; the object level, component level, and subtask level.

- a. <u>Object Level</u>: Performance values for each object in the subtask, i.e. Gauge 1, Light 1, etc. are reported.
- b. <u>Component Level</u>: Performance values for each component of the subtask. The columns of performance data with the headings "All Gauges" and "All Lights" denote the combination of each of the four gauges and two lights into their respective component groups.
- c. <u>Subtask Level:</u> Performance values for both components of the subtask. The "Total System" column details a participant's overall performance on the entire subtask.

# 7.3.3.1.2. Tracking Section

In section **7.3.2.12 Tracking Log**, all of the fields contained in the Tracking Log were detailed. The fields from that section will be used to explain how metrics are calculated for the Tracking subtask.

Please note that unless otherwise specified, every data sample acquired throughout a trial or condition is used in computing these metrics. For the metrics that use an independent parameter, known as an RMS Interval, to govern sampling rate, the number of samples can be computed using the equation

$$samples = \frac{time \ in \ seconds}{RMS \ Interval}$$

where *time in seconds* refers to the length of the condition or trial, depending on whether the performance summary is located in a "Condition" folder or the root performance directory, respectively. For more information on the parameter that governs this sampling rate, please see section **8.2.4.2 Tracking RMS Interval (Seconds)**.

Performance metrics for the Tracking subtask include the following:

a. <u>Target RMSD (Euclidian Distance)</u>: This metric describes the participant's ability to maintain the reticle's position at or around the center crosshair of the Tracking subtask. This metric is one of the metrics that is computed using data sampled at an independent rate, which was designed to reduce some of the noise in participant scores. This metric is defined as

$$Target RMSD (Euc.Dist.) = \sqrt{\frac{\sum_{i=1}^{samples} (Euclidian Distance From Target)^2}{samples}}$$

b. <u>X-Axis Target RMSD:</u> This metric describes the participant's lateral deviation of the reticle relative to the X-axis position of the center crosshair. This metric is one of the metrics that is computed using data sampled at an independent rate. This metric is defined as

X-Axis Target RMSD = 
$$\sqrt{\frac{\sum_{i=1}^{samples} X - Axis Deviation}{samples}}$$

c. <u>Y-Axis Target RMSD:</u> This metric describes the participant's vertical deviation of the reticle relative to the Y-axis position of the center crosshair. This metric is one of the metrics that is computed using data sampled at an independent rate. This metric is defined as

$$Y-Axis Target RMSD = \sqrt{\frac{\sum_{i=1}^{samples} Y-Axis Deviation}{samples}}$$

d. <u>Target RMSD (Summed)</u>: In prior releases of AF-MATB, this was the only metric calculated for the Tracking subtask. While the <u>Target RMSD (Euclidian Distance)</u> metric provides a better picture of how well participants control the reticle, it was important to preserve legacy metrics for the sake of transparency and to avoid confusion. This metric is defined as

- e. <u>%Time Inside Range</u>: This metric was designed to provide a snapshot of a participant's performance in terms of their percentage of success in keeping the reticle inside the Tracking subtask outline, configured as either a circle or square by the researcher.
- f. <u>Outside RMSD (Euclidian Distance)</u>: This metric describes the degree to which a participant was unable to keep the reticle inside the Tracking subtask outline. Due to the different options in configuring this subtask, different means must be used to compute the RMSD value. When the Tracking outline is configured as a circle, the following equation can be used to calculate this metric:

Outside RMSD (Euc. Dist. for Circle)

$$= \begin{cases} \sqrt{\frac{\sum_{i=1}^{samples} (Euc.Dist.From Target - Circle Radius)^2}{samples}}, Euc.Dist. > Circle Radius \end{cases}$$

In order to calculate this metric when the outline is configured as a square, distance from the closest edge must be computed. For both the X- and Y-axis coordinate of the reticle, the following deviations are calculated, respectively:

#### **XDeviation**

 $= \begin{cases} (Reticle XPos. - LeftBorder XPos.)^2 & Reticle XPos. < Left Border XPos. \\ (Reticle XPos. - Right Border XPos.)^2, & Reticle XPos. > Right Border XPos. \\ 0, & Else \end{cases}$ 

#### YDeviation

 $= \begin{cases} (Reticle YPos. - Bottom Border YPos.)^2 & Reticle YPos. < Bottom Border YPos. \\ (Reticle YPos. - Top Border YPos.)^2, & Reticle YPos. > Top Border YPos. \\ 0, & Else \end{cases}$ 

If the value of either the *XDeviation* and/or *YDeviation* score is not 0, indicating that the reticle was out of range in at least one dimension, then the Euclidian distance from the nearest outside edge of the Tracking outline is computed using the following equation:

*Euc. Dist. From Outside Edge* =  $\sqrt{(XDeviation + YDeviation)}$ 

Finally, the RMSD metric is computed for a square by the task using the following equation:

Outside RMSD (Euc. Dist. for Square)  
= 
$$\sqrt{\frac{\sum_{i=1}^{samples} (Euc. Dist. From Outside Edge)^2}{samples}}$$

g. <u>X-Axis Outside RMSD:</u> This metric describes the degree to which a participant was unable to keep the reticle within the confines of the X-axis limits of the square Tracking outline. This metric is not calculated when a circle outline is used. Because of the left and right borders of the X-axis, construction of the deviation score requires consideration the reticle's position with regard to these borders. This is represented by the following equation:

#### XDevScore

$$= \begin{cases} (Reticle XPos. - Left Border XPos.)^2, Reticle XPos. < Left Border XPos. \\ (Reticle XPos. - Right Border XPos.)^2, Reticle XPos. > Right Border XPos. \end{cases}$$

Please note that all *Pos.* values in this equation refer to the X-axis coordinates of the term only. After calculating these deviation scores, an outside RMSD score can be computed using the following:

X-Axis Outside RMSD = 
$$\sqrt{\frac{\sum_{i=1}^{samples} XDevScore}{samples}}$$

h. <u>Y-Axis Outside RMSD:</u> This metric describes the degree to which a participant was unable to keep the reticle within the confines of the Y-axis limits of the square Tracking outline. This metric is not calculated when a circle outline is used. Because of the top and bottom borders of the Y-axis, construction of the deviation score requires consideration the reticle's position with regard to these borders. This is represented by the following equation:

### YDevScore

 $= \begin{cases} (Reticle YPos. -Bottom Border YPos.)^2, & Reticle YPos. < Bottom Border YPos. \\ (Reticle YPos. -Top Border YPos.)^2, & Reticle YPos. > Top Border YPos. \end{cases}$ 

Please note that all *Pos.* values in this equation refer to the Y-axis coordinates of the term only. After calculating these deviation scores, an outside RMSD score can be computed using the following:

Y-Axis Outside RMSD = 
$$\sqrt{\frac{\sum_{i=1}^{samples} YDevScore}{samples}}$$

i. <u>Outside RMSD (Summed)</u>: Similar to the <u>Target RMSD (Summed)</u> metric, this simply calculates the total error along both dimensions of the Tracking subtask. This metric is defined as

Outside RMSD (Summed) = X-Axis Outside RMSD + Y-Axis Outside RMSD

Like the <u>X-Axis Outside RMSD</u> and <u>Y-Axis Outside RMSD</u> metrics, like metric is calculated only when the Tracking subtask outline is configured as a square.

## 7.3.3.1.3. Resource Management Section

In section **7.3.2.10 Resource Management Log**, all of the fields contained in the Resource Management Log are detailed. The fields from that section will be used to explain how metrics are calculated for the Resource Management subtask.

Please note that unless otherwise specified, every data sample acquired during a trial or condition is used in computing these metrics. For the metrics that use an independent parameter, known as an RMS Interval, to govern sampling rate, the number of samples can be computed using the equation

 $samples = \frac{time \ in \ seconds}{RMS \ Interval}$ 

where *time in seconds* refers to the length of the condition or trial, depending on whether the performance summary is located in a "Condition" folder or the root performance directory, respectively. For more information on the parameter that governs this sampling rate, please see section **8.2.9.7 Resource Management RMS Interval (Seconds)**.

Performance metrics for the Tracking subtask include the following:

a. <u>Target RMSD (Euclidian Distance)</u>: This metric describes the participant's ability to maintain the volumes of Tanks A and B at or around the specified target volume. By treating Tank A and Tank B as separate axes, Euclidian distance can be used to assess how well the participant is maintaining both tanks. This is one of the metrics that is computed using data sampled at an independent rate and was designed to reduce some of the noise in participant scores. This metric is defined as

$$Target RMSD (Euc.Dist.) = \sqrt{\frac{\sum_{i=1}^{samples} (Euclidian Distance From Target)^2}{samples}}$$

b. <u>Tank A Target RMSD:</u> This metric describes the deviation of the volume of Tank A relative to its target volume. This is one of the metrics that is computed using data sampled at an independent rate. This metric is defined as

$$Tank \ A \ Target \ RMSD = \sqrt{\frac{\sum_{i=1}^{samples} Tank \ A \ Deviation}{samples}}$$

c. <u>Tank B Target RMSD</u>: This metric describes the deviation of the volume of Tank B relative to its target volume. This is one of the metrics that is computed using data sampled at an independent rate. This metric is defined as

$$Tank \ B \ Target \ RMSD = \sqrt{\frac{\sum_{i=1}^{samples} Tank \ B \ Deviation}{samples}}$$

d. <u>Target RMSD (Summed)</u>: In prior releases of AF-MATB, this was the only metric calculated for the Resource Management subtask. While the <u>Target RMSD (Euclidian Distance)</u> metric provides a better picture of how well participants control the tank volumes, was important to preserve legacy metrics for the sake of transparency and to avoid confusion. This metric is defined as

Target RMSD(Summed) = Tank A Target RMSD + Tank B Target RMSD

e. <u>%Time Inside Range</u>: This metric provides a snapshot of the participant's performance using their percentage of success at keeping both Tank A and Tank B's volumes within the <u>Main Tank Automation Upper</u> and <u>Lower Limits</u> configured by the researcher.

Distribution A: Approved for public release; distribution unlimited. 88 ABW Cleared 12/03/2014; 88ABW-2014-5667.

f. <u>Outside RMSD (Euclidian Distance)</u>: This metric describes the degree to which a participant was unable to keep the tank volumes inside the limits. In order to calculate this metric, the distance from the closet limit must first be computed. For both Tank A and Tank B's volume, the following deviations are calculated, respectively:

**TankADeviation** 

 $= \begin{cases} (Tank \ A \ Volume - \ Lower \ Limit)^2 & Tank \ A \ Volume. < Lower \ Limit \\ (Tank \ A \ Volume - \ Upper \ Limit)^2, & Tank \ A \ Volume. > Upper \ Limit \\ 0, & Else \end{cases}$ 

**TankBDeviation** 

 $=\begin{cases} (Tank \ B \ Volume - \ Lower \ Limit)^2 & Tank \ B \ Volume. < Lower \ Limit \\ (Tank \ B \ Volume - \ Upper \ Limit)^2, & Tank \ B \ Volume. > Upper \ Limit \\ 0, & Else \end{cases}$ 

If the value of either the *TankADeviation* and/or *TankBDeviation* score is not 0, indicating that at least one tank is indeed out of range, then the Euclidian distance from the outside nearest limit is computed using the following equation:

*Euc.Dist.From Outside Edge* =  $\sqrt{(TankADeviation + TankBDeviation)}$ 

Finally, the RMSD metric is computed for a square by the task using the following equation:

 $Outside \ RMSD \ (Euc. Dist.) = \sqrt{\frac{\sum_{i=1}^{samples} (Euc. Dist. From \ Outside \ Edge)^2}{samples}}$ 

g. <u>Tank A Outside RMSD:</u> This metric describes the degree to which a participant was unable to keep Tank A's volume within the limits. This is represented by the following equation:

 $ADevScore = \begin{cases} (Tank \ A \ Volume - Lower \ Limit)^2, \ Tank \ A \ Volume < Lower \ Limit)^2, \ Tank \ A \ Volume > Upper \ Limit)^2, \end{cases}$ 

After calculating these deviation scores, an outside RMSD score can be computing using the following:

$$Tank \ A \ Outside \ RMSD = \sqrt{\frac{\sum_{i=1}^{samples} ADevScore}{samples}}$$

h. <u>Tank B Outside RMSD:</u> This metric describes the degree to which a participant was unable to keep Tank A's volume within the limits. This is represented by the following equation:

$$BDevScore = \begin{cases} (Tank \ B \ Volume - Lower \ Limit)^2, \ Tank \ B \ Volume < Lower \ Limit \\ (Tank \ B \ Volume - Upper \ Limit)^2, \ Tank \ B \ Volume > Upper \ Limit \end{cases}$$

After calculating these deviation scores, an outside RMSD score can be computing using the following:

$$Tank \ B \ Outside \ RMSD = \sqrt{\frac{\sum_{i=1}^{samples} BDevScore}{samples}}$$

i. <u>Outside RMSD (Summed)</u>: Similar to the <u>Target RMSD (Summed)</u> metric, this simply calculates the total error along both dimensions of the Resource Management subtask. This metric is defined as

Outside RMSD (Summed) = Tank A Outside RMSD + Tank B Outside RMSD

### 7.3.3.1.4. Communications Section

Performance metrics for the Communications subtask include the following:

- a. <u>Total Communications:</u> Total number of CEs that occur.
- b. <u>True Communications:</u> Number of TCs that occur.
- c. <u>False Communications:</u> Number of FCs that occur.
- d. <u>Correct Responses:</u> Number of times a participant responded to a TC with the correct channel and frequency.
- e. <u>False Alarms:</u> Number of times a participant responded to a FC with the correct channel and frequency.
- f. <u>Event Timeouts:</u> Number of times a participant failed to respond to TCs within the allotted timeframe.
- g. <u>True Accuracy Errors:</u> Number of times a participant responds to a TC with either the correct channel or correct frequency, but not both.
- h. <u>False Accuracy Errors:</u> Number of times a participant responds to a FC with either the correct channel or correct frequency, but not both.
- i. <u>Unexplained Responses:</u> Number of times a participant responds to some CE that has not been accounted for, but did not respond with both the correct channel and frequency.
- j. <u>No-Event Responses:</u> Number of participant responses that could not be attributed to any CE.
- k. <u>Mean Correct Response Time:</u> Average time it took a participant to correctly respond to a TC.
- 1. <u>StDev Correct Response Time:</u> Standard deviation of response times of a participant when correctly responding to TCs.

As stated in section **7.3.2.1 Correct Comm Log**, response to the Communications subtask requires a scoring hierarchy that is used to help interpret participant responses. This hierarchy reflects the order of the fields of this section of the Standard Performance Summary, starting with <u>Correct Responses</u>. This means that as participants respond to CEs, the scoring algorithm will first check to see if the participant's response meets the conditions to be considered a

Correct Response, and will work its way down until the list until an the conditions for an appropriate scoring category are met.

#### 7.3.3.2. **System Automation Performance Summary**

The System Automation Performance Summary provides a summary of the participant's performance for the System Monitoring subtask when Automated Mode is engaged. Metrics contained in this section include:

- a. System Automation Events: Number of System Monitoring automation events that occurred. This number is the sum of gauge malfunctions detected by the automation and gauge malfunctions the automation failed to detect.
- b. System Automation Failures: Number of gauge malfunctions that the automation failed to detect.
- c. Correct Responses: Number of times a participant correctly identified that the automation failed to detect a gauge malfunction.
- d. <u>Response Timeouts:</u> Number of times a participant failed to identify that the automation failed to detect a gauge malfunction.
- e. Response False Alarms: Number of times a participant incorrectly identified that the automation failed to detect a gauge malfunction.
- f. Mean Correct Response Time: Average time it took a participant to correctly identify that the automation failed to detect a gauge malfunction.
- g. StDev Correct Response Time: Standard deviation of response times of a participant when correctly identifying that the automation failed to detect a gauge malfunction.

#### 7.3.3.3. **Resource Automation Performance Summary**

The Resource Automation Performance Summary provides a summary of the participant's performance for the Resource Management subtask when Automated Mode is engaged using Automation Algorithm 2. Metrics contained in this section include:

- a. <u>Resource Automation Events:</u> Number of Resource Management automation malfunctions that occurred.
- b. Correct Responses: Number of times a participant correctly identified an automation malfunction.
- c. Response Timeouts: Number of times a participant failed to identify an automation malfunction.
- d. <u>Response False Alarms:</u> Number of times a participant incorrectly identified an automation malfunction.
- e. <u>Mean Correct Response Time:</u> Average time it took a participant to correctly identify an automation malfunction.
- f. StDev Correct Response Time: Standard deviation of response times of a participant when correctly identifying an automation malfunction.

## 7.3.3.4. Information Throughput Summary

The Information Throughput Summary provides participant performance values specific to the HMI component of the HOIM. This summary is provided when researchers configure the task to enable IT Mode. Please note that a certain configuration is required in order to ensure the validity of information. For more information on this, please refer to the work by Phillips & Walters (2013).

The fields contained in this summary required for computing the <u>Input Baud Rates</u> and <u>Fractional Accuracies</u>:

- a. <u>Time (seconds)</u>: The length of the condition or trial, which is required in order to compute the <u>Input Baud Rates</u> discussed later in this section.
- b. <u>Resource Management Automation Lower Limit (Liters)</u>: This field represents the minimum acceptable volume that Tank A or Tank B must be at to be considered within range. Please note that this parameter is required for this model, despite the fact that the Resource Management subtask's Automated Modes are locked-out for this task configuration.
- c. <u>Resource Management Automation Upper Limit (Liters)</u>: This field represents the maximum acceptable volume that Tank A or Tank B must be at to be considered within range. Please note that this parameter is required for this model, despite the fact that the Resource Management subtask's Automated Modes are locked-out for this task configuration.
- d. <u>Resource Management Net Flow Rate:</u> This describes the net flow rate for the Resource Management subtask. This equation for this value is:

Net Flow Rate = Single Pump Fill Rate - Single Tank Drop Rate

- e. <u>Tracking Outline Circle Radius (Pixels)</u>: The radius of the circular Tracking outline.
- f. <u>Tracking Subtask Average Velocity (pixels/second)</u>: The average velocity of the Tracking reticle's movement. Despite the researcher being able to configure the speed of the reticle, some variability is inherent due to the nature of this subtask. As a result, this value is actually empirically derived from all samples in a given condition or trial to ensure the most accurate data.

The second section contained in this summary computes the <u>Input Baud Rates</u> and <u>Fractional</u> <u>Accuracy</u> values. Both of these values are computed for the following components of the task:

- a. <u>Gauges</u>
- b. Lights
- c. <u>Tracking</u>
- d. <u>Resource Management</u>
- e. Communications

For more information on <u>Input Baud Rates</u> and <u>Fractional Accuracies</u>, including how they are computed, please see Phillips et al. (2007), Phillips et al. (2007), Phillips & Walters (2013), Phillips, Walters, & McKinley (2013), and Walters (2012).

## 7.3.4. DataStructure Details and Descriptions

The only file generated by AF-MATB that does not carry a .res extension is the DataStructure file, which carries a .mat extension. This file is formatted as a MATLAB data file and can only be read using MATLAB or custom software applications that support importing .mat files.

This file contains all of the information discussed in the previous sections, including all log files and performance summaries, the script that was loaded, and some other basic trial and configuration information that could be useful in post-processing (using MATLAB). The ability to post-process was the exact purpose of this file, as researchers can load this file and instantly be able to manipulate data without worrying about reading in the text from the log files.

# 8.0 AF-MATB CONFIGURATION UTILITY

# 8.1. Utility Introduction

The Configuration Utility is one of the key components differentiating our software package from other MATB packages. Titled the "Parameters utility" in our previous release, this utility received a new name and a major redesign. Now, the Configuration Utility is segmented into different groups of parameters, thus making task configuration easier.

The following sections will detail all of the functionality in this utility. Users will be provided with descriptions for each parameter in this utility, as well as acceptable values for that parameter and its default value.

# 8.2. Parameter Groups

When the Configuration Utility is launched, the user is presented with an interface (see Figure 101) consisting of 7 clickable buttons and one dropdown menu. This dropdown menu is titled "Select a Parameter Group" and will be referred to from here on out as the Group Dropdown. The Group Dropdown is the key object used for navigating this utility. For more information on the clickable buttons, please see section **8.3 Main Function** Buttons.



Figure 101 – The appearance of the AF-MATB Configuration Utility when launched.

#### 8.2.1. Navigating The Group Dropdown

Clicking on the Group Dropdown reveals a list of 10 different groups of parameters the user can select (see Figure 102). Clicking on any of these 10 groups renders a list of all of the parameters for that group (see Figure 103). Exiting the Group Dropdown before making a selection will preserve the previous state of the utility.



Figure 102 - Ten parameter groups are available to select from the dropdown menu in the AF-MATB Configuration Utility.

<b>3</b>	AF_MATB_Con	figurationUtility_V	/300		- • ×
AF-MATB System Param V					
	AF-MATB System Parameters				
	Input Options	GUI Buttons Only Keyboard Only Both GUI & Keyboard	Default Value		
	Enable Pausing?	• Yes O No	Default Value		
	Enable Manual Event Triggering?		Default Value		
	Enable Manual Pump Repair?	• Yes No	Default Value		
	Display Date & Time?	Yes  No	Default Value		
	Enable Scheduling?	• Yes O No	Default Value		
	Enable Escape key?	• Yes No	Default Value		Save Values
	Enable Home key?	Yes  No	Default Value		
	Timer Cycling Rate (Hz)	10	Default Value		Load Config File
	Intial Ramp-up Time Delay (Seconds)	5	Default Value		Set All Default Values
	NASA-TLX Ramp-Up Time Delay (Seconds)	5	Default Value	s	ave and Continue to Script Generator
	Resource Management Automation Algorithm	Algorithm 1    Algorithm 2	Default Value	L	oad and Continue to Script Generator
	Enable Subtask Automation Indicators?	Green V	Default Value		Save and Continue to AF-MATB
	Enable Information Throughout Mode?	Yes No	Default Value		Load and Continue to AF-MATB
		0 0 1			V. 3.00

Figure 103 – The appearance of the AF-MATB Configuration Utility after the "AF-MATB System Parameters" selection was selected from the Group Dropdown.

Each time a new parameter group is selected from the Group Dropdown, either to go to a new parameter group or to return the utility to its initial appearance (see Figure 101) by selecting the "Select a Parameter Group" field, the utility automatically verifies the validity of all parameters in the current group before opening the new group. As shown in Figure 104, if the user enters an invalid value for any parameter, and/or forgets to enter a value for any parameter, and then attempts to select another parameter group via the Group Dropdown, the utility will display a dialog list on the left side of the window, detailing the changes that need to be made before the user can continue (see Figure 105). Additionally, the parameters with errors will turn red to highlight the problems.

Once all problems for that group have been remedied, if the user makes a selection via the Group Dropdown, the command will be executed. If the user clicks on the same parameter group that is currently open, they'll notice that the text labels will return to their original color and the error dialog list will close (see Figure 103).

2	AF_MATB_Con	figurationUtility_V	300		- 🗆 💌
AF-MATB System Param V					
	AF-MATB System Parameters				
	Input Options	GUI Buttons Only Keyboard Only Both GUI & Keyboard	Default Value		
	Enable Pausing?	• Yes O No	Default Value		
	Enable Manual Event Triggering?	• Yes O No	Default Value		
	Enable Manual Pump Repair?	• Yes O No	Default Value		
	Display Date & Time?	• Yes • No	Default Value		
Please enter a number from 10 to 20 for the Timer	Enable Scheduling?	● Yes ○ No	Default Value		
Please enter a number greater than 0 for the Task's	Enable Escape key?	Yes     No	Default Value		Save Values
	Enable Home key?	Yes  No	Default Value		
	Timer Cycling Rate (Hz)	9898	Default Value		Load Config File
	Intial Ramp-up Time Delay (Seconds)	5	Default Value		Set All Default Values
	NASA-TLX Ramp-Up Time Delay (Seconds)		Default Value	S	ave and Continue to Script Generator
×					
	Resource Management Automation Algorithm	Algorithm 1	Default Value	L	oad and Continue to Script Generator
		Algorithm 2			Save and Continue to AF-MATB
	Enable Subtask Automation Indicators?	Green 🗸	Default Value		
	Enable Information Throughput Mode?	○ Yes	Default Value		Load and Continue to AF-MATB
					V. 3.00

Figure 104 – The user made an invalid entry in the field of one parameter, and omitted a value for the field of another parameter. Note the appearance of the error dialog list left side of the window.



Figure 105 – Close-up of the error dialog list, illustrating the problems for that parameter group, as well as the acceptable values for that group.

# 8.2.2. AF-MATB System Parameters Group

In the AF-MATB Systems Parameters group (see Figure 106), the user has access to overall system functionality and task configuration options. These options range from the task input methods to options for task appearance.

Please note that every parameter in this group has an associated *Default Value* button that will reset that parameter to its default value. These *Default Value* buttons are present in every parameter group in the Configuration Utility.

AF-MATB System Parameters		
Input Options	GUI Buttons Only    Keyboard Only    Both GUI & Keyboard	Default Value
Enable Pausing?	• Yes • No	Default Value
Enable Manual Event Triggering?	Yes     No	Default Value
Enable Manual Pump Repair?	• Yes • No	Default Value
Display Date & Time?	Yes     No	Default Value
Enable Scheduling?	Yes     No	Default Value
Enable Escape key?	Yes     No	Default Value
Enable Home key?	Yes     No	Default Value
Timer Cycling Rate (Hz)	10	Default Value
Intial Ramp-up Time Delay (Seconds)	5	Default Value
NASA-TLX Ramp-Up Time Delay (Seconds)	5	Default Value
Resource Management Automation Algorithm	Algorithm 1    Algorithm 2	Default Value
Enable Subtask Automation Indicators?	Green 🗸	Default Value
Enable Information Throughput Mode?	○ Yes	Default Value

Figure 106 – Options available in the AF-MATB System Parameters group in the AF-MATB Configuration Utility.

#### 8.2.2.1. Input Options

This parameter controls the task's user input settings. When this parameter is set to "Both GUI & Keyboard," the participant will be able to respond to task events using either the keyboard commands outlined in sections **6.1 System Monitoring Commands**, **6.2 Resource Management Response Commands**, and **6.3 Communications Response Commands**, or the GUI buttons discussed in those sections and illustrated in Figure 048, Figure 049, Figure 050, Figure 051, and Figure 052. When this parameter is set to "GUI Buttons Only," the participant must use the GUI buttons to respond to task events, as the associated keyboard commands will be disabled. Conversely, when this parameter is set to "Keyboard Only," the participant must use the keyboard commands to respond to task events, as the associated GUI buttons will be disabled. Please note that this parameter will only affect the sections explicitly mentioned above, and that certain commands and GUI buttons may not be available based on how other parameters are configured.

The default value for this parameter is "Both GUI and Keyboard."

Appropriate values for this parameter are "GUI Buttons Only," "Keyboard Only," or "Both GUI and Keyboard."



Figure 107 – Input Options parameter in the AF-MATB System Parameters group.

Please note that this parameter is one of only three parameters that, when configured in a certain manner, will disable the ability to configure other options throughout the Configuration Utility. If the "GUI Buttons Only" radio button is selected, users will be notified that this selection will also reconfigure other parameters throughout the utility (see Figure 124). For more information on this behavior, please see section **8.2.5.9 Frequency Entry Method**.



Figure 108 – Warning message displayed when users click "GUI Buttons Only" to the Input Options parameter

#### 8.2.2.2. Enable Pausing?

This parameter controls the ability to pause the task. When this parameter is set to "No," the *Pause/Break* command discussed in section **6.4 System Commands** will be disabled.

The default value for this parameter is "Yes"

Appropriate values for this parameter are "Yes" or "No."

Enable Pausing?	Enable Pausing?	• Yes	N₀	Default Value
-----------------	-----------------	-------	----	---------------

Figure 109 – Enable Pausing? parameter in the AF-MATB System Parameters group.

## 8.2.2.3. Enable Manual Event Triggering?

This parameter controls the user's ability to execute manual commands for the individual subtasks. When this parameter is set to "No," all keyboard commands in sections **6.5.2 System Management**, **6.5.3.1.1 Malfunction Commands**, **6.5.3.2 Automation Algorithm 2** Commands, **6.5.3.3 Other Commands**, **6.5.4 Tracking**, **6.5.5 Communications**, and **6.5.6** Task Component Visibility Commands, will be disabled. Please note that this parameter does not affect the functionality of commands in section **6.5.3.1.2 Timeout Commands**, as these are controlled by the following parameter (section **8.2.2.4 Enable Manual Pump Repair?**).

The default value for this parameter is "Yes."

Appropriate values for this parameter are "Yes" or "No."

Enable Manual Event Triggering?	• Yes • No	Default Value
---------------------------------	------------	---------------

Figure 110 – Enable Manual Event Triggering? parameter in the AF-MATB System Parameters group.

## 8.2.2.4. Enable Manual Pump Repair?

This parameter controls the user's ability to manually repair pump failures. When this parameter is set to "No," all keyboard commands in section **6.5.3.1.2 Timeout Commands** will be disabled.

The default value for this parameter is "Yes."

Appropriate values for this parameter are "Yes," or "No."

Enable Manual Pump Repair?	• Yes	⊖ No	Default Value
----------------------------	-------	------	---------------

Figure 111 – Enable Manual Pump Repair? parameter in the AF-MATB System Parameters group.

Please note that in the event that IT Mode is enabled (see section **8.2.2.14 Enable Information Throughput Mode?**), this parameter will automatically be set to "No" and cannot be modified until IT Mode is disabled (see Figure 112).

Enable Manual Pump Repair?	O Yes	No	Default Value
----------------------------	-------	----	---------------

Figure 112 – How the Enable Manual Pump Repair? parameter appears when the Enable IT Mode? parameter has been set to "Yes."

#### 8.2.2.5. Display Date and Time?

This parameter determines whether the date and the program run-time (i.e. time elapsed since the trial started) are displayed.

The default value for this parameter is "Yes."

Appropriate values for this parameter are "Yes" or "No."

Figure 113 – Display Date and Time? parameter in the AF-MATB System Parameters group.

## 8.2.2.6. Enable Scheduling?

This parameter controls the Scheduling window in the AF-MATB.

The default value for this parameter is "Yes."

Appropriate values for this parameter are "Yes" or "No."

Enable Scheduling?	• Yes	⊖ No	Default V	alue
--------------------	-------	------	-----------	------

 $Figure \ 114-Enable \ Scheduling? \ parameter \ in \ the \ AF-MATB \ System \ Parameters \ group.$ 

#### 8.2.2.7. Enable Escape Key?

This parameter controls the functionality of the *Esc* key in the AF-MATB. This functionality was implemented to ensure that participants could not accidentally end trials during testing sessions. When this parameter is set to "No," the *Esc (Escape)* command discussed in section **6.4 System Commands** will be disabled.

The default value for this parameter is "Yes."

Appropriate values for this parameter are "Yes" or "No."

Enable Escape key?	• Yes	⊖ No		Default Value
--------------------	-------	------	--	---------------

Figure 115 – Enable Escape key? parameter in the AF-MATB System Parameters group.

## 8.2.2.8. Enable Home Key?

This parameter controls the functionality of the *Home* key in the AF-MATB. This functionality was implemented to ensure that participants could not accidentally interrupt trials during testing sessions. When this parameter is set to "No," the *Home* command discussed in section **6.4 System Commands** will be disabled.

The default value for this parameter is "Yes."

Appropriate values for this parameter are "Yes" or "No."

Enable Home key?	• Yes	⊖ No	Default Value
------------------	-------	------	---------------

Figure 116 – Enable Home key? parameter in the AF-MATB System Parameters group.

### 8.2.2.9. Timer Cycling Rate (Hz)

This parameter determines how fast the task refreshes during execution. Please note that while this parameter is configurable within a limited range, it is highly recommended that this value not be modified, as it may result in system instability.

The default value for this parameter is 10.

Appropriate values for this parameter are 10 to 20.

Timer Cycling Rate (Hz)	10	Default Value
-------------------------	----	---------------

Figure 117 – Timer Cycling Rate (Hz) parameter in the AF-MATB System Parameters group.

## 8.2.2.10. Initial Ramp-up Time Delay (Seconds)

This parameter determines the length of the delay between the user initializing the trial by pressing *Space Bar* and the task actually beginning. The ability to adjust this delay is helpful when the researcher is synchronizing multiple systems.

The default value for this parameter is 5.

Appropriate values for this parameter are real numbers greater than 0.

Figure 118 – Initial Ramp-up Time Delay (Seconds) parameter in the AF-MATB System Parameters group.

## 8.2.2.11. NASA-TLX Ramp-Up Time Delay (Seconds)

If the NASA-TLX is administered, this parameter specifies how quickly the AF-MATB will resume after the user completes the NASA-TLX.

The default value for this parameter is 5.

Appropriate values for this parameter are real numbers greater than 0.

NASA-TLX Ramp-Up Time Delay (Seconds)	5	Default Value

 $Figure \ 119-{\sf NASA-TLX} \ Ramp-Up \ Time \ Delay \ (Seconds) \ parameter \ in \ the \ AF-MATB \ System \ Parameters \ group.$ 

# 8.2.2.12. Resource Management Automation Algorithm

This parameter controls which of the two automation algorithms the Resource Management subtask will use when in Automated Mode.

The default value for this parameter is "Algorithm 1."

Appropriate values for this parameter are "Algorithm 1" or "Algorithm 2."



Figure 120 – Resource Management Automation Algorithm parameter in the AF-MATB System Parameters group.

Please note that if the IT Mode is enabled (see section **8.2.2.14 Enable Information Throughput Mode?**), this parameter will automatically be set to "Algorithm 1" and it cannot be modified until IT Mode is disabled (see Figure 121).



Figure 121 – How the Resource Management Automation Algorithm parameter appears when the Enable IT Mode? parameter has been set to "Yes."

Also please note that this parameter is one of only three parameters that, when configured in a certain manner, will disable the ability to configure other options throughout the Configuration Utility. If the "Algorithm 2" radio button is selected, the utility will automatically set the Hide Automation Behavior? parameter in the Resource Management Subtask Automation Parameters group to "Yes" and the ability to configure it will be disabled. For more information, please see section **8.2.9.9 Hide Automation Behavior**.

### 8.2.2.13. Enable Subtask Automation Indicators?

This parameter specifies which color the System Monitoring, Tracking, or Resource Management subtask banners will be when their respective Automated Modes are enabled. This was designed to be an overt indication for the user that the automation in that subtask is active.

The default value of this parameter is "Green."

Appropriate values for this parameter are "None," "Green," "Blue," "Yellow," or "Orange."

Enable Subtask Automation Indicators?	Green	¥	Default Value

Figure 122 – Enable Subtask Automation Indicators? parameter in the AF-MATB System Parameters group.

# 8.2.2.14. Enable Information Throughput Mode?

This parameter configures the task to operate in IT Mode.

The default value for this parameter is "No."

Appropriate values for this parameter are "Yes" or "No."



 $Figure \ 123- {\it Enable Information Throughput Mode? parameter in the AF-MATB System Parameters group.}$ 

Please note that this parameter is one of only three parameters that, when configured in a certain manner, will disable the ability to configure other options throughout the Configuration Utility. If the "Yes" radio button is selected, users will be notified that this selection will also reconfigure other parameters throughout the utility (see Figure 124).



Figure 124 – Warning message displayed when users click "Yes" to the Enable Information Throughput Mode? parameter.

If the user clicks "No", the radio button will return to its default "No" state. However, if the user clicks "Yes", the utility will automatically set several parameters to their designated values, as well as disable the ability to configure those parameters. For more information on this behavior, please see sections **8.2.2.4 Enable Manual Pump Repair?**, **8.2.2.12 Resource Management Automation Algorithm**, **8.2.4.7 Tracking Range Outline Shape**, and **8.2.4.8 Circle Tracking Outline Radius (Pixels)**.

## 8.2.3. System Monitoring Subtask Basic Parameters Group

In the System Monitoring Subtask Parameters group, the user can configure parameters specific to the System Monitoring subtask (see Figure 125). It should be noted that these parameters are reserved for operation of the subtask when in Manual Mode. For parameters that govern the behavior of this subtask when in Automated Mode, see section **8.2.7 System Monitoring Subtask Automation Parameters Group**.

System Monitoring Subtask Basic Parameters				
Gauge Speed Lower Limit (Pixels/Cycle)	2	Default Value		
Gauge Speed Upper Limit (Pixels/Cycle)	4	Default Value		
End of Range Delay Max (Cycles)	10	Default Value		
Correct Fault Identification Pause (Cycles)	10	Default Value		
Gauge Malfunction Timeout (Seconds)	10	Default Value		
Indicator Light Malfunction Timeout (Seconds)	5	Default Value		

Figure 125 – Options available in the System Monitoring Subtask Basic Parameters group in the AF-MATB Configuration Utility.

All parameters in this section, except for the Gauge Malfunction Timeout (Seconds) and Indicator Light Malfunction Timeout (Seconds) parameters, are influenced by the cycling rate of the task discussed in section **8.2.2.9 Timer Cycling Rate (Hz)**. If the cycling rate of the task is modified, the necessary parameters in this section should also be modified accordingly.

## 8.2.3.1. Gauge Speed Lower Limit (Pixels/Cycle)

This parameter determines the minimum speed of the indicator for the Gauges component of the System Monitoring subtask.

The default value for this parameter is 2.

Appropriate values for this parameter are real numbers greater than 0.

Gauge Speed Lower Limit (Pixels/Cycle)	2	Default Value
----------------------------------------	---	---------------

Figure 126 - Gauge Speed Lower Limit (Pixels/Cycle) parameter in the System Monitoring Subtask Basic Parameters group.

## 8.2.3.2. Gauge Speed Upper Limit (Pixels/Cycle)

This parameter determines the minimum speed of the indicator for the Gauges component of the System Monitoring subtask. The speed of the indicators is generated from a random distribution ranging from the Gauge Speed Lower and Upper Limit parameters. The greater the difference between these two values, the greater the variability in speeds. The closer the distribution's mean is to 0, the slower the indicators will travel, while the closer the distribution's mean is to infinity, the faster the indicators will travel.

The default value for this parameter is 4.

Appropriate values for this parameter are real numbers greater than the Gauge Speed Lower Limit (Pixels/Cycle) parameter discussed in the previous section



Figure 127 – Gauge Speed Lower Limit (Pixels/Cycle) parameter in the System Monitoring Subtask Basic Parameters group.

# 8.2.3.3. End of Range Delay Max (Cycles)

This parameter determines the maximum amount of time a Gauge indicator will rest at an extreme of its range before reversing direction towards the other extreme. The duration of this delay is generated from a random distribution ranging from 0 to the value of this parameter. The greater the difference between these two values, the greater the variability in delays.

The default value for this parameter is 10.

Appropriate values for this parameter are integers greater than 0.

End of Range Delay Max (Cycles)	10	Default Value
---------------------------------	----	---------------

Figure 128 – End of Range Delay Max (Cycles) parameter in the System Monitoring Subtask Basic Parameters group.

# 8.2.3.4. Correct Fault Identification Pause (Cycles)

This parameter determines the amount of time a Gauge indicator will rest at the center after a participant has correctly identified a Gauge malfunction. This value also determines the display time for the yellow rectangle that appears at the bottom of a gauge after a malfunction is correctly identified.

The default value for this parameter is 10.

Appropriate values for this parameter are integers greater than 0.



Figure 129 - Correct Fault Identification Pause (Cycles) parameter in the System Monitoring Subtask Basic Parameters group.

Distribution A: Approved for public release; distribution unlimited. 88 ABW Cleared 12/03/2014; 88ABW-2014-5667.

## 8.2.3.5. Gauge Malfunction Timeout (Seconds)

This parameter determines the duration of a gauge malfunction before it is automatically restored to normal operation.

The default value for this parameter is 10.

Appropriate values for this parameter are real numbers greater than 0.

Gauge Malfunction Timeout (Seconds)	10	Default Value
-------------------------------------	----	---------------

Figure 130 – Gauge Malfunction Timeout (Seconds) parameter in the System Monitoring Subtask Basic Parameters group.

#### 8.2.3.6. Indicator Light Malfunction Timeout (Seconds)

This parameter determines the duration of a light malfunction before it is automatically restored to normal operation.

The default value for this parameter is 5.

Appropriate values for this parameter are real numbers greater than 0.

Indicator Light Malfunction Timeout (Seconds)	5	Default Value
-----------------------------------------------	---	---------------

Figure 131 – Indicator Light Malfunction Timeout (Seconds) in the System Monitoring Subtask Basic Parameters group.

## 8.2.4. Tracking Subtask Basic Parameters Group

In the Tracking Subtask Basic Parameters group, the user can configure parameters specific to the general function of the Tracking subtask (see Figure 132).

Tracking Subtask Basic Parameters		
Tracking Gain	65	Default Value
Tracking RMS Interval	5	Default Value
Center Range (Pixels)	8	Default Value
Vertical Spacing for Crosshairs (Pixels)	50	Default Value
Horizontal Spacing for Crosshair (Pixels)	50	Default Value
Horizontal Crosshair Count	5	Default Value
Tracking Range Outline Shape	Square    Circle	Default Value
Circle Tracking Outline Radius (Pixels)	150	Default Value
Square Tracking Outline Width (Pixels)	300	Default Value

Figure 132 – The Tracking Subtask Basic Parameters group in the AF-MATB Configuration Utility.

## 8.2.4.1. Tracking Gain

This parameter determines the sensitivity of the joystick for the Tracking Task. Please note that while this parameter is configurable, it is highly recommended that this value not be modified.

The default value for this parameter is 65.

Appropriate values for this parameter are real numbers greater than 40.



Figure 133 – Tracking Gain parameter in the Tracking Subtask Basic Parameters group.
# 8.2.4.2. Tracking RMS Interval (Seconds)

Recall in section **7.3.3.1.2 Tracking Section** that some measures are computed using an independent sampling rate that is lower that results in a subset of all position data stored in the Tracking Log. This parameter determines the interval at which data are stored from that subset in order to compute the <u>Target RMSD (Euc. Dist)</u>, <u>X-Axis Target RMSD</u>, and <u>Y-Axis Target RMSD</u> metrics. Operationally, this parameter means that after every specified interval, one data point is stored in a log used to compute the previously mentioned metrics.

The default value for this parameter is 5.

Appropriate values for this parameter are integers greater than 0.

Tracking RMS Interval	5	Default Value
-----------------------	---	---------------

Figure 134 – Tracking RMS Interval (Seconds) parameter in the Tracking Subtask Basic Parameters group.

# 8.2.4.3. Center Range (Pixels)

Used exclusively for the Tracking subtask's Automated Mode, this parameter defines a radius from the center of the subtask. This radius is used to establish an acceptable range for reticle movement. As this number approaches 0, the reticle will deviate less from the center of the subtask when in Automated Mode.

The default value for this parameter is 8.

Appropriate values for this parameter are real numbers greater than 0.

Center Range (Pixels)	8	Default Value
-----------------------	---	---------------

Figure 135 – Center Range (Pixels) parameter in the Tracking Subtask Basic Parameters group.

# 8.2.4.4. Vertical Spacing for Crosshairs (Pixels)

This parameter determines the distance in pixels between the vertical crosshairs.

The default value for this parameter is 50.

Appropriate values for this parameter are real numbers between 25 and 55.

Vertical Spacing for Crosshairs (Pixels)	50	Default Value
------------------------------------------	----	---------------

Figure 136 - Vertical Spacing for Crosshairs (Pixels) parameter in the Tracking Subtask Basic Parameters group.

# 8.2.4.5. Horizontal Spacing for Crosshairs (Pixels)

This parameter determines the distance in pixels between the horizontal crosshairs.

The default value for this parameter is 50.

Appropriate values for this parameter are real numbers between 25 and 55.



Figure 137 – Horizontal Spacing for Crosshairs (Pixels) parameter in the Tracking Subtask Basic Parameters group.

# 8.2.4.6. Horizontal Crosshair Count

This parameter determines the number of horizontal crosshairs that are displayed in the subtask window. This number represents the number of crosshairs present on each side of the center crosshair in the Tracking subtask.

The default value for this parameter is 5.

Appropriate values for this parameter range from 3 to 5.



Figure 138 – Horizontal Crosshair Count parameter in the Tracking Subtask Basic Parameters group.

# 8.2.4.7. Tracking Range Outline Shape

This parameter defines the shape of the Tracking range.

The default entry for this parameter is "Square".

Appropriate entries for this parameter are "Square" or "Circle".



Figure 139 – Tracking Range Outline Shape parameter in the Tracking Subtask Basic Parameters group.

Please note that if IT Mode is enabled (see section **8.2.2.14 Enable Information Throughput Mode?**), this parameter will automatically be set to "Circle" and cannot be modified until IT Mode is disabled (see Figure 140).



Figure 140 – The Tracking Range Outline Shape parameter when the Enable IT Mode? parameter has been set to "Yes."

# 8.2.4.8. Circle Tracking Outline Radius (Pixels)

This parameter defines the pixel radius of the tracking range when "Circle" is selected as the outline shape for the range.

The default value for this parameter is 150.

Appropriate values for this parameter are real numbers between 30 and 175.



 $Figure \ 141-Circle \ Tracking \ Outline \ Radius \ (Pixels) \ parameter \ in \ the \ Tracking \ Subtask \ Basic \ Parameters \ group.$ 

Please note that if IT Mode is enabled (see section **8.2.2.14 Enable Information Throughput Mode?**), this parameter will automatically be set to "49.5" and it cannot be modified until IT Mode is disabled (see Figure 142).



Figure 142 – How the Circle Tracking Outline Radius (Pixels) parameter appears when the Enable IT Mode? parameter has been set to "Yes."

# 8.2.4.9. Square Tracking Outline Width (Pixels)

This parameter defines the pixel radius of the tracking range when "Square" is selected as the outline shape for the range.

The default value for this parameter is 300.

Appropriate values for this parameter are real numbers between 40 and 350.

Square Tracking Outline Width (Pixels)	300	Default Value
----------------------------------------	-----	---------------

Figure 143 – Square Tracking Outline Width (Pixels) parameter in the Tracking Subtask Basic Parameters group.

## 8.2.5. Communications Subtask Parameters Group

In the Communications Subtask Parameters group, the user can configure parameters specific to the function of the Communications subtask (see Figure 144).

Users should note that for this group of parameters, only a select few parameters affect the functionality of the Communications subtask. Most of the parameters in this group affect only the appearance of the subtask and do not, in any way, change the duration or the content of the default audio files provided with the AF-MATB Package, or the scoring behavior of this subtask. Please refer to the documentation below for more information.

Communications Subtask Parameters		
Callsign Label	NGT504	Default Value
Comm Slot 1 Label	NAV1	Default Value
Comm Slot 2 Label	NAV2	Default Value
Comm Slot 3 Label	COM1	Default Value
Comm Slot 4 Label	COM2	Default Value
Enter Confirmation Highlight (Cycles)	12	Default Value
Target Communication Timeout (Seconds)	15	Default Value
Communications Event Duration (Seconds)	8	Default Value
Frequency Entry Method	Arrow Keys Number Pad	Default Value
Slot 1 & 2 Lower Limit	108.7	Default Value
Slot 1 & 2 Upper Limit	117.7	Default Value
Slot 1 & 2 Frequency Increments	0.2	Default Value
Slot 3 & 4 Lower Limit	117.9	Default Value
Slot 3 & 4 Upper Limit	126.9	Default Value
Slot 3 & 4 Frequency Increments	0.2	Default Value

Figure 144 – Options available in the Communications Subtask Parameters group in the AF-MATB Configuration Utility.

Distribution A: Approved for public release; distribution unlimited. 88 ABW Cleared 12/03/2014; 88ABW-2014-5667.

### 8.2.5.1. Callsign Label

This parameter determines the callsign name displayed next to the "Callsign" label in the Communications subtask. Note that changing this parameter will not modify the scoring behavior of the subtask. For example, changing this parameter to "NLS217" will not remap "NLS217" Communications Events as True Communications.

The default value for this parameter is "NGT504."

Appropriate values for this parameter are at least one character with a maximum of eight characters, though it is strongly recommended that this parameter not be modified.



Figure 145 – Callsign Label parameter in the Communications Subtask Parameters group.

#### 8.2.5.2. Comm Slot 1 Label

This parameter determines the label for Comm Slot 1. Note that changing this parameter will not modify the scoring of the subtask, as the scoring algorithm will still recognize this slot as "NAV1" when scoring Communications Events.

The default value for this parameter is "NAV1."

Appropriate values for this parameter are at least one character with a maximum of four characters, though it is strongly recommended that this parameter not be modified.

Comm Slot 1 Label	NAV1	Default Value
-------------------	------	---------------

Figure 146 – Comm Slot 1 Label parameter in the Communications Subtask Parameters group.

# 8.2.5.3. Comm Slot 2 Label

This parameter determines the label for Comm Slot 2. Note that changing this parameter will not modify the scoring of the subtask, as the scoring algorithm will still recognize this slot as "NAV2" when scoring Communications Events.

The default value for this parameter is "NAV2."

Appropriate values for this parameter are at least one character with a maximum of four characters, though it is strongly recommended that this parameter not be modified.

Comm Slot 2 Label	NAV2	Default Value
-------------------	------	---------------

Figure 147 - Comm Slot 2 Label parameter in the Communications Subtask Parameters group

# 8.2.5.4. Comm Slot 3 Label

This parameter determines the label for Comm Slot 3. Note that changing this parameter will not modify the scoring of the subtask, as the scoring algorithm will still recognize this slot as "COM1" when scoring Communications Events.

The default value for this parameter is "COM1."

Appropriate values for this parameter are at least one character with a maximum of four characters, though it is strongly recommended that this parameter not be modified.



 $Figure \ 148-Comm \ Slot \ 3 \ Label \ parameter \ in \ the \ Communications \ Subtask \ Parameters \ group.$ 

#### 8.2.5.5. Comm Slot 4 Label

This parameter determines the label for Comm Slot 4. Note that changing this parameter will not modify the scoring of the subtask, as the scoring algorithm will still recognize this slot as "COM2" when scoring Communications Events.

The default value for this parameter is "COM2."

Appropriate values for this parameter are at least one character with a maximum of four characters, though it is strongly recommended that this parameter not be modified.

Comm Slot 4 Label	COM2	Default Value
-------------------	------	---------------

Figure 149 – Comm Slot 4 Label parameter in the Communications Subtask Parameters group.

# 8.2.5.6. Enter Confirmation Highlight (Cycles)

This parameter determines the length of time the *Enter* button will stay green, signaling the participant that their response has been recorded. Please note this parameter is influenced by the cycling rate of the task discussed in section **8.2.2.9 Timer Cycling Rate (Hz)**. If the cycling rate of the task is modified, this parameter should also be modified accordingly.

The default value for this parameter is 12.

Appropriate values for this parameter are integers greater than 0.



 $Figure \ 150-Enter \ Confirmation \ Highlight \ (Cycles) \ parameter \ in \ the \ Communications \ Subtask \ Parameters \ group.$ 

# 8.2.5.7. Target Communication Timeout (Seconds)

This parameter determines the length of time allowed for a participant to respond to a Communications Event before the event times-out.

The default value for this parameter is 15.

Appropriate values for this parameter are real numbers greater than 0.

Target Communication Timeout (Seconds)	15	Default Value

 $Figure \ 151-Target \ Communication \ Timeout \ (Seconds) \ parameter \ in \ the \ Communications \ Subtask \ Parameters \ group.$ 

# 8.2.5.8. Communication Event Duration (Seconds)

This parameter specifies the duration of the audio files for the Script Generator Utility and AF-MATB. In the event that a researcher elects to use audio files of a shorter duration, lowering this value will allow the Script Generator Utility to pack more events into a given window of time. For more information, please see section **9.2.2.3 Communications Subtask**. Please note that modifying this parameter does not modify the audio files provided with the AF-MATB Package in any way.

The default value for this parameter is 6.

Appropriate values for this parameter are real numbers greater than 0. It is strongly recommended that this parameter not be modified.



Figure 152 – Communications Event Duration (Seconds) parameter in the Communications Subtask Parameters group.

# 8.2.5.9. Frequency Entry Method

As discussed in section **6.3 Communications Response Commands**, this parameter configures the Communications subtask entry methods. Selecting the "Arrow Keys" option enables the traditional entry method; selecting the "Number Pad" option enables the new entry method.

The default value for this parameter is "Arrow Keys."

Appropriate values for this parameter are "Arrow Keys" or "Number Pad."



 $\label{eq:Figure 153-Frequency Entry Method parameter in the Communications Subtask Parameters group$ 

Distribution A: Approved for public release; distribution unlimited. 88 ABW Cleared 12/03/2014; 88ABW-2014-5667.

Please note that keyboard functionality must be enabled for this entry method, since mouse input is not supported for this option. If the user has configured the Input Options parameter (discussed in section **8.2.2.1 Input Options**) as "GUI Buttons Only," this parameter will automatically be set to "Arrow Keys" and cannot be modified until the Input Options parameter is changed to either "Keyboard Only" or "Both GUI & Keyboard" (see Figure 154).



Figure 154 – How Frequency Entry Method parameter appears when the Input Options parameter has been set to "GUI Buttons Only."

# 8.2.5.10. Slot 1 & 2 Lower Limit

This parameter determines the lowest frequency available for Comm Slots 1 & 2 when the Communications subtask's entry method is configured as "Arrow Keys." Note that changing this parameter will not modify the included audio files in any way, so users must take care to ensure that the frequencies of all Communications Events for "NAV1" and "NAV2" are at or above this limit.

The default value for this parameter is 108.7.

Appropriate values for this parameter are real numbers greater than zero.



Figure 155 – Slot 1 & 2 Lower Limit parameter for the Communications Subtask Parameters group.

# 8.2.5.11. Slot 1 & 2 Upper Limit

This parameter determines the highest frequency available for Comm Slots 1 & 2 when the Communications subtask's entry method is configured as "Arrow Keys." Note that changing this parameter will not modify the included audio files in any way, so users must take care to ensure that the frequencies of all Communications Events for "NAV1" and "NAV2" are at or below this limit.

The default value for this parameter is 117.7.

Appropriate values for this parameter are real numbers greater than the value discussed in section **8.2.5.10 Slot 1 & 2 Lower Limit** plus at least one additional increment (discussed in section **8.2.5.12 Slot 1 & 2 Frequency Increments**).



Figure 156 – Slot 1 & 2 Upper Limit parameter for the Communications Subtask Parameters group.

Distribution A: Approved for public release; distribution unlimited. 88 ABW Cleared 12/03/2014; 88ABW-2014-5667.

# 8.2.5.12. Slot 1 & 2 Frequency Increments

This parameter determines how the frequency value will increment as participants scroll through frequencies for Comm Slots 1 & 2 when the Communications subtask's entry method is configured as "Arrow Keys." Changing this parameter will not modify the included audio files in any way, so users should ensure that they have not excluded frequencies that occur in any Communications Event for "NAV1" or "NAV2" when modifying this parameter.

The default value for this parameter is 0.2.

Appropriate values for this parameter are real numbers greater than zero.



Figure 157 – Slot 1 & 2 Frequency Increments parameter for the Communications Subtask Parameters group.

The Configuration Utility verifies that the previously specified limits for Comm Slots 1 & 2 conform to the increment specified for these slots. In the event that users specify an Upper Limit that cannot be reached using the previously-specified Lower Limit and Increment parameters, they will be notified and the configuration utility will update the Upper Limit to conform. The warning shown in Figure 158 was elicited when the user left the Lower Limit and Increment parameters for Comm Slots 1 & 2 as default values, but set the Comm Slots 1 & 2 Upper Limit to 117.8



Figure 158 – Notification of incongruent Lower Limit, Increment, and Upper Limit specifications for Comm Slots 1 & 2.

# 8.2.5.13. Slot 3 & 4 Lower Limit

This parameter determines the lowest frequency available for Comm Slots 3 & 4 when the Communications subtask's entry method is configured as "Arrow Keys." Changing this parameter will not modify the included audio files in any way, so users should ensure that the frequencies of all Communications Events for "COM1" and "COM2" are at or above this limit.

The default value for this parameter is 117.9.

Appropriate values for this parameter are real numbers greater than zero.

Slot 3 & 4 Lower Limit	117.9	Default Value
------------------------	-------	---------------

Figure 159 – Slot 3 & 4 Lower Limit parameter for the Communications Subtask Parameters group.

#### 8.2.5.14. Slot 3 & 4 Upper Limit

This parameter determines the highest frequency available for Comm Slots 3 & 4 when the Communications subtask's entry method is configured as "Arrow Keys." Note that changing this parameter will not modify the included audio files in any way, so users should ensure that the frequencies of all Communications Events for "COM1" and "COM2" are at or below this limit.

The default value for this parameter is 126.9.

Appropriate values for this parameter are real numbers greater than the value discussed in section **8.2.5.13 Slot 3 & 4 Lower Limit** plus at least one additional increment (discussed in section **8.2.5.15 Slot 3 & 4 Frequency Increments**).



Figure 160 – Slot 3 & 4 Upper Limit parameter for the Communications Subtask Parameters group.

#### 8.2.5.15. Slot 3 & 4 Frequency Increments

This parameter determines how the frequency value will increment as participants scroll through frequencies for Comm Slots 3 & 4 when the Communications subtask's entry method is configured as "Arrow Keys." Changing this parameter will not modify the included audio files in any way, so users should ensure when modifying this parameter that they have not excluded frequencies that occur in any Communications Event for "COM1" or "COM2."

The default value for this parameter is 0.2.

Appropriate values for this parameter are real numbers greater than zero.



Figure 161 – Slot 3 & 4 Frequency Increments parameter for the Communications Subtask Parameters group.

The Configuration Utility verifies that the previously specified limits for Comm Slots 3 & 4 conform to the increment specified for these slots. In the event that users specify an Upper Limit that cannot be reached using the previously-specified Lower Limit and Increment parameters, they will be notified and the configuration utility will update the Upper Limit to conform. The warning shown in Figure 162 was elicited when the user left the Lower Limit and Increment parameters for Comm Slots 3 & 4 as default values, but set the Comm Slots 3 & 4 Upper Limit to 126.8



Figure 162 – Notification of incongruent Lower Limit, Increment, and Upper Limit specifications for Comm Slots 3 & 4.

#### 8.2.6. Resource Management Subtask Basic Parameters Group

In the Resource Management Subtask Parameters, the user can configure parameters specific to the general function of the Resource Management subtask (see Figure 163). This group allows the user to configure the capacity of the tanks, as well as the flow rates of the pumps, and other parameters that govern the behavior of the subtask when in Manual Mode.



Figure 163 – The Resource Management Subtask Basic Parameters group in the AF-MATB Configuration Utility.

## 8.2.6.1. Pump 1 & Pump 3 Flow Rate (Liters/Minute)

This parameter determines the rate at which Pump 1 and Pump 3 can pump fuel into Tank A and Tank B, respectively.

The default value for this parameter is 1800.

Appropriate values for this parameter are real numbers greater than 0.

Pump 1 & Pump 3 Flow Rate (Liters/Minute)	1800	Default Value
-------------------------------------------	------	---------------

Figure 164 – Pump 1 & Pump 3 Flow Rate (Liters/Minute) parameter for the Resource Management Subtask Basic Parameters group.

## 8.2.6.2. Pump 2 & Pump 4 Flow Rate (Liters/Minute)

This parameter determines the rate at which Pump 2 and Pump 4 can pump fuel into Tank A and Tank B, respectively.

The default value for this parameter is 1600.

Appropriate values for this parameter are real numbers greater than 0.



Figure 165 – Pump 2 & Pump 4 Flow Rate (Liters/Minute) parameter for the Resource Management Subtask Basic Parameters group.

#### 8.2.6.3. Pump 5 & Pump 6 Flow Rate (Liters/Minute)

This parameter determines the rate at which Pump 5 and Pump 6 can pump fuel into Tank C and Tank D, respectively.

The default value for this parameter is 1600.

Appropriate values for this parameter are real numbers greater than 0.

Pump 5 & Pump 6 Flow Rate (Liters/Minute)	1600	Default Value
-------------------------------------------	------	---------------

Figure 166 – Pump 5 & Pump 6 Flow Rate (Liters/Minute) parameter for the Resource Management Subtask Basic Parameters group.

# 8.2.6.4. Pump 7 & Pump 8 Flow Rate (Liters/Minute)

This parameter determines the rate at which Pump 7 and Pump 8 can pump fuel into Tank B and Tank A, respectively.

The default value for this parameter is 2000.

Appropriate values for this parameter are real numbers greater than 0.

Pump 7 & Pump 8 Flow Rate (Liters/Minute)	2000	Default Value
-------------------------------------------	------	---------------

Figure 167 – Pump 7 & Pump 8 Flow Rate (Liters/Minute) parameter for the Resource Management Subtask Basic Parameters group.

# 8.2.6.5. Tank A & Tank B Drop Rate (Liters/Minute)

This parameter determines the rate at which Tank A and Tank B will lose fuel.

The default value for this parameter is 2000.

Appropriate values for this parameter are real numbers greater than 0.

Tank A & Tank B Drop Rate (Liters/Minute)	2000	Default Value
-------------------------------------------	------	---------------

Figure 168 – Tank A & Tank B Drop Rate (Liters/Minute) parameter for the Resource Management Subtask Basic Parameters group.

# 8.2.6.6. Tank A & Tank B Maximum (Liters)

This parameter determines the maximum possible capacity of Tank A and Tank B.

The default value for this parameter is 5000.

Appropriate values for this parameter are real numbers greater than 0.



Figure 169 – Tank A & Tank B Maximum parameter for the Resource Management Subtask Basic Parameters group.

# 8.2.6.7. Tank C & Tank D Maximum (Liters)

This parameter determines the maximum possible capacity of Tank C and Tank D.

The default value for this parameter is 2000.

Appropriate values for this parameter are real numbers greater than 0.

Tank C & Tank D Maximum (Liters)	2000	Default Value
----------------------------------	------	---------------

Figure 170 – Tank C & Tank D Maximum parameter for the Resource Management Subtask Basic Parameters group.

# 8.2.6.8. Tank A Starting Volume (Liters)

This parameter determines the initial volume of Tank A when the task begins (i.e. *Space Bar* is pressed).

The default value for this parameter is 2500.

Appropriate values for this parameter are integers greater than 0 and less than the value for the Tank B Maximum discussed in section **8.2.6.6 Tank A & Tank B Maximum (Liters)**.

Tank A Starting Volume (Liters)	2500	Default Value
---------------------------------	------	---------------

Figure 171 – Tank A Starting Volume (Liters) parameter for the Resource Management Subtask Basic Parameters group.

# 8.2.6.9. Tank B Starting Volume (Liters)

This parameter determines the initial volume of Tank B when the task begins (i.e. *Space Bar* is pressed).

The default value for this parameter is 2500.

Appropriate values for this parameter are integers greater than 0 and less than the value for the Tank B Maximum discussed in section **8.2.6.6 Tank A & Tank B Maximum (Liters)**.

Tank B Starting Volume (Liters)	2500	Default Value
---------------------------------	------	---------------

Figure 172 – Tank B Starting Volume (Liters) parameter for the Resource Management Subtask Basic Parameters group.

### 8.2.6.10. Tank C Starting Volume (Liters)

This parameter determines the initial volume of Tank C when the task begins (i.e. *Space Bar* is pressed).

The default value for this parameter is 1000.

Appropriate values for this parameter are integers greater than 0 and less than the value for the Tank B Maximum discussed in section **8.2.6.7 Tank C & Tank D Maximum (Liters)**.

Tank C Starting Volume (Liters)	1000	Default Value
---------------------------------	------	---------------

Figure 173 – Tank C Starting Volume (Liters) parameter for the Resource Management Subtask Basic Parameters group.

# 8.2.6.11. Tank D Starting Volume (Liters)

This parameter determines the initial volume of Tank D when the task begins (i.e. *Space Bar* is pressed).

The default value for this parameter is 1000.

Appropriate values for this parameter are integers greater than 0 and less than the value for the Tank B Maximum discussed in section **8.2.6.7 Tank C & Tank D Maximum (Liters)**.



Figure 174 – Tank D Starting Volume (Liters) parameter for the Resource Management Subtask Basic Parameters group.

#### 8.2.6.12. Pump Failure Duration (Seconds)

This parameter determines the duration of a Pump Failure. For more information on this malfunction, please see sections **3.2.1 Manual Mode**, **6.5.3.1.2 Timeout Commands**, and **8.2.2.4 Enable Manual Pump Repair**?

The default value for this parameter is 10.

Appropriate values for this parameter are real numbers greater than 0.

Pump Failure Duration (Seconds)	10	Default Value
---------------------------------	----	---------------

Figure 175 – Pump Failure Duration (Seconds) parameter for the Resource Management Subtask Basic Parameters group.

# 8.2.7. System Monitoring Subtask Automation Parameters Group

In the System Monitoring Subtask Automation, the user can configure parameters specific to the System Monitoring subtask when operating in Automated Mode (see Figure 176). For more information on either of the following parameters, see section **3.1.2 Automated Mode**.

	System Monitoring Subtask Automation Parameters				
	Automation Fault Restore Time (Seconds)	4	Default Value		
	Automation Failure Duration (Seconds)	10	Default Value		

Figure 176 – The System Monitoring Subtask Automation Parameters group in the AF-MATB Configuration Utility.

# 8.2.7.1. Automation Fault Restore Time (Seconds)

This parameter determines the duration of a Gauge malfunction that will be successfully detected and corrected by the System Monitoring automation.

The default value of this parameter is 4.

Appropriate values for this parameter are real numbers greater than 0.



Figure 177 – The Automation Fault Restore Time (Seconds) parameter in the System Monitoring Subtask Automation Parameters group.

# 8.2.7.2. Automation Failure Duration (Seconds)

This parameter determines the duration of a Gauge malfunction that is not detected and corrected by the System Monitoring automation. In this case, this duration determines the amount of time a participant has to identify the malfunction before the malfunction times-out and the gauge resumes normal function.

The default value of this parameter is 10.

Appropriate values for this parameter are real numbers greater than 0.

Automation Failure Duration (Seconds)	10	Default Value	

Figure 178 – The Automation Failure Duration (Seconds) parameter in the System Monitoring Subtask Automation Parameters group.

# 8.2.8. Tracking Subtask Difficulty Parameters Group

In the Tracking Subtask Difficulty Parameters group, the user can configure parameters that govern the difficulty of the Tracking subtask (see Figure 179). These parameters allow the user to configure the speed of the reticle movement and the frequency of changes in direction of the reticle.

The subtask will first determine how long the reticle will travel in one direction. This value is generated from a random distribution using the direction limits provided, and then the speed of movement is computed, which is generated from a separate random distribution based on the speed limits provided. After computing the direction limit and speed, the exact heading of the reticle is calculated. The reticle will then maintain the calculated speed and heading for the duration of the direction limit until either the direction limit is reached or the reticle reaches a subtask boundary, which would cause the reticle to bounce off the boundary. Once the direction limit expires, a new direction limit and a new speed limit are selected from their respective distributions, and a new heading is calculated. This process is repeated until the task expires.

Tracking Subtask Difficulty Parameters		
Low Difficulty Reticle Speed Lower Limit (Pixels/Cycle)	2	Default Value
Low Difficulty Reticle Speed Upper Limit (Pixels/Cycle)	4	Default Value
Low Difficulty Direction Change Lower Limit (Cycles)	20	Default Value
Low Difficulty Direction Change Upper Limit (Cycles)	60	Default Value
Moderate Difficulty Reticle Speed Lower Limit (Pixels/Cycle)	4	Default Value
Moderate Difficulty Reticle Speed Upper Limit (Pixels/Cycle)	6	Default Value
Moderate Difficulty Direction Change Lower Limit (Cycles)	10	Default Value
Moderate Difficulty Direction Change Upper Limit (Cycles)	30	Default Value
High Difficulty Reticle Speed Lower Limit (Pixels/Cycle)	6	Default Value
High Difficulty Reticle Speed Upper Limit (Pixels/Cycle)	8	Default Value
High Difficulty Direction Change Lower Limit (Cycles)	3	Default Value
High Difficulty Direction Change Upper Limit (Cycles)	9	Default Value

Figure 179 – The Tracking Subtask Difficulty Parameters group in the AF-MATB Configuration Utility.

Please note that the direction and speed distributions operate in a similar manner to the distribution discussed in section **8.2.3.3 End of Range Delay Max (Cycles)** in that as the difference between the upper and lower limits approaches zero, the length of time between changes in reticle direction and speed will become less random. Similarly, as the direction or

speed distributions are shifted towards zero, the reticle will change directions more frequently, and move at a slower speed, respectively.

Finally, all parameters in this section are affected by the cycling rate of the task discussed in section **8.2.2.9 Timer Cycling Rate (Hz)**. If the cycling rate of the task is modified, the parameters in this section should also be modified.

# 8.2.8.1. Low Difficulty Reticle Speed Lower Limit (Pixels/Cycle)

This parameter determines the lower limit of the distribution used to control the speed of the reticle at Low Difficulty.

The default value of this parameter is 2.

Appropriate values for this parameter are real numbers greater than 0 and less than 50.



Figure 180 – Low Difficulty Reticle Speed Lower Limit (Pixels/Cycles) parameter in the Tracking Subtask Difficulty Parameters group.

# 8.2.8.2. Low Difficulty Reticle Speed Upper Limit (Pixels/Cycle)

This parameter determines the upper limit of the distribution used to control the speed of the reticle at Low Difficulty.

The default value of this parameter is 4.

Appropriate values for this parameter are real numbers less than 50 and greater than the value for the Lower Limit discussed in section **8.2.8.1 Low Difficulty Reticle Speed Lower Limit** (**Pixels/Cycle**).

Low Difficulty Reticle Speed Upper Limit (Pixels/Cycle)	4	Default Value
---------------------------------------------------------	---	---------------

Figure 181 – Low Difficulty Reticle Speed Upper Limit (Pixels/Cycles) parameter in the Tracking Subtask Difficulty Parameters group.

# 8.2.8.3. Low Difficulty Direction Change Lower Limit (Cycles)

This parameter determines the lower limit of the distribution used to control how often (based on duration) the reticle changes direction during Low Difficulty Tracking.

The default value of this parameter is 20.

Appropriate values for this parameter are integers greater than 0.

Low Difficulty Direction Change Lower Limit (Cycles)	20	Default Value
		· CC 1/ D

Figure 182 – Low Difficulty Direction Change Lower Limit (Cycles) parameter in the Tracking Subtask Difficulty Parameters group.

# 8.2.8.4. Low Difficulty Direction Change Upper Limit (Cycles)

This parameter determines the upper limit of the distribution used to control how often (based on duration) the reticle changes direction during Low Difficulty.

The default value of this parameter is 60.

Appropriate values for this parameter are integers greater than the value for the Lower Limit discussed in section **8.2.8.3 Low Difficulty Direction Change Lower Limit (Cycles)**.

Low Difficulty Direction Change Upper Limit (Cycles)	60	Default Value
------------------------------------------------------	----	---------------

Figure 183 – Low Difficulty Direction Change Upper Limit (Cycles) parameter in the Tracking Subtask Difficulty Parameters group.

# 8.2.8.5. Moderate Difficulty Reticle Speed Lower Limit (Pixels/Cycle)

This parameter determines the lower limit of the distribution used to control the speed of the reticle at Moderate Difficulty.

The default value of this parameter is 4.

Appropriate values for this parameter are real numbers greater than 0 and less than 50.

Moderate Difficulty Reticle Speed Lower Limit (Pixels/Cycle)	4	Default Value
--------------------------------------------------------------	---	---------------

Figure 184 – Moderate Difficulty Reticle Speed Lower Limit (Pixels/Cycles) parameter in the Tracking Subtask Difficulty Parameters group.

**8.2.8.6.** Moderate Difficulty Reticle Speed Upper Limit (Pixels/Cycle) This parameter determines the upper limit of the distribution used to control the speed of the reticle at Moderate Difficulty.

The default value of this parameter is 6.

Appropriate values for this parameter are real numbers less than 50 and greater than the value for the Lower Limit discussed in section **8.2.8.5 Moderate Difficulty Reticle Speed Lower Limit** (Pixels/Cycle).

Moderate Difficulty Reticle Speed Upper Limit (Pixels/Cycle)	6	Default Value
--------------------------------------------------------------	---	---------------

Figure 185 – Moderate Difficulty Reticle Speed Upper Limit (Pixels/Cycles) parameter in the Tracking Subtask Difficulty Parameters group.

# 8.2.8.7. Moderate Difficulty Direction Change Lower Limit (Cycles)

This parameter determines the lower limit of the distribution used to control how often (based on duration) the reticle changes direction during Moderate Difficulty.

The default value of this parameter is 10.

Appropriate values for this parameter are integers greater than 0.

Moderate Difficulty Direction Change Lower Limit (Cycles)	10	Default Value
-----------------------------------------------------------	----	---------------

Figure 186 – Moderate Difficulty Direction Change Lower Limit (Cycles) parameter in the Tracking Subtask Difficulty Parameters group.

# 8.2.8.8. Moderate Difficulty Direction Change Upper Limit (Cycles)

This parameter determines the upper limit of the distribution used to control how often (based on duration) the reticle changes direction during Moderate Difficulty.

The default value of this parameter is 30.

Appropriate values for this parameter are integers greater than the value for the Lower Limit discussed in section **8.2.8.7 Moderate Difficulty Direction Change Lower Limit (Cycles)**.

Moderate Difficulty Direction Change Upper Limit (Cycles)	30	Default Value	
Figure 187 – Moderate Difficulty Direction Change Upper Limit (Cycles) parameter in the Tracking Subtask Difficulty			

Figure 187 – Moderate Difficulty Direction Change Upper Limit (Cycles) parameter in the Tracking Subtask Difficulty Parameters group.

# 8.2.8.9. High Difficulty Reticle Speed Lower Limit (Pixels/Cycle)

This parameter determines the lower limit of the distribution used to control the speed of the reticle at High Difficulty.

The default value of this parameter is 6.

Appropriate values for this parameter are real numbers greater than 0 and less than 50.

High Difficulty Reticle Speed Lower Limit (Pixels/Cycle)	6	Default Value
----------------------------------------------------------	---	---------------

Figure 188 – High Difficulty Reticle Speed Lower Limit (Pixels/Cycles) parameter in the Tracking Subtask Difficulty Parameters group.

# 8.2.8.10. High Difficulty Reticle Speed Upper Limit (Pixels/Cycle)

This parameter determines the upper limit of the distribution used to control the speed of the reticle at High Difficulty.

The default value of this parameter is 8.

Appropriate values for this parameter are real numbers less than 50 and greater than the value for the Lower Limit discussed in section **8.2.8.9 High Difficulty Reticle Speed Lower Limit** (**Pixels/Cycle**).

High Difficulty Reticle Speed Upper Limit (Pixels/Cycle)	8	Default Value
----------------------------------------------------------	---	---------------

Figure 189 – High Difficulty Reticle Speed Upper Limit (Pixels/Cycles) parameter in the Tracking Subtask Difficulty Parameters group.

# 8.2.8.11. High Difficulty Direction Change Lower Limit (Cycles)

This parameter determines the lower limit of the distribution used to control how often (based on duration) the reticle changes direction during High Difficulty.

The default value of this parameter is 3.

Appropriate values for this parameter are integers greater than 0.

High Difficulty Direction Change Lower Limit (Cycles)	3	Default Value	
-------------------------------------------------------	---	---------------	--

Figure 190 – High Difficulty Direction Change Lower Limit (Cycles) parameter in the Tracking Subtask Difficulty Parameters group.

# 8.2.8.12. High Difficulty Direction Change Upper Limit (Cycles)

This parameter determines the upper limit of the distribution used to control how often (based on duration) the reticle changes direction during High Difficulty.

The default value of this parameter is 9.

Appropriate values for this parameter are integers greater than the value for the Lower Limit discussed in section **8.2.8.11 High Difficulty Direction Change Lower Limit (Cycles)**.

High Difficulty Direction Change Upper Limit (Cycles)	9	Default Value
-------------------------------------------------------	---	---------------

Figure 191 – High Difficulty Direction Change Upper Limit (Cycles) parameter in the Tracking Subtask Difficulty Parameters group.

# 8.2.9. Resource Management Subtask Automation Parameters

In the Resource Management Subtask Automation Parameters group, the user can configure parameters responsible for governing the Resource Management subtask when in Automated Mode (see Figure 192). Please note that this group also contains parameters that are required by the Resource Management subtask for both Manual Mode and Automated Mode.

Resource Management Subtask Automation Parameters			
Tank Volume Update Interval (Seconds)	2	Default Value	
Main Tank Automation Lower Limit (Liters)	2350	Default Value	
Main Tank Automation Upper Limit (Liters)	2650	Default Value	
Supply Tank Automation Lower Limit (Liters)	450	Default Value	
Supply Tank Automation Upper Limit (Liters)	1200	Default Value	
Automation Maximum Tank Difference (Liters)	300	Default Value	
Resource Management RMS Interval (Seconds)	5	Default Value	
RMS Target Value (Liters)	2500	Default Value	
Hide Automation Behavior?	Yes 💿 No	Default Value	
Automation Fault Duration (Seconds)	10	Default Value	

Figure 192 – The Resource Management Subtask Automation Parameters group in the AF-MATB Configuration Utility.

# 8.2.9.1. Tank Volume Update Interval (Seconds)

This parameter determines the amount of time that will elapse between visual updates for the Resource Management subtask, allowing the researcher to adjust the visual noise associated with this subtask. As this value approaches 0, the volume display will update more rapidly. Please note that this parameter does not affect the data logging for this subtask, and that this parameter is required when in operating in either Automated or Manual Mode, but was included in this section due to its convolution with other parameters (see section **8.2.9.10 Automation Fault Duration (Seconds)** for more information).

The default value for this parameter is 2.

Appropriate values for this parameter are real numbers greater than 0.



Figure 193 – Tank Volume Update Interval (Seconds) parameter in the Resource Management Subtask Automation Parameters group.

# 8.2.9.2. Main Tank Automation Lower Limit (Liters)

This parameter defines the lower limit of the acceptable fuel range of Tanks A and B and works in conjunction with the Upper Limit discussed in the following section (8.2.9.3 Main Tank Automation Upper Limit (Liters)). In either Manual Mode or Automated Mode, this parameter is used to display the acceptable fuel range indicator discussed in section 3.2 Resource Management and illustrated in Figure 013, as well as to establish the "range" used for scoring, discussed in sections 7.3.2.10 Resource Management Log and 7.3.3.1.3 Resource Management Section. Tank A or Tank B volumes that are greater than or equal to the value of this parameter, but less than or equal to those of the Upper Limit, are considered "in range," while values that do not meet these criteria are "outside of range." Finally, when in Automated Mode, this parameter specifies that if the automation is functioning properly, Tanks A and B will not have less than this amount of fuel in either tank at any time.

The default value for this parameter is 2350.

Appropriate values for this parameter are real numbers greater than 0, but less than the Upper Limit discussed in section **8.2.9.3 Main Tank Automation Upper Limit (Liters)** and RMS Target Value discussed in section **8.2.9.8 RMS Target Value (Liters)**.

Main Tank Automation Lower Limit (Liters)	2350	Default Value
Figure 194 – Main Tank Automation Lower Limit (Liters) parameter in t	the Resource Manag	gement Subtask Automation
Parameters group.		

#### 8.2.9.3. Main Tank Automation Upper Limit (Liters)

This parameter defines the upper limit of the acceptable fuel range of Tanks A and B, and works in conjunction with the Lower Limit discussed in the following section (8.2.9.2 Main Tank Automation Lower Limit (Liters)). In either Manual Mode or Automated Mode, this parameter is used to display the acceptable fuel range indicator discussed in section 3.2 Resource Management and illustrated in Figure 013, as well as to establish the "range" used for scoring, discussed in sections 7.3.2.10 Resource Management Log and 7.3.3.1.3 Resource Management Section. Tank A or Tank B volumes that are less than or equal to the value of this parameter, but greater than or equal to those of the Lower Limit, are considered "in range," while values that do not meet these criteria are "outside of range." Finally, when in Automated Mode, this parameter specifies that if the automation is functioning properly, Tanks A and B will not have more than this amount of fuel in either tank at any time.

The default value for this parameter is 2650.

Appropriate values for this parameter are real numbers less than the maximum of Tanks A and B discussed in section **8.2.6.6 Tank A & Tank B Maximum (Liters)** but greater than Lower Limit discussed in section **8.2.9.2 Main Tank Automation Lower Limit (Liters)** and RMS Target Value discussed in section **8.2.9.8 RMS Target Value (Liters)**.

Main Tank Automation Upper Limit (Liters)	2650		Default Value
-------------------------------------------	------	--	---------------

Figure 195 – Main Tank Automation Upper Limit (Liters) parameter in the Resource Management Subtask Automation Parameters group.

#### 8.2.9.4. Supply Tank Automation Lower Limit (Liters)

This parameter defines the lower limit of the acceptable fuel range of Tanks C and D, in conjunction with the Upper Limit discussed in the following section (8.2.9.4 Supply Tank Automation Lower Limit (Liters)). The automation for the Resource Management subtask not only requires that an acceptable range be established for Tanks A and B, but also for Tanks C and D. Please note that while this parameter controls the automation, neither this parameter nor the associated Upper Limit has any effect on any performance metric for this subtask.

The default value for this parameter is 450.

Appropriate values for this parameter are real numbers greater than 0, but less than the Upper Limit discussed in section **8.2.9.5 Supply Tank Automation Upper Limit (Liters)**.

Supply Tank Automation Lower Limit (Liters)         450         Default Value	Supply Tank Automation Lower Limit (Liters)	450	Default Value
-------------------------------------------------------------------------------	---------------------------------------------	-----	---------------

Figure 196 – Supply Tank Automation Lower Limit (Liters) parameter in the Resource Management Subtask Automation Parameters group.

#### 8.2.9.5. Supply Tank Automation Upper Limit (Liters)

This parameter defines the upper limit of the acceptable fuel range of Tanks C and D, in conjunction with the Lower Limit discussed in the following section (8.2.9.4 Supply Tank Automation Lower Limit (Liters)). The automation for the Resource Management subtask not only requires that an acceptable range be established for Tanks A and B, but also for Tanks C and D. Please note that while this parameter controls the automation, neither this parameter nor the associated Lower Limit has any effect on any performance metric for this subtask.

The default value for this parameter is 1200.

Appropriate values for this parameter are real numbers less than the maximum of Tanks C and D discussed in section 8.2.6.7 Tank C & Tank D Maximum (Liters) but greater than Lower Limit discussed in section 8.2.9.4 Supply Tank Automation Lower Limit (Liters).

Supply Tank Automation Upper Limit (Liters)	1200	Default Value
<b>'igure 197</b> – Supply Tank Automation Upper Limit (Liters) parameter in	the Resource Mana	gement Subtask Automation

Figure 197 – Supply Tank Automation Upper Limit (Liters) parameter in the Resource Management Subtask Automation Parameters group.

## 8.2.9.6. Automation Maximum Tank Difference (Liters)

This parameter determines the maximum difference between Tanks A and B permitted by the automation. When the Resource Management is configured to use automation Algorithm 1 (see section **8.2.2.12 Resource Management Automation Algorithm**) and automation is engaged, the automation will use this value as the limit to determine when Pumps 7 and 8 should be used to equalize the volumes of Tanks A and B.

The default value for this parameter is 300.

Appropriate values for this parameter are real numbers greater than 0 and less than the maximum of Tanks A and B discussed in section **8.2.6.6 Tank A & Tank B Maximum (Liters)**.

Automation Maximum Tank Difference (Liters)	300	Default Value
---------------------------------------------	-----	---------------

Figure 198 – Autopilot Maximum Tank Difference (Liters) parameter in the Resource Management Subtask Automation Parameters group.

#### 8.2.9.7. Resource Management RMS Interval (Seconds)

As described in section **7.3.3.1.3 Resource Management Section**, some measures are computed using an independent sampling rate that is lower than others and results in a subset of all data stored in the Resource Management Log. This parameter determines the interval at which data are stored for that subset. These stored data are used to compute the <u>Target RMSD (Euc. Dist)</u>, <u>Tank A Target RMSD</u>, and <u>Tank B Target RMSD</u> metrics. Operationally, this parameter means that after every specified interval, one data point is stored in a log that is used to compute the previously mentioned metrics.

The default value for this parameter is 5.

Appropriate values for this parameter are integers greater than 0.

Resource Management RMS Interval (Seconds)	5	Default Value
--------------------------------------------	---	---------------

Figure 199 – Resource Management RMS Interval (Seconds) parameter in the Resource Management Subtask Automation Parameters group.

## 8.2.9.8. RMS Target Value (Liters)

This parameter establishes the exact value for the target volumes of Tanks A and B. In either Manual Mode or Automated Mode, this parameter is used to display the target volume discussed in section **3.2 Resource Management** and illustrated in Figure 013, as well as to establish the "target value" used for scoring, discussed in sections **7.3.2.10 Resource Management Log** and **7.3.3.1.3 Resource Management Section**. Please note that this parameter is used by the Resource Management subtask while operating in either Manual or Automated Mode, but was included in this section due to its convolution with other parameters in this section.

The default value for this parameter is 2500.

Appropriate values for this parameter are real numbers greater the Main Tank Automation Lower Limit (see section **8.2.9.2 Main Tank Automation Lower Limit (Liters**)) and less than the Main Tank Automation Upper Limit (see section **8.2.9.3 Main Tank Automation Upper Limit** (**Liters**)). Ideally, the value of this parameter should be at the midpoint of the two limits.

RMS Target Value (Liters)	2500	Default Value
---------------------------	------	---------------

Figure 200 – RMS Target Value (Liters) parameter in the Resource Management Subtask Automation Parameters group.

#### 8.2.9.9. Hide Automation Behavior

This parameter determines whether the Resource Management's automation should mask the status of the pumps, as discussed in section **3.2.2.1 Automation Algorithm 1** and illustrated in Figure 015.

The default value for this parameter is "No."

Appropriate values for this parameter are "Yes" or "No."



Figure 201 – Hide Automation Behavior? parameter in the Resource Management Subtask Automation Parameters group.

Please note that if the Resource Management Automation Algorithm is set to "Algorithm 2" (see section **8.2.2.12 Resource Management Automation Algorithm**), this parameter will automatically be set to "Yes," and it cannot be modified unless "Algorithm 1" is selected (see Figure 202).



Figure 202 – How the Hide Automation Behavior? parameter appears when the Resource Management Automation Algorithm parameter has been set to "Algorithm 2."

#### 8.2.9.10. Automation Fault Duration (Seconds)

This parameter determines the duration of an automation failure when the Resource Management subtask is operating in Automated Mode using Algorithm 2. Please note that the duration of automation failures must be greater than a multiple of the Tank Volume Update Interval (section **8.2.9.1 Tank Volume Update Interval (Seconds**)). This is to ensure that participants monitoring this subtask have the entire specified duration to detect an automation failure, and why the Update Interval was included in this section.

The default value of this parameter is 10.

Appropriate values for this parameter are real numbers greater than and a multiple of the Tank Volume Update Interval (see section **8.2.9.1 Tank Volume Update Interval (Seconds**)).

Automation Fault Duration (Seconds)	10	Default Value
-------------------------------------	----	---------------

Figure 203 – Automation Fault Duration (Seconds) parameter in the Resource Management Subtask Automation Parameters group.

#### 8.2.10. **Subtask Visibility Parameters**

In the Subtask Visibility Parameters group, the user can control which subtasks are initially visible when the task loads (see Figure 204). These parameters only control the initial appearance of AF-MATB or the appearance of AF-MATB if a Config File is loaded. If a Script File is loaded into the task, it must contain separate instructions to hide the desired subtasks (see section 9.2.2.7 Subtask Component Visibility for more information).

Subtask Visibility Parameters		
Initially hide System Monitoring subtask?	○ Yes	Default Value
Initially hide Tracking subtask?	O Yes  No	Default Value
Initially hide Communications subtask?	O Yes  No	Default Value
Initially hide Resource Management subtask?	O Yes  No	Default Value
Initially hide Scheduling window?	O Yes  No	Default Value
Initially hide Pump Status window?	O Yes  No	Default Value

Figure 204 - Subtask Visibility Parameters window in the AF-MATB Configuration Utility.

# 8.2.10.1. Initially hide System Monitoring subtask?

This parameter determines whether the System Monitoring subtask will be hidden when AF-MATB is initially loaded.

The default value of this parameter is "No."

Appropriate values for this parameter are "Yes," or "No."



Figure 205 – Initially hide System Monitoring subtask? parameter in the Subtask Visibility Parameters group.

# 8.2.10.2. Initially hide Tracking subtask?

This parameter determines whether the Tracking subtask will be hidden when AF-MATB is initially loaded.

The default value of this parameter is "No."

Appropriate values for this parameter are "Yes" or "No."

Initially hide Tracking subtask?	O Yes	● No	Default Value
----------------------------------	-------	------	---------------

Figure 206 – Initially hide Tracking subtask? parameter in the Subtask Visibility Parameters group.

# 8.2.10.3. Initially hide Communications subtask?

This parameter determines whether the Communications subtask will be hidden when AF-MATB is initially loaded.

The default value of this parameter is "No."

Appropriate values for this parameter are "Yes" or "No."

Initially hide Communications subtask?	() Yes	€ No	Default Value

 $\label{eq:Figure 207-Initially hide Communications subtask? parameter in the Subtask Visibility Parameters group.$ 

# 8.2.10.4. Initially hide Resource Management subtask?

This parameter determines whether the Resource Management subtask will be hidden when AF-MATB is initially loaded.

The default value of this parameter is "No."

Appropriate values for this parameter are "Yes" or "No."

 $Figure \ 208 - \ Initially \ hide \ Communications \ subtask? \ parameter \ in \ the \ Subtask \ Visibility \ Parameters \ group.$ 

# 8.2.10.5. Initially hide Scheduling window?

This parameter determines whether the Scheduling window will be hidden when AF-MATB is initially loaded.

The default value of this parameter is "No."

Appropriate values for this parameter are "Yes" or "No."

Initially hide Scheduling window?	🔿 Yes 💿 No	Default Value
-----------------------------------	------------	---------------

Figure 209 – Initially hide Scheduling window? parameter in the Subtask Visibility Parameters group.

# 8.2.10.6. Initially hide Pump Status window?

This parameter determines whether the Pump Status window will be hidden when AF-MATB is initially loaded.

The default value of this parameter is "No."

Appropriate values for this parameter are "Yes" or "No."

Initially hide Pump Status window?	⊖ Yes	● No	Default Value
------------------------------------	-------	------	---------------

Figure 210 – Initially hide Pump Status window? parameter in the Subtask Visibility Parameters group.

#### 8.2.11. Port Triggering Parameters

As discussed in section **4.5 Serial and Digital Port-Triggering**, this version of AF-MATB supports the ability to send serial and/or digital triggers from the computer running AF-MATB to any secondary acquisition system. In this section of the Configuration Utility, the user can access the settings specific to this feature (see Figure 211).



Figure 211 – The Port Triggering Parameters group in the AF-MATB Configuration Utility.

#### 8.2.11.1. Event List

The Event List in the Port Triggering Parameters group is a clickable list that details all task actions that have been configured to send a trigger, and their associated event codes. Referring to Figure 212, the event code for "F1 Gauge Fault (Normal)" is "1." When it is highlighted, as is the case here, the event code will be automatically populated into the Change Event Code field (see section **8.2.11.4 Change Event Code**).

The functionality of the 130 different entries in the Event List can be configured so that only a selected subset of the entries sends event codes. For example, if a user decides that all event codes except those that pertain to the System Monitoring subtask are not of interest to him, the user can elect to delete those event codes by simply leaving the Change Event Code field empty or by updating the value to "NaN" (Not a Number). Figure 212 illustrates the use of the NaN event code for the "F6 Light Fault." This means that no event code has been programmed for this event and that when this event occurs, even if port-triggering is enabled, no event code will be sent.

F1 Gauge Fault (Normal) = 1	~
F2 Gauge Fault (Normal) = 2	
F3 Gauge Fault (Normal) = 3	
F4 Gauge Fault (Normal) = 4	
F5 Light Fault = 5	
F6 Light Fault = NaN	
F1 Gauge Correct Response (Normal) = 7	
F2 Gauge Correct Response (Normal) = 8	
F3 Gauge Correct Response (Normal) = 9	
F4 Gauge Correct Response (Normal) = 10	
F5 Light Correct Response = 11	
F6 Light Correct Response = 12	
F1 Gauge FA Response (Normal) = 13	
F2 Gauge FA Response (Normal) = 14	
F3 Gauge FA Response (Normal) = 15	
F4 Gauge FA Response (Normal) = 16	
F5 Light FA Response = 17	
F6 Light FA Response = 18	
F1 Gauge Timeout (Normal) = 19	
F2 Gauge Timeout (Normal) = 20	
F3 Gauge Timeout (Normal) = 21	
F4 Gauge Timeout (Normal) = 22	5
F5 Linht Timeout = 23	·

Figure 212 – The Event List, used in the Port Triggering Parameters group to show the event codes of every event in AF-MATB.

# 8.2.11.2. Enable Port Triggering?

This parameter determines whether or not port-triggering will be enabled in AF-MATB. Please note that if the task encounters a problem initializing either serial or digital communication, the task will automatically disable all port triggering.

The default value of this parameter is "No."

Appropriate values for this parameter are "Yes" or "No."



Figure 213 – The Enable Port Triggering? parameter in the Port Triggering Parameters group.

# 8.2.11.3. Triggering Mechanism

This parameter determines if port-triggering will be delivered through a serial port, a digital port, or through both ports.

The default value of this parameter is "Serial."

Appropriate values for this parameter are "Serial," "Digital," or "Both."



Figure 214 – The Triggering Mechanism parameter in the Port Triggering Parameters group.

# 8.2.11.4. Change Event Code

As discussed in section **8.2.11.1 Event List**, this feature allows the user to change, re-map, or delete the event codes of any entry in the Event List. By clicking on an entry in the Event List, the event code associated with that entry is populated in the field (see Figure 215). In order to change the event code, the user simply updates the field with the new information by either entering in a new integer to change the event code (see Figure 216) or by either leaving it blank or entering NaN to remove that event code (see Figure 217), and clicking the *Change Event Code* button (see Figure 218 & Figure 219).







Figure 216 – A user entering in a new event code for the "F1 Gauge Fault (Normal)" entry. They have changed the event code in the entry field, but have not hit the *Change Event Code* button. Note that the event code for this entry in the Event List is still





Figure 217 – A user removing the event code for the "F1 Gauge Fault (Normal)" entry. They have cleared the event code in the entry field, but have not hit the *Change Event Code* button. Note that the event code for this entry in the Event List is still "1."



Figure 218 – A user confirming the new event code for the "F1 Gauge Fault (Normal)" entry. Note that the event code for this entry is now "1234."


Figure 219 – A user confirming the removal of an event code for the "F1 Gauge Fault (Normal)" entry. Note that the event code for this entry is now "NaN."

If an invalid event code is entered, such as a negative value, users will be notified and their changes will not be saved.



Figure 220 – Notification message explaining to the user that they have entered an invalid event code.

Finally, please note that clicking on the *Default Value* button next to the *Change Event Code* button will automatically reset the currently selected entry on the Event List to its default value, as well as reset the value in the field.

#### 8.2.11.5. Serial Port Address

This parameter defines the serial (also known as COM) port that should be opened by the AF-MATB task to communicate with the acquisition system. Please note that the task supports only 1 open COM port.

The default value of this parameter is "Com1."

Appropriate values for this parameter are any available COM ports.

|--|

Figure 221 - The Serial Port Address parameter in the Port Triggering Parameters group.

## 8.2.11.6. Digital Port Address

This parameter defines the digital port that should be opened by the AF-MATB task to communicate with the acquisition system. In order to maximize the event codes that can be sent through the DAQ, this parameter should always be set to default.

The default value of this parameter is "port0."

Appropriate values for this parameter are "port0" or "port1," though it is highly recommended that this value not be changed.



Figure 222 – The Digital Port Address parameter in the Port Triggering Parameters group.

## 8.2.11.7. Digital Device Nickname

This parameter defines the nickname of the DAQ as recognized by the Measurement & Automation Explorer (MAX) software. As discussed in section **4.5 Serial and Digital Port-Triggering**, each DAQ, when first plugged into the task computer, is assigned a device nickname by MAX. This nickname is used by both the MAX software and NIDAQmx drivers to differentiate one DAQ from another. The drivers used by AF-MATB require this identifier in order to call the appropriate hardware when sending digital triggers. As such, it is imperative that this parameter be configured properly, with a value identical to that of the nickname indicated in MAX. Please refer to section **4.5 Serial and Digital Port-Triggering** for information on determining what nickname your DAQ has been given by MAX, and how to rename your DAQ if necessary.

The default value of this parameter is "Dev1."

Appropriate values for this parameter are any device nicknames generated by the MAX software.



Figure 223 – The Digital Device Nickname parameter in the Port Triggering Parameters group.

## 8.2.11.8. Load Separate Event Configuration

The *Load Separate Event Configuration* button allows users to import, if desired, only the event codes from a Config File, allowing researchers can reduce the time required to construct a Config File. For example, one user can create a Config File with all of the desired options for a study, and simultaneously, another user can create a Config File with the desired modifications to the events codes. Then, the files can be merged.

To use this function, the user clicks on the *Load Separate Event Configuration* button, which launches the typical file selection GUI (see Figure 224). Using this GUI, the user can click on any Config or Script File and then click the *Open* button. The Configuration Utility will then load only the event codes from the selected file and render them in the Event List.

•	Select File to Open	×
Look in:	🕌 AF-MATB 💽 🖛 🗈 💣 📰 🔻	
Recent places Desktop Libraries Computer	Name AF-MATB System Files Test_02-Sep-2013_21-12-40_CONFIG	Date modifie 8/17/2013 12: 9/2/2013 9:12
	<	>
	File name: <ul> <li>Files of type:</li> <li>All MATLAB files</li> <li> <li> </li> </li></ul> <li> <ul> <li></li></ul></li>	Open Cancel

Figure 224 – Typical file selection GUI that is launched after pressing the Load Separate Event Configuration button.

## 8.2.11.9. Remove All Event Codes

This button allows the user to remove the event codes from all entries in the Event List. This was designed for situations where the user wanted event codes from only a small subset of entries in the Event List, allowing the user to clear all and only define event codes for the entries of interest, which would take far less time than removing the event code from every entry, one at a time.

## 8.2.11.10. Set All Event Codes to Defaults

This button allows the user to reset all entries in the Event List back to their default event codes. Users can use this button to ensure they are starting from default values.

157

Distribution A: Approved for public release; distribution unlimited. 88 ABW Cleared 12/03/2014; 88ABW-2014-5667.

## 8.3. Main Function Buttons

This section will detail the functionality of the main function buttons located on the right of side of the utility window of the AF-MATB Configuration Utility window.



Figure 225 – The main function buttons located on the right of the AF-MATB Configuration Utility window.

#### 8.3.1. Save Values

Pressing this button allows users to store the parameters as a .mat file, which then can be loaded into AF-MATB or the Script Generator Utility. After pressing the *Save Values* button, the Configuration Utility performs one last check of all the parameter, ensuring that all values are valid. If they are not valid, the errors will be highlighted and a message, called a status text, will appear in the lower left corner of the screen to inform the user that the saving process has been interrupted (see Figure 226). If there are no issues with any of the parameters, a file saving GUI will launch and the user will be able to enter a filename for this Config File into the appropriate field and press *Save* (see Figure 227). Any files saved by this utility will automatically append a date-and-time stamp to end of the file name input by the user. This ensures that no files will be accidentally overwritten. For example, the Config File in Figure 227 was saved as "Test1," and the utility automatically added the "\_03-Sep-2013\_13-07-14\_CONFIG" to the file name before saving it. Once the user enters a file name and presses *Save*, the Config File should successfully save, notifying the user when completed via the status text (see Figure 228).

<b>3</b>	AF_MATB_Con	figurationUtility_V	300		×
AF-MATB System Param V					
	AF-MATB System Parameters				
	( Input Options	GUI Buttons Only Keyboard Only Both GUI & Keyboard	Default Value		
	Enable Pausing?	• Yes O No	Default Value		
	Enable Manual Event Triggering?	Yes      No	Default Value		
	Enable Manual Pump Repair?	• Yes O No	Default Value		
	Display Date & Time?	• Yes O No	Default Value		
Please enter a number from 10 to 20 for the Timer	Enable Scheduling?		Default Value		
	Enable Escape key?		Default Value		Save Values
	Enable Home key?	• Yes O No	Default Value		L
	Timer Cycling Rate (Hz)		Default Value		Load Config File
	Intial Ramp-up Time Delay (Seconds)	5	Default Value		Set All Default Values
	NASA-TLX Ramp-Up Time Delay (Seconds)	5	Default Value	Sa	ve and Continue to Script Generator
×					
	Resource Management Automation Algorithm	Algorithm 1	Default Value	Lo	ad and Continue to Script Generator
Please fix highlighted	Enable Culticals & demotion Indianters?	Algorithm 2	Default Volue		Save and Continue to AF-MATB
issues before saving.	Enable Sublask Automation Indicators?	Ves No	Default Value		Load and Continue to AF-MATB
	Enable Information Throughput Mode?		Derault Value		∀. 3.00

Figure 226 – The AF-MATB Configuration Utility showing the user attempting to save a Config File with a missing value for one of the parameters in the AF-MATB System Parameters group. Note the text in the lower left of the window known as the status text, notifying the user of the issue.

	Save Parameters File As		×
Save įn:	Scripts 💌	+ 🗈 💣 📰 -	
Recent places Desktop Libraries Computer	Name AF-MATB System Files Test1_03-Sep-2013_13-07-14_CONFIG	Date modified 8/7/2013 12:48 PM 9/3/2013 1:07 PM	Type File folder Microsoft
Network	< File name: Save as type: MAT-files (*.mat)	<b>-</b>	> Save Cancel

Figure 227 – The GUI that allows users to save Config Files from this utility. Note the Config File known as "Test1" already saved in this directory. Specifically, note the addition of a date-and-time stamp, as well as "\_CONFIG" appended to the file name.

<b>3</b>	AF_MATB_Cont	figurationUtility_V	300		- 🗆 🗙
AF-MATB System Param V					
	AF-MATB System Parameters				
	Input Options	GUI Buttons Only Keyboard Only Both GUI & Keyboard	Default Value		
	Enable Pausing?	● Yes ○ No	Default Value		
	Enable Manual Event Triggering?	• Yes O No	Default Value		
	Enable Manual Pump Repair?	• Yes No	Default Value		
	Display Date & Time?	● Yes ○ No	Default Value		
	Enable Scheduling?	● Yes ○ No	Default Value		
	Enable Escape key?	Yes      No	Default Value		Save Values
	Enable Home key?	• Yes O No	Default Value		
	Timer Cycling Rate (Hz)	10	Default Value		Load Config File
	Intial Ramp-up Time Delay (Seconds)	5	Default Value		Set All Default Values
	NASA-TLX Ramp-Up Time Delay (Seconds)	5	Default Value	S	ave and Continue to Script Generator
	Resource Management Automation Algorithm	Algorithm 1     Algorithm 2	Default Value	L	oad and Continue to Script Generator
Config File Successfully	Enable Subtask Automation Indicators?	Green V	Default Value		Save and Continue to AF-MATB
Created.	Enable Information Throughput Mode?	○ Yes	Default Value		Load and Continue to AF-MATB
		·			V. 3.00

Figure 228 – The Configuration Utility after the user has successfully saved a Config File, as notified by the status text.

## 8.3.2. Load Config File

When launching the Configuration Utility, users are also able to load previously-created Config or Script Files in order to modify and create a new Config File, or to simply see how the task is configured. To load a file, press the *Load Config File* button, which will launch the file selection GUI illustrated in Figure 224. Using this GUI, the user can click on any Config or Script File and then click the *Open* button to load that file into the utility. Once loading is completed, all parameters will be updated with their new values, and the user will be notified that the process was completed successfully (see Figure 229). In the event that loading was not successfully completed, the user will be notified via a status text (see Figure 230).

Please note that because of the changes made to this utility, Config Files generated with previous versions of this utility are not compatible with this version. Loading them will produce an error.

<b>Ø</b>	AF_MATB_Con	figurationUtility_V	/300	- 🗆 🗙
AF-MATB System Param V				
	- AF-MATB System Parameters			
	Input Options	GUI Buttons Only Keyboard Only Both GUI & Keyboard	Default Value	
	Enable Pausing?	• Yes O No	Default Value	
	Enable Manual Event Triggering?	• Yes • No	Default Value	
	Enable Manual Pump Repair?	• Yes O No	Default Value	
	Display Date & Time?	• Yes No	Default Value	
	Enable Scheduling?	● Yes ○ No	Default Value	
	Enable Escape key?	• Yes O No	Default Value	Save Values
	Enable Home key?	Yes   No	Default Value	
	Timer Cycling Rate (Hz)	10	Default Value	Load Config File
	Intial Ramp-up Time Delay (Seconds)	5	Default Value	Set All Default Values
	NASA-TLX Ramp-Up Time Delay (Seconds)	5	Default Value	Save and Continue to Script Generator
	Resource Management Automation Algorithm	Algorithm 1     Algorithm 2	Default Value	Load and Continue to Script Generator
Files Successfully Loaded.	Enable Subtask Automation Indicators?	Green 🗸	Default Value	Save and Continue to AF-MATB
	Enable Information Throughput Mode?	O Yes  No	Default Value	Load and Continue to AF-MATB
				V. 3.00

Figure 229 – The AF-MATB Configuration Utility notifying the user that the loading process was completed via the status text.

4	AF_MATB_Con	figurationUtility_V	300		- • ×
AF-MATB System Param Y					
	AF-MATB System Parameters				
	Input Options	GUI Buttons Only Keyboard Only Both GUI & Keyboard	Default Value		
	Enable Pausing?	● Yes ○ No	Default Value		
	Enable Manual Event Triggering?		Default Value		
	Enable Manual Pump Repair?	• Yes • No	Default Value		
	Display Date & Time?	Yes      No	Default Value		
	Enable Scheduling?	Yes      No	Default Value		
	Enable Escape key?	• Yes O No	Default Value		Save Values
	Enable Home key?	● Yes ○ No	Default Value		
	Timer Cycling Rate (Hz)	10	Default Value		
	Intial Ramp-up Time Delay (Seconds)	5	Default Value		Set All Default Values
	NASA-TLX Ramp-Up Time Delay (Seconds)	5	Default Value	s	ave and Continue to Script Generator
	Resource Management Automation Algorithm	Algorithm 1     Algorithm 2	Default Value	L	oad and Continue to Script Generator
Config File Damaged	Enable Subtask Automation Indicators?	Green V	Default Value		Save and Continue to AF-MATB
r andre to Load.	Enable Information Throughput Mode?	◯ Yes	Default Value		Load and Continue to AF-MATB
					V. 3.00

Figure 230 – The AF-MATB Configuration Utility, notifying the user that the loading process was not completed via the status text.

162

Distribution A: Approved for public release; distribution unlimited. 88 ABW Cleared 12/03/2014; 88ABW-2014-5667.

## 8.3.3. Set All Default Values

This button, located on the right side of the Configuration Utility, is used to clear the settings for every parameter in the utility and to reset the parameters to their default values. Please note that the event codes in the Port Triggering Parameters group are not reset by this button, as this functionality is reserved for the *Set All Event Codes to Defaults* button discussed in section **8.2.11.10 Set All Event Codes to Defaults**. After pressing this button, the message in the status window in the lower left portion of the screen will read, "All Variables Reset to Defaults."

2	AF_MATB_Cor	nfigurationUtility_V	/300		- 🗆 🗙
AF-MATB System Param V					
	AF-MATB System Parameters				
	Input Options	GUI Buttons Only     Keyboard Only     Both GUI & Keyboard	Default Value		
	Enable Pausing?	• Yes No	Default Value		
	Enable Manual Event Triggering?	• Yes No	Default Value		
	Enable Manual Pump Repair?	• Yes O No	Default Value		
	Display Date & Time?	Yes      No	Default Value		
	Enable Scheduling?	Yes      No	Default Value		
	Enable Escape key?	Yes      No	Default Value		Save Values
	Enable Home key?	Yes   No	Default Value		
	Timer Cycling Rate (Hz)	10	Default Value		Load Config File
	Intial Ramp-up Time Delay (Seconds)	5	Default Value		Set All Default Values
	NASA-TLX Ramp-Up Time Delay (Seconds)	5	Default Value	s	ave and Continue to Script Generator
	Resource Management Automation Algorithm	Algorithm 1	Default Value	L	oad and Continue to Script Generator
All Variables Reset to Defaults	Enable Subtask Automation Indicators?	Green	Default Value		Save and Continue to AF-MATB
	Enable Information Throughput Mode?	Ves No	Default Value		Load and Continue to AF-MATB
		·			V. 3.00

Figure 231 – The AF-MATB System Parameters group after the *Set All Default Values* button has been pressed. Note the status text notifying the user of confirmation that the action was completed.

## 8.3.4. Save and Continue to Script Generator

In some situations, users may want to rapidly load Config Files into the Script Generator for testing. Rather than requiring users to save a Config File, close the Configuration Utility, and then open the Script Generator Utility, users are able to pass information directly into the utility. All of the previous functionality discussed in section **8.3.1 Save Values** applies here, except that after saving, the user will also be notified that the Script Generator will now be loaded (see Figure 232). After a few seconds, the Configuration Utility will close, and the Script Generator Utility will launch, with all of the parameters loaded.

•	AF_MATB_Con	figurationUtility_V	300		- 🗆 🗙
AF-MATB System Param V					
	AF-MATB System Parameters				
	Input Options	GUI Buttons Only Keyboard Only Both GUI & Keyboard	Default Value		
	Enable Pausing?	• Yes O No	Default Value		
	Enable Manual Event Triggering?	● Yes ○ No	Default Value		
	Enable Manual Pump Repair?	Yes      No	Default Value		
	Display Date & Time?	Yes      No	Default Value		
	Enable Scheduling?	Yes      No	Default Value		
	Enable Escape key?	Yes      No	Default Value		Save Values
	Enable Home key?	Yes      No	Default Value		Load Config File
	Timer Cycling Rate (Hz)	10	Default Value		Loud connig rile
	Intial Ramp-up Time Delay (Seconds)	5	Default Value		Set All Default Values
	NASA-TLX Ramp-Up Time Delay (Seconds)	5	Default Value	s	ave and Continue to Script Generator
	Resource Management Automation Algorithm	Algorithm 1     Algorithm 2	Default Value	L	oad and Continue to Script Generator
Loading AF-MATB	Enable Subtask Automation Indicators?	Green V	Default Value		Save and Continue to AF-MATB
Script Generator	Enable Information Throughput Mode?	Yes No	Default Value		Load and Continue to AF-MATB
					V. 3.00

Figure 232 – The Configuration Utility notifying the user that the AF-MATB Script Generator Utility will launch momentarily via the status text.

## 8.3.5. Load and Continue to Script Generator

The functionality of this button is very similar to the *Save and Continue to Script Generator* button, except in this case, users can load a previously created Config or Script File into the Configuration Utility. Just as in the previous section,, the user is notified that the Script Generator Utility is loading after pressing this button (see Figure 232), and when the Script Generator Utility launches, all of the parameters from the Config or Script File will be loaded.

## 8.3.6. Save and Continue to AF-MATB

This functionality is virtually identical to the *Save and Continue to Script Generator* button discussed in section **8.3.4 Save and Continue to Script Generator**, except that in this case, the Configuration Utility will automatically load all of the parameters into the AF-MATB. As usual, users are notified that AF-MATB will launch from the Configuration Utility (see Figure 233). After entering a Participant ID, the user will be notified that a file has been successfully loaded, as discussed in section **7.2.1 Phase 1: Loading Config and Script Files**.

<b>8</b>	AF_MATB_Con	figurationUtility_V	300		- 🗆 🗙
AF-MATB System Param V					
	AF-MATB System Parameters				
	Input Options	GUI Buttons Only Keyboard Only Both GUI & Keyboard	Default Value		
	Enable Pausing?	● Yes ○ No	Default Value		
	Enable Manual Event Triggering?	● Yes ○ No	Default Value		
	Enable Manual Pump Repair?	• Yes No	Default Value		
	Display Date & Time?	● Yes ○ No	Default Value		
	Enable Scheduling?	● Yes ○ No	Default Value		
	Enable Escape key?	Yes      No	Default Value		Save Values
	Enable Home key?	Yes   No	Default Value		
	Timer Cycling Rate (Hz)	10	Default Value		Load Config File
	Intial Ramp-up Time Delay (Seconds)	5	Default Value		Set All Default Values
	NASA-TLX Ramp-Up Time Delay (Seconds)	5	Default Value	s	ave and Continue to Script Generator
	Resource Management Automation Algorithm	Algorithm 1     Algorithm 2	Default Value	L	oad and Continue to Script Generator
Loading AF-MATB	Enable Subtask Automation Indicators?	Green V	Default Value		Save and Continue to AF-MATB
	Enable Information Throughput Mode?	Yes No	Default Value		Load and Continue to AF-MATB
					V. 3.00

Figure 233 – The Configuration Utility notifying the user that AF-MATB will launch momentarily via the status text.

## 8.3.7. Load and Continue to AF-MATB

This functionality is virtually identical to the *Load and Continue to Script Generator* button discussed in section **8.3.5 Load and Continue to Script Generator**, except that in this case, the Configuration Utility will automatically load all of the parameters into the AF-MATB. As usual, users are notified that AF-MATB will launch from the Configuration Utility (see Figure 233). After entering a Participant ID, the user will be notified that a file has been successfully loaded, as discussed in section **7.2.1 Phase 1: Loading Config and Script Files**.

# 9.0 AF-MATB SCRIPT GENERATOR UTILITY

# 9.1. Utility Introduction

Like the Configuration Utility, the Script Generator Utility (see Figure 234) is also a key component that differentiates our software package from other packages, and like the Configuration Utility, the Script Generator Utility has also received a major redesign that allowed us to support the new features added to AF-MATB, as well as add some new features to the Script Generator Utility itself. Ultimately, the purpose of the Script Generator Utility has remained the same as in our previous release, to allow users to quickly and efficiently create and customize scripts.

The following sections will detail the functionality in this utility, and users will be provided with instructions for constructing and modifying Script Files.



Figure 234 - The AF-MATB Script Generator Utility.

# 9.2. 6 Components of the Script Generator Utility

This utility, like the Configuration Utility, was divided into several sections to improve usability. Each component plays a unique role in the construction of a Script File. These components are discussed in detail in the following sections.

## 9.2.1. Variables Currently Loaded

The first component in the Script Generator utility is the "Variable Currently Loaded" section (see Figure 235). This component is comprised of a list of all parameters discussed in section **8.0 AF-MATB CONFIGURATION UTILITY** and a button labeled *Reset Script Generator*.

## 9.2.1.1. Parameters List

The foundation of any script generated using this utility starts with the parameters., so it only makes sense that when the Script Generator Utility is initially launched, all parameters are assigned their default values, as illustrated by the Script Generator Utility's status text located in the lower right of the Script Generator Utility (see Figure 235). These parameters are crucial to the construction of a Script File because the parameters discussed previously (in sections **8.2.3.5 Gauge Malfunction Timeout (Seconds)**, **8.2.3.6 Indicator Light Malfunction Timeout (Seconds)**, **8.2.5.7 Target Communication Timeout (Seconds)**, **8.2.5.8 Communication Event Duration (Seconds)**, **8.2.7 System Monitoring Subtask Automation Parameters Group**, **8.2.8 Tracking Subtask Difficulty Parameters Group**, and **8.2.9.10 Automation Fault Duration (Seconds)**) directly influence the maximum values the user can enter into the "Script Parameters" component (see section **9.2.2 Script Parameters**).

Using this component, users can check any parameter, not just those previously mentioned, to verify its value. This can be especially useful when a custom Config or Script File has been loaded into the Script Generator Utility, either via the Configuration Utility (see sections 8.3.4 Save and Continue to Script Generator or 8.3.5 Load and Continue to Script Generator) or via the main function button (see section 9.2.6.2 Load Config or Script File).

Script Parameters         Condition Name       Condition Length       Communications Subtask       Tracking Subtask       System Mondoring Subtask       Rescurce Management Subtask       Subtask       Component Visibility         Condition 1       Seconds       True Commo       Fable Commo       Difficulty       Automation       Ligits       Gauges       Automation       Fabre Shub-Offs       Task Window       Hiddee         Condition 1       Seconds       Delete Current Condition       Ligits       Gauges       Automation       Fabre Shub-Offs       Task Window       Hiddee         Create New Condition       Delete Current Condition       Tracking Low       Hone       Image: Commo Shub	0	AF_MA	TB_ScriptGeneratorUtili	ty_V300				_ [	×
Condition Name Condition Length Communications Subtask Tracking Subtask System Montoring Subtask Resource Management Subtask S	Script Parameters								
Condition 1       Seconds       True Comms       Difficulty       Automation       Lights       Gauges       Automation       Failures       Stud-offs       Task Window       Hidder         Create New Condition       Delete Current Condition       Pailures       Sub-offs       System Monitoring       Image: Condition Stude of Control of	Condition Name Condition Length Communications Sub	ask Tracking Subtask	System Monitoring Sub	ask	Resource M	lanagement Su	btask	Subtask Component Visibility	
Tracking Low       Itone       None       System Montoring         Create New Condition       Delete Current Condition       FixxSize = 300         CallerD = NGTS04       CenterRange = 8         CommOption = 1       CommSlot1 = NAV1         CommSlot2 = NAV2       CommSlot2 = NAV2         CommSlot3 = COM1       CommSlot4 = COM2         CommSlot4 = COM2       CommSlot4 = COM2         CommSlot4 = COM2       CommSlot4 = So         CommSlot4 = So       CommSlot4 = So         CommSlot4 = So       CommSlot4 = So         CommSlot4 = COM2       CommSlot4 = COM2         CommSlot4 = So       CommSlot4 = So         ComstarShiftVertical = So       Save Script Fie         Load Configo Script Fie       Load & Continue to AF-MATB     <	Condition 1 V Seconds True Comms False C	omms Difficulty	Automation Lights	Gauges	Automation	Failures	Shut-offs	Task Window Hi	dden
Create New Condition       Delete Current Condition         Conditions That Will Comprise This Script       Schedule Current Vulticate Script         CommSlot1 = NAV1       CommSlot2 = NAV2         CommSlot3 = COM1       CommSlot4 = COM2         CommSlot4 = COM2       CommSlot4 = COM2         CommSlot4 = COM2       CommSlot4 = Script         CommSlot4 = Script       Reset Script Generator Utility         Reset Script Generator Utility       Save Script File         Load Config or Script File       Load Config or Script File         Save & Continue to AF-MATB       Load & Continue to AF-MATB		Tracking Low 🗸	None 🗸		None 🗸			System Monitoring 🗸 [	
Delete Selection Pause Duration Pause Duration Delete Selection Default AF-MATB parameters used.	Create New Condition       Delete Current Condition         Conditions That Will Comprise This Script         Conditions That Will Comprise This Script         Schedule Current Condition         Delete Selected Condition         Generate Script	Variables Currently L BoxSize = 300 CallerID = NGT504 CenterRange = 8 CommOption = 1 CommSlot1 = NAV1 CommSlot2 = NAV2 CommSlot3 = COM1 CommSlot4 = COM2 CommSlackoutStatus CommsElackoutStatus CommsElackoutStatus CommsElackoutStatus CommsElackoutStatus CommsElackoutStatus ComsshairShiftVertica Reset So	oaded 8 s = 0 aded) ttal = 50 I = 50 cript Generator Utility		ine Timelin Conditi	Time ne on	Variables Curr BoxSirea 300 CallerD = NGT: CanterRange CommSid1 = NGT: CommSid2 = N CommSid2 = N CommSid4 = C CommSid4 = C C CommSid4 = C C C C S S S C C C S S S S C C S S S C C S S S S C S S C S S S S S S S S S S S S S S S S S S S S	rently Loaded 504 8 1 1AV1 1AV2 50M1 50M1 50M2 50M1 50M2 50 50 50 50 50 50 50 50 50 50	
Custom parameters not loaded.		· ·	Pause Duration	Delete Sele	ection	Save & Cor Defaul Cus	htinue to AF-MA It AF-MA stom para	ATB parameters used ameters not loaded.	³ :.

Figure 235 – The Script Generator Utility, highlighting the "Variables Currently Loaded" component.

167

Distribution A: Approved for public release; distribution unlimited. 88 ABW Cleared 12/03/2014; 88ABW-2014-5667.

## 9.2.1.2. Reset Script Generator Utility

This button allows the user to reset the utility, returning all of the parameters to their default values. It also clears any loaded script parameters (see section **9.2.2 Script Parameters**) and information that was present in any of the other components. After pressing this button, users will be asked to confirm that they want to reset the utility. If they select "Yes," the utility will execute the reset.

#### 9.2.2. Script Parameters

The second component in this utility is the "Script Parameters" section (see Figure 236). This component allows users to specify the exact composition of an unlimited number of conditions. Specifically, users can name conditions, define condition durations, as well as specify the number of event occurrences, the automation states, and the visibility of each of the windows in the AF-MATB.

Fields in this component must contain positive integers (with the exception of the "Condition Name" discussed in section **9.2.2.1 Condition Name** and the "Condition Length" discussed in section **9.2.2.2 Condition Length**). If the user attempts to save a condition or generate a script without meeting the requirements of the "Script Parameters" component, the user will be notified via the "Script Parameter Errors" list.

<b>4</b>	1		AF_MA	TB_ScriptGeneratorUtility_V300		- 🗆 🗙
Script Parameter	rs					
Condition Nam	ne Condition Len	gth Communications Subtask	Tracking Subtask	System Monitoring Subtask	Resource Management Subtask	Subtask Component Visibility
Condition 1	✓ Seconds	True Comms False Comm	s Difficulty	Automation Lights Gauges	Automation Failures Shut-offs	Task Window Hidden
			Tracking Low 🗸	None Y	None 🗸	System Monitoring
Create New (	Condition Delet	te Current Condition	Parameter Errors	•		
ript Parameters ——						
Condition Name	Condition Length	Communications Subtask	Tracking Subtask	System Monitoring Subtask	Resource Management Subtast	k Subtask Component Visibility
ondition 1 🗸	Seconds	True Comms False Comms	Difficulty	Automation Lights Gauge	es Automation Failures Sh	ut-offs Task Window Hidde
			Tracking Low 🗸	None V	None V	System Monitoring 🗸 🗌
Create New Conditio	Delete Cu	rrent Condition	ameter Errors			
	Generate Scrip	<u> </u>		Task Pause     Confirm S     Event Time     Pause Duration	election Save Script File Save & Continue to AF-M Default AF-M Custom par	Load Config or Script File AATB Load & Continue to AF-MATB IATB parameters used. rameters not loaded.

Figure 236 – The Script Generator Utility, highlighting the "Script Parameters" component.

#### 9.2.2.1. Condition Name

The entry field, "Condition Name" dropdown menu, and *Create New Condition* and *Delete Current Condition* buttons are used to manage the number of conditions in the Script Generator Utility (see Figure 237).

Condition Name	
Condition 1	
Create New Condition	Delete Current Condition

Figure 237 – The "Condition Name" portion of the "Script Parameters" component of the AF-MATB Script Generator Utility.

#### 9.2.2.1.1. Entry Field

When the utility is first opened, the "Condition Name" dropdown will display "Condition 1," and the entry field will be empty. Users must enter at least one character into this field in order to meet the naming requirements. It is recommended that users enter a condition name that is brief and descriptive, as this information will be conveyed in the Transition Log discussed in section **7.3.2.13 Transition Log** and in the Condition List discussed in section **9.2.3 Conditions That Will Comprise This Script**.

#### 9.2.2.1.2. "Condition Name" Dropdown

The "Condition Name" dropdown allows the values for the currently-displayed condition to be saved when the user selects a new condition from the dropdown. Alternatively, a user can also save the current condition by selecting its name from the "Condition Name" dropdown. For example, Figure 238 illustrates the user "building" a condition in which the first condition will be labeled "Test1" and will last 300 seconds. Figure 239 shows the user clicking on the dropdown to reveal the available conditions stored in the utility. By clicking on the name of the current condition, "Test1", all values in the "Script Parameters" component will be saved. Then, the name in the dropdown will be updated; in the example illustrated by Figure 240, the name is changed from "Condition1" to "Test1." Finally, a status text will notify the user that the condition has been saved (see Figure 241).

Condition Name	Condition Length	Communications Subtask	
Condition 1 🗸	Seconds	True Comms False Comms	
Test1	300	9 9	

Figure 238 – An example demonstrating the user creating a condition named "Test1."

Condition Name	Condition Length	Communications Subtask
Condition 1 🗸	Seconds	True Comms False Comms
Condition 1	300	9 9

Figure 239 – The user click on the "Condition Name" dropdown to save the values entered into the "Script Parameters" component.

Condition Name	Condition Length	n Communications Subtask	
Test1 ✓	Seconds	True Comms False Comms	
Test1	300	9 9	

Figure 240 – The user has clicked on the condition originally labeled "Condition 1" and updated it. Note how the name in the "Condition Name" dropdown now reads "Test1."

Save Script File	Load Config or Script File
Save & Continue to AF-MATB	Load & Continue to AF-MATB

# Condition Saved

Figure 241 – Status text confirming to the user that the condition was saved.

An additional benefit of the "Condition Name" dropdown is that users can view any conditions that have been stored in the utility, as illustrated in Figure 242 and Figure 243. In Figure 242, the user has clicked on the "Condition Name" dropdown and will select the second condition stored in the utility, known as "Test2." After clicking on "Test2" in the dropdown, the fields in the "Script Parameters" component will be populated with all of the information pertaining to that condition. Note in Figure 243 that after clicking on "Test2," the "Condition Name," "Seconds," "Targets," and "Distractors" fields of the "Script Parameters" component have been updated with the values that correspond to that condition.

Condition Name	Condition Name Condition Length Communications Subta		ons Subtask
Test1 V	Seconds True Comms F		False Comms
Test1	300	9	•
Test2	300		3
Test3			

Figure 242 – The "Condition Name" dropdown, demonstrating 3 conditions that are currently stored in the utility. As illustrated by the blue highlighting and the current condition "Test1," the user is preparing to switch from "Test1" to "Test2."

Distribution A: Approved for public release; distribution unlimited. 88 ABW Cleared 12/03/2014; 88ABW-2014-5667.

Condition Name	Condition Name Condition Length Communications Subta		
Test2 ✓	Seconds	True Comms False Comms	
Test2	200	4 4	

Figure 243 – A section of the "Script Parameters" component after the user has used the "Condition Name" dropdown to switch conditions. Not the change in information in the "Seconds," "Targets," and "Distractors" fields from Figure 242.

## 9.2.2.1.3. Create New Condition Button

The *Create New Condition* button allows the user to create a new slot in the "Condition Name" dropdown, allowing the user to specify parameters for a new condition. When this button is pressed, the utility will first check to ensure that all fields for the current condition in the "Script Parameters" component are valid. In the event that they aren't valid, a new condition will not be created. In the event that all fields do contain valid values (see Figure 244), the information will be saved, just as it would be if the user selected a condition via the dropdown (discussed in section **9.2.2.1.2 "Condition Name" Dropdown**), and a new condition will be immediately created. This condition will automatically be labeled "Condition 2" in the dropdown until otherwise updated, and all information in the fields will be cleared and reset (see Figure 245).

Condition Name	Condition Name Condition Length Communicat		
Condition 1 🗸	Seconds	True Comms False Comms	
Test1	300	9 9	

Figure 244 – A user, ready to create a new condition in the Script Generator Utility.

Condition Name Condition Length		Communications Subtask	
Condition 2 🗸	Seconds	True Comms False Comms	

Figure 245 - The "Script Parameters" component after the user has pressed the Create New Condition button.

## 9.2.2.1.4. Delete Current Condition Button

The *Delete Current Condition* button allows the user to remove the currently selected condition in the "Script Parameters" component of the Scrip Generator Utility. After pressing this button, the currently selected condition will be removed, and a remaining condition will be immediately displayed. Users can delete any condition stored in the utility, even those that have not been completed. If a user elects to delete the last condition stored in the utility, all fields will be cleared and the dropdown will be reset back to "Condition 1," just as it was when the utility was first started (Figure 236).

#### 9.2.2.2. Condition Length

This portion of the "Script Parameters" component of the Script Generator Utility allows the user to specify the duration for the currently selected condition (Figure 246).

Acceptable values for this parameter are real numbers greater than or equal to 30.



Figure 246 – The "Condition Length" portion of the "Script Parameters" component of the AF-MATB Script Generator Utility.

#### 9.2.2.3. Communications Subtask

This portion in the "Script Parameters" component of the Script Generator Utility allows the user to specify the total number of CEs scheduled to occur in a given trial, broken down into TCs and FCs (Figure 247). For more information on this subtask, please see section **3.4 Communications**.

The utility calculates the maximum number of events that can be scheduled based on: the duration of a condition (section **9.2.2.2 Condition Length**), the duration of each audio signal played (section **8.2.5.8 Communication Event Duration (Seconds**)), and the amount of time a participant has to respond (section **8.2.5.7 Target Communication Timeout (Seconds**)). However, due to the nature of this subtask, the "True Comms" and "False Comms" occurrences are not considered discrete events, but instead are treated as sum. Therefore, acceptable values for these both the "True Comms" and "False Comms" fields are positive integers greater than or equal to zero, with the sum of the two values being less than or equal to the total number of possible events determined by the utility.



Figure 247 – The "Communications Subtask" portion of the "Script Parameters" component of the AF-MATB Script Generator Utility.

#### 9.2.2.4. Tracking Subtask

This portion of the "Script Parameters" component of the Script Generator Utility allows the user to specify the difficulty of the Tracking subtask (see Figure 248). Recall that in section **8.2.8 Tracking Subtask Difficulty Parameters Group**, the AF-MATB Configuration Utility allowed users to define the three levels of difficulty for the Tracking Subtask. This dropdown allows users to select one of the three levels they previously defined. Alternatively, users can elect to enable the Automated Mode for the Tracking subtask. For more information on this subtask, please see section **3.3 Tracking**.



Figure 248 – The "Condition Name" portion of the "Script Parameters" component of the AF-MATB Script Generator Utility.

#### 9.2.2.5. System Monitoring Subtask

This portion of the "Script Parameters" component of the Script Generator Utility allows the user to control the System Monitoring subtask. The subtask must first be configured to be in Automated or Manual Mode. If in Manual Mode, the two empty fields will allow the user to set the number of Light occurrences and Gauge occurrences for a given condition, respectively; Manual Mode is illustrated by the "Automation" dropdown in Figure 249 that reads "None." More information about Manual Mode is available in section **3.1.1 Manual Mode**. In Manual Mode, the acceptable values for the Light and Gauge occurrences are positive integers greater than or equal to zero that do not exceed the maximum values calculated by the utility. These values are derived using the condition's length (discussed in section 9.2.2.2 Condition Length) and the subtask's Light timeout (discussed in section 8.2.3.6 Indicator Light Malfunction Timeout (Seconds)) and Gauge timeout values (discussed in section 8.2.3.5 Gauge Malfunction Timeout (Seconds)), respectively. Please note that when in Manual Mode, the Lights and Gauges components are independent, meaning that the number of Light events does not influence the number of Gauge events, or vice versa. Additionally, each object of each component operates on a separate channel, meaning that for the Lights component, both lights may malfunction simultaneously. The same holds true for the Gauges component, which allows for all possible combinations of Gauges to malfunction simultaneously. These multiple channels are what allow for high event densities for this subtask when configured in Manual Mode.



Figure 249 – The "System Monitoring Subtask" portion of the "Script Parameters" component of the AF-MATB Script Generator Utility when configured for Manual Mode.

In Automated Mode, the user is able to control the reliability of the automation by manipulating the number of times the automation successfully detects a problem ("Restores") and the number of times the automation fails to detect a problem ("Failures"). Notice that when the "Automation" dropdown is illustrated in Figure 250 changes from "None" to "Gauges," the "Lights" and "Gauges" labels for the two entry fields change to "Restores" and "Failures" For more information on the Automated Mode, see section **3.1.2 Automated Mode**.

Acceptable values for this portion are positive integers greater than or equal to zero. The maximum number of events for this subtask is derived using the condition's length (discussed in section 9.2.2.2 Condition Length), as well as the restore time (see section 8.2.7.1 Automation Fault Restore Time (Seconds)) and failure duration (see section 8.2.7.2 Automation Failure Duration (Seconds)). As such, the sum of the scheduled "Restores" and "Failures" must not exceed this maximum value, similar to how the maximum is computed for the Communications subtask (see section 9.2.2.3 Communications Subtask).Finally, unlike when the System Monitoring subtask is configured for Automated Mode, only one channel is available for the Gauges, which will not allow the event behaviors of any two Gauges to ever overlap.



Figure 250 – The "System Monitoring Subtask" portion of the "Script Parameters" component of the AF-MATB Script Generator Utility when configured for Automated Mode.

The System Monitoring Subtask is one of two portions of the "Script Parameters" that is restricted depending on the parameters loaded into the utility. If a Config or Script File is loaded into the utility with instructions to enable IT Mode, the utility will restrict the System Monitoring subtask to Manual Mode (see Figure 251)

System Monitoring Subtask			
Automation Lights Gauges			
None	~		

Figure 251 – The "System Monitoring Subtask" portion of the "Script Parameters" component of the AF-MATB Script Generator Utility when IT Mode is enabled.

## 9.2.2.6. Resource Management Subtask

This portion of the "Script Parameters" component of the Script Generator Utility allows the user to control the Resource Management subtask. Like the System Monitoring subtask, users must first configure this subtask to be in Automated or Manual Mode. If in Manual Mode, the two empty fields in this portion will allow the user to set the number of Pump Failures and Pump Shut-Offs, respectively. The Manual Mode, illustrated by Figure 252, shows the "Automation" dropdown reads "None." Further information on the Manual Mode is available in section **3.2.1 Manual Mode**.

For the "Failures" field, acceptable values are positive integers greater than or equal to zero that do not exceed the maximum value calculated by the utility. These values are derived using the condition's length (discussed in section **9.2.2.2 Condition Length**) and the duration of Pump Failures (discussed in section **8.2.6.12 Pump Failure Duration (Seconds**)). Like the System Monitoring subtask when in Manual Mode, each one of the eight pumps in the Resource Management subtask operate on a separate channel, meaning that any combination of the pumps can malfunction simultaneously.



Figure 252 – The "Resource Management Subtask" portion of the "Script Parameters" component of the AF-MATB Script Generator Utility when configured for Manual Mode.

For the "Shut-Offs" field, acceptable values are positive integers greater than or equal to zero. Because these events have no set duration, there is no limit to the number of events the user can schedule. However, if there are more than 5 "Failures" and the number of the "Shut-Offs" is greater than 45% of the number of "Failures," the user will be warned that there is an increased likelihood that a Pump Shut-Off will prematurely end a Pump Failure (see Figure 253).



Figure 253 – The Script Generator Utility warning message when user schedules a large number of Pump Shut-Offs relative to Pump Failures.

When this subtask is in Automated Mode, the "Shut-Offs" field is set to 0 and disabled (see Figure 254) because the Pump Shut-Offs no longer have any impact on the participant. The "Failures" field, however, remains active, though the events scheduled using this field differ based on the parameters currently loaded into the Script Generator Utility. When a Config or Script File is loaded into the utility that configures the Resource Management subtask to use Automation Algorithm 1 (see section 8.2.2.12 Resource Management Automation Algorithm), the "Failures" field allows the user to schedule Pump Failures identical to those when operating in Manual Mode (as discussed in section 3.2.2.1 Automation Algorithm 1). In this situation, the acceptable values for this field are positive integers greater than or equal to zero that do not exceed the maximum value derived using the condition's length (discussed in section 9.2.2.2 Condition Length) and the duration of Pump Failures (discussed in section 8.2.6.12 Pump Failure Duration (Seconds)).

When the Resource Management subtask is configured to use Automation Algorithm 2, the "Failures" field is used to schedule automation failures, as discussed in section **3.2.2.2 Automation Algorithm 2**. Acceptable values are positive integers greater than or equal to zero; though in this case, the maximum limit calculated by this utility is derived from the condition's length (discussed in section **9.2.2.2 Condition Length**) and the duration of the Resource Management's automation failure parameter (discussed in section **8.2.9.10 Automation Fault Duration (Seconds**)), not the Pump Failure duration. And, like the System Monitoring subtask, when in Automated Mode, these failures operate on one channel, meaning that no failures can occur simultaneously.

Resource Management Subtask			
Automation Failures Shut-offs			
Automated V		0	



Finally, this is the second portion of the "Script Parameters" that is restricted depending on the parameters loaded into the utility. If a Config or Script File is loaded into the utility with instructions to enable IT Mode, the utility will restrict the Resource Management subtask to Manual Mode and disable the ability to schedule any events (see Figure 255).

Resource Management Subtask			
Automation	Failures	Shut-offs	
None 🗸	0	0	

Figure 255 – The "System Monitoring Subtask" portion of the "Script Parameters" component of the AF-MATB Script Generator Utility when IT Mode is enabled.

#### 9.2.2.7. Subtask Component Visibility

The "Subtask Component Visibility" portion of the "Script Parameters" component of the Script Generator Utility (see Figure 256) allows the user to set the visibility of each of the six windows of the AF-MATB. The "Task Window" dropdown should be used to select the desired window, and then the user should check the "Hidden" checkbox to ensure that window is hidden during that condition. Users can select any combination of windows to hide.

Please note that the windows that are scheduled to be hidden will be visible when the task is initially loaded, and then they will disappear once the task begins if the user has not set any of the six windows to be initially hidden using the Configuration Utility (see section **8.2.10 Subtask Visibility Parameters**). However, once the trial starts, if a window was scheduled to be hidden, it will disappear. Also note that, if users plans to load a Script File into AF-MATB, they must ensure that the Script File contains instructions to hide those windows even though a user may configure one or more windows to be initially hidden using the Configuration Utility. If not, AF-MATB will load with the windows hidden, but they will reappear when the trial begins.

Subtask Component Visib	ility
Task Window	Hidden
System Monitoring 🗸	

Figure 256 – The "Subtask Component Visibility" portion of the "Script Parameters" component of the AF-MATB Script Generator Utility.

## 9.2.2.8. Script Parameter Errors

The last portion of this component of the Script Generator Utility is a list that allows the user to identify any problems that have prevented the utility from saving the information in the current condition (see Figure 257). If the user has omitted some field in the "Script Parameters" component or entered a value for one field that is an invalid value, this list will detail any errors, by highlighting the problem fields (see Figure 258) and providing descriptions and proposed solutions to any problems (see Figure 259). Please note that for any field that contains data that produces an error in this component, the field label will turn red until the problem has been addressed.

_	- Script Parameter Errors		
		~	
L		+	
-			1

Figure 257 – The "Script Parameter Errors" portion of the "Script Parameters" component of the AF-MATB Script Generator Utility.

_	Script Parameters												
	Condition Name	ion Name Condition Length Communications Subtast		Communications Subtask	Tracking Subtask	System Monitoring Subtask			Resource Ma	inagement Su	ubtask	Subtask Component	t ∨isibility
	Condition 1	-	Seconds	True Comms False Comms	Difficulty	Automation	Lights	Gauges	Automation	Failures	Shut-offs	Task Window	Hidden
	Test1		300	19 19	Tracking Low	None 🗸	8	-1	None 🗸	8		System Monitoring	•
	Create New Co	ondi	tion Delete Cu	urrent Condition	arameter Errors acrease the number of Commun ter an integer greater than or e tter an integer greater than or e	ication occurence qual to zero for thi qual to zero for thi	s. The total n is condition's is condition's	umber of Com System Monit Resource Ma	munication occurren oring Gauge count. nagement Task pum	nces needs t np shut-off pa	o be less than arameter.	n or equal to 23 for this cond	dition.

**Figure 258** – The "Script Parameters" component, illustrating a case in which the user has exceeded the number of possible event occurrences for the Communications subtask, entered an unacceptable value for the Gauges component of the System Monitoring subtask, and omitted the Resource Management subtask's "Shut-Offs" field. Note the text labels for the fields have turned red to highlight the issue.

rences. The total number of Communication occurrences needs to be less than or equal to 23 for this condition.	~
for this condition's System Monitoring Gauge count.	
for this condition's Resource Management Task pump shut-off parameter.	U
	urences. The total number of Communication occurrences needs to be less than or equal to 23 for this condition. o for this condition's System Monitoring Gauge count. o for this condition's Resource Management Task pump shut-off parameter.

Figure 259 – The "Script Parameter Errors" portion of the "Script Parameters" component. Note the verbose error descriptions designed to help the user understand the issue.

Please note that although this "Script Parameter Errors" object is located in the "Script Parameters" component, this error list also serves to notify the user of problems in the "Event Timeline Information" component (see section **9.2.4.1** "Event" Lists and Timeline Modification) and "Schedule Custom Events" component (see section **9.2.5 Schedule Custom Events**).

## 9.2.3. Conditions That Will Comprise This Script

The third component in this utility is the "Conditions That Will Comprise This Script" section (see Figure 260). This component allows the user to select any of the successfully created conditions from the "Script Parameters" component and add them to the text list, known as the condition list. This permits the user to build Script Files that contain multiple conditions.

4				AF_MAT	FB_ScriptGenera	torUtility_V300				- 🗆 🗾
Script Parameters										
Condition Name Condi	tion Length	Communications Subtas	(	Tracking Subtask	System Monito	oring Subtask	Resource Ma	nagement S	ubtask	Subtask Component Visibility
Condition 1 V Se	conds	True Comms False Com	ns	Difficulty	Automation L	ights Gauges	Automation	Failures	Shut-offs	Task Window Hidden
			Trac	cking Low 🗸	None Y		None 🗸			System Monitoring 🗸
Create New Condition Conditions That Will Comprise Schedule Current Condition General	Delete Curr e This Script— Delete Sek	ent Condition	Event Event	Conditions Tha	rent Condition Generat	This Script Delete Select e Script Delete Sel	ted Condition	Save & Cc Defau Cu	Variables Cu BoxSize = 30 CallerD = NG CommSlot1 = CommSlot2 = CommSlot3 = CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot5 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlo	rrently Loaded T504 = 8 = 1 NAV1 NAV2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2

Figure 260 – The Script Generator Utility, highlighting the "Conditions That Will Comprise This Script" component.

In order to use this feature, the user should select the desired condition from the "Condition Name" dropdown. For example, referring back to Figure 242, the user has saved three conditions in the "Script Parameters" component. If the user wants to build a script that contains the condition "Test3," the user simply uses the "Condition Name" dropdown to select "Test3" from the dropdown, and then clicks the *Schedule Current Condition* button and "Test3" will appear as the first item in the condition list (see Figure 261).

The *Schedule Current Condition* button also allows users to save a condition in the "Script Parameters" component just like when they press *Create New Condition* button or use the "Condition Name" dropdown. As illustrated in Figure 262, if a condition is not saved and the *Schedule Current Condition* button is pressed, the condition will automatically be saved and then added to the condition list. If the condition cannot be saved, it will not be scheduled and the errors will be detailed in the "Script Parameter Errors" list (see section 9.2.2.8 Script Parameter Errors).

As previously stated, the condition list allows users to build scripts that contain different conditions. For example, the three conditions saved in the "Script Parameters" section can be different levels of difficulty that increase from "Test1" to "Test3." Instead of having discrete

scripts for each level of difficulty, the user can elect to construct a script that consists of one level of difficulty, immediately followed by another level of difficulty (see Figure 263).



Figure 261 – The Script Generator Utility after the user has pressed the Schedule Current Condition button.

		AF_MA	TB_ScriptGeneratorUtilit	y_V300					
Script Parameters									
Condition Name Condition Length Con	ommunications Subtask	Tracking Subtask	System Monitoring Subt	ask	Resource Ma	anagement S	ubtask	Subtask Component	Visibility
Condition 3 V Seconds True	e Comms False Comms	Difficulty	Automation Lights	Gauges	Automation	Failures	Shut-offs	Task Window	Hidden
Test3 300	9 9	Tracking Low 🗸	None V 9	9	None 🗸	9	9	System Monitoring	<b>~</b>
Create New Condition Delete Current	t Condition	ameter Errors							^ ~
Conditions That Will Comprise This Script	ed Condition	ent Timeline Information Event Time Event Trigger	Event Description	R Show	Change Event T eview Timeline w Complete Timeline w Selected Conditio	ime ? n	Variables Cu BoxSize = 30 CallerID = NG CenterRange CommOption CommSlot2 = CommSlot3 = CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot5 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommSlot4 CommS	rrently Loaded 0 1504 = 8 = 1 NAV1 COM1 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 COM2 C	~
Generate Script		Event Time Event Desc	Event Time Pause Duration	Confirm Se Delete Sel	ection	Sar Save & Co	ve Script File	Load Config or Sc IATB Load & Continue to J	ript File AF-MATB

Figure 262 - The Script Generator Utility, with the user having entered in a third condition that has not yet been saved.

Test1 Test2	^
	~
Schedule Current Condition	Delete Selected Condition
Generate	e Script

— Conditions That Will Comprise This Script ——

Figure 263 – The "Conditions That Will Comprise This Script" component, illustrating a script with two different conditions.

Alternatively, the user can construct a script with multiple repetitions of the same condition, allowing for performance to be split into different blocks. For example, if the user is interested in having a participant complete a 10-minute trial, but is also interested in breaking down their performance by 2-minute epochs, one efficient approach might be to save a 2-minute condition in the "Script Parameters" component and then schedule five of these conditions, back-to-back (see Figure 264).

The advantage to this approach is twofold. First, as discussed in any of the "Multiple Condition" sections (see **7.3.1 Performance Directory Types**), it allows AF-MATB to perform the analysis

181

for the researcher. By using the performance summaries contained in the "Condition" directories generated by AF-MATB as the analysis for each epoch, and the performance summary located in the root as an analysis of overall performance, researchers can use the analysis tools of AF-MATB to do the work for them. Second, by using blocked conditions, users ensures that the distribution of events for each epoch is identical, as opposed to using one 10-minute block where equal numbers of events are not guaranteed to exist in each epoch, making comparison between epochs more difficult.

reata	^
Test3	
Test3	
Test3	
Test3	
	~
Schedule Current Condition	Delete Selected Condition
Schedule Current Condition	Delete Selected Condition

Figure 264 – The "Conditions That Will Comprise This Script" component, illustrating a script with five repeated conditions.

The Script Generator Utility requires that at least one condition be present in the condition list in order to generate a script. If the user presses the *Generate Script* button without any conditions in the condition list, the user will be notified that no script will be generated (see Figure 265).

![](_page_207_Picture_4.jpeg)

Figure 265 – Script Generator Utility notification when the user attempts to generate a script with no conditions.

Once one condition is in the list, pressing the *Generate Script* button will construct a script based on all of the parameters defined in the "Variables Currently Loaded" component, as well as the "Script Parameters" component. The details of the script will be displayed in the "Event Timeline Information" component (see section **9.2.4 Event Timeline Information**) when completed. There is no limit to the number of conditions the user can schedule, though large numbers of conditions may increase the amount of time required to generate the script.

In order to delete any condition from the condition list, the user should simply click on that condition (see Figure 266) in the list and press the *Delete Selected Condition* button. That

condition will then be removed, as is the case in Figure 267. The user is able to use the *Delete Selected Condition* button to remove any and all conditions form the list.

Finally, please note that any modification to any field in the "Script Parameters" or "Conditions That Will Comprise This Script" components will result in the clearing of both the "Conditions That Will Comprise This Script" and "Event Timeline Information" components. This ensures that a script is generated using the most current information.

	,
Test2	
Test3	
Test1	
Schedule Current Condition	Delete Selected Condition
Schedule Current Condition	Delete Selected Condition

Figure 266 – The user clicking on the second of four conditions in the condition list.

- Conditions That Will Comprise	This Script	
Test1 Test3 Test1		
	~	/
Schedule Current Condition	Delete Selected Condition	
Generate	e Script	

Figure 267 – The condition list after the user has deleted the second of four previously listed conditions.

#### 9.2.4. Event Timeline Information

The fourth component in this utility is the "Event Timeline Information" section (see Figure 268). This component allows the user to select any event after the user has pressed the *Generate Script* button and edit its onset time, providing the user with complete control over the script.

This component consists of several objects; the "Event Time," "Event Trigger," and "Event Description" lists, the *Change Event Time* button and its associated entry field, the *Review Timeline* button, and the Timeline Selector.

![](_page_209_Figure_0.jpeg)

Figure 268 – The Script Generator Utility, highlighting the "Event Timeline Information" component.

#### 9.2.4.1. "Event" Lists and Timeline Modification

After a script is generated, the utility populates the "Event Time," "Event Trigger," and "Event Description" lists with all the details relevant to that script (see Figure 269). The user is able to scroll through any of the three lists and click on any item of interest.

— Event Timeline	Info	rmation —				
Event Time		Event Trigg	jer	Event Description		
0.01	^	49	^	Tracking Difficulty Low	^	Change Event Time
0.01		129		System Monitoring Automation		
0.01		32		Resource Management Autom		
0.01		106		Subtask Visible: System Monit		
0.01		107		Subtask Visible: Tracking		Review Timeline
0.01		108		Subtask Visible: Communicatio		
0.01		109		Subtask Visible: Resource Ma		
0.01		110		Information Visible: Scheduling		Show Complete Timeline
0.01		111		Information Visible: Pump Statu	~	Show Selected Condition
13.5386	۷	54 '	<b>~</b>	< >		

Figure 269 – The "Event Timeline Information" component after a script was generated.

Clicking on an item will highlight the corresponding information on the other two lists, and populate the empty field with the event time shown in the "Event Time" list. As shown in Figure 270, after the user clicked on the "Event Time" item "37.2507", the "Event Trigger" and "Event Description" lists jumped to the corresponding values, meaning that at 37.2507 seconds, a Light 2 (also known as the red light in the System Monitoring subtask) Fault with the event code "6" occurs. Also note that the empty field now contains the "Event Time" value selected by the user.

Event Time	Event Trigger	Event Description		
0.01 0.01 13.5386 21.9201	106 ^ 107 108 109	Subtask Visible: Tracking Subtask Visible: Communicatio Subtask Visible: Resource Ma Information Visible: Scheduling	37.2507	Change Event Time
30.3475 37.2507 51.2108 54.9926	110 111 54 74 78	True Communication Played: N False Communication Played: N False Communication Played: N Light 2 Fault	Show	Complete Timeline

Figure 270 - The "Event Timeline Information" component after the user has clicked on the "Event Time" item.

This component provides the user with the option to change any event onset. For example, in order for a user to change the event onset from 37.2507 seconds to 10 seconds, the user clicks on the item, and then enters an acceptable value (10) into the field (see Figure 271) and then presses the Change Event Time button. After pressing the event button, the "Event Time," "Event Trigger," and "Event Description" lists will reset back to the top, and a status text will notify the user that the event time was successfully updated (see Figure 272).

Event Time	;	Event Tri	gger	Event Description			
0.01 0.01 13.5386 21.9201 30.3475 37.2507	^	106 107 108 109 110 111	^	Subtask Visible: Tracking Subtask Visible: Communicatio Subtask Visible: Resource Ma Information Visible: Schedulin <u>c</u> Information Visible: Pump Statu True Communication Played: N	^	10 Rev	Change Event Time
51.2108 54.9926 55.3223 69.6333	~	54 74 78	~	False Communication Played: 1 False Communication Played: 1 Light 2 Fault	~	<ul> <li>Show</li> <li>Show</li> </ul>	Complete Timeline Selected Condition

Figure 271 – The user is about to change the event time of from 37.2507 to 10 seconds.

			AF_MA	TB_ScriptGene	ratorUtili	ty_V300				_ 0	
ript Parameters											
Condition Name Condition I	ength Communication	ns Subtask	Tracking Subtask	System Mo	nitoring Sub	task	Resource Management Subtask			Subtask Component Visibility	
est1 v Second	is True Comms F	alse Comms	Difficulty	Automation Lights Gau		Gauges	Automation	Failures Shut-offs		Task Window Hide	idder
Test1 300	9	9	Tracking Low 🗸	None 🗸	9	9	None 🗸	9	1	System Monitoring 🗸	
Create New Condition D	elete Current Condition	Script Pa	rameter Errors								
nditions That Will Comprise Thi 11 12	s Script	C C C C C C C C C C C C C C C C C C C	ent Timeline Information Event Time Event Trigger 11 1 129 12 129 12 106 11 107 107 108 109 11 108 109 110 109 110 109 110 100 109 110 100 10	Event Desc Tracking Difficulty / System Monitoring Resource Manager Subtask Visible: Cr Subtask Visible: Cr Subtask Visible: Cr Subtask Visible: Riformation Visible: Information Visible: <	cription Automation ment Autom system Monit acking pommunicatio ssource Ma Scheduling Pump Stati	Ri V Shor	Change Event eview Timeline w Complete Timeli w Selected Condit	. Time ne tion	Variables Cu BoxSize = 30 CallerID = NG CenterRange CommOption CommSlot1 = CommSlot2 = CommSlot4 = CommSlot4 = CommSlot4 CommSlot4 = CommSlot4 CommSlot4 = CommSlot4 = CommSlo4 = Comm	rrently Loaded 0 1504 = 8 = 1 NAV1 NAV2 COM1 COM2 ogths = 8 butStatus = 0 butStatus = 0 butStatus = 50	
chedule Current Condition	elete Selected Condition		Schedule Custom Events	ription Task Pa	ause 🗸	]			CrossnairShi	Reset Script Generator Utility	
				Eve	ent Time	Confirm Se Delete Sel	lection	Save & C	ave Script File ontinue to AF-N	Load Config or Script File	в
			~	v Paus	e Duration				Event	Time Updated	

Figure 272 – The Script Generator Utility after an event's onset time was modified.

Please note that when changing an event's onset time, if the user enters an invalid value, the user will be notified via the "Script Parameter Errors" list in the "Script Parameters" component. In the example illustrated in Figure 273, the user is attempting to change an event onset time from ".01" to "-jk9k." After pressing the *Change Event Time* button, the "Event Time," "Event Trigger," and "Event Description" lists will reset back to the top, but the "Script Parameter Errors" list will notify the user to enter a time that is within the appropriate range (Figure 274).

Also note that an event that occurs in one condition cannot be moved into any other condition. The range shown in Figure 275 shows that an event that occurs in the first condition can only be changed to a time between 0 and 299.

Additionally, the Script Generator Utility will adjust the upper limit of the acceptable time range to ensure that the event is completed before the condition ends. For example, if the user attempts to modify the onset of a Light event with a 5-second timeout to after the condition ends, the upper limit of the acceptable time range will be changed to 294 (see Figure 276).

The last and most critical thing to note about modifying an event's onset is that this is done at the user's risk, because this functionality does not verify that these modifications will not result in events overlapping. Overlapping two identical events may produce AF-MATB instability.

Event Time		Event Trig	gger	Event Description			
0.01	^	49	^	Tracking Difficulty Low	^	-jk9k	Change Event Time
0.01		129		System Monitoring Automation			
0.01		32		Resource Management Autom			
0.01		106		Subtask Visible: System Monit			
0.01		107		Subtask Visible: Tracking		Rev	view Timeline
0.01		108		Subtask Visible: Communicatio			
0.01		109		Subtask Visible: Resource Ma			
0.01		110		Information Visible: Scheduling		Show	Complete Timeline
0.01		111		Information Visible: Pump Statu	~	O Show	Selected Trial
10	$\mathbf{v}$	6	$\sim$	< >		0.000	

Figure 273 – Value the user is attempting to update that event's onset time to.

Script Parameters	ndition Length Seconds 300 Delete Cu	Communications True Comms Fak	Subtask e Comms 9 – Script Pa	Tracking S Difficu Tracking Low	Subtask Itty	System Automation None Y	Monitoring Sul Lights	ntask Gauges	Resource Ma	nagement S Failures	ubtask Shut-offs	Subtask Component Vi	sibility
Condition Name Cor Test1  Create New Condition	ndition Length Seconds 300 Delete Cu	Communications True Comms Fals	Subtask e Comms 9 - Script Pa Please e	Tracking S Difficu Tracking Low	Subtask lity	System Automation None V	Monitoring Sul	otask Gauges	Resource Ma	nagement S Failures	ubtask Shut-offs	Subtask Component Vi	sibility
Test1	Seconds 300 Delete Cu	True Comms Fals	e Comms 9 – Script Pa Please et	Difficu Tracking Low	v.	Automation	Lights	Gauges	Automation	Failures	Shut-offs	Task \&indow	
Test1 Create New Condition	300 Delete Cu	9	9 – Script Pa	Tracking Low	*	None 🗸				Failures Shut-offs		Task Window H	Hidden
Create New Condition	Delete Cu	rrent Condition	– Script Pa Please er	warmatan Ermana			None 🗸 9		None 🗸	<b>v</b> 9		System Monitoring	
				nter a number betv	ween 0 and 29	99 to modify this	time.						<b>^</b>
- conations That Will Compr Test1 Test2	rise This Script	~	0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.	Vent Time Inform Event Time 01 01 01 01 01 01 01 01 01 01 01 01 01	mation Event Trigger 49 129 32 106 107 108 109 110 111 6 ¥	Event D Tracking Difficu System Monitori Resource Mana Subtask Visible: Subtask Visible: Subtask Visible: Information Visible Information Visible	escription ty Low ng Automation gement Autom System Monit Tracking Communicatic Resource Ma ble: Scheduling ble: Pumo Stat >	▲ 0.01	Change Event T eview Timeline w Complete Timelini w Selected Condition	ime 2 n	<ul> <li>variables Cur</li> <li>BoxSize = 300</li> <li>CallerID = NGT</li> <li>CenterRange</li> <li>CommSlot1 = 1</li> <li>CommSlot2 = 1</li> <li>CommSlot3 = (</li> <li>CommSlot4 = (</li> <li>CommTimeLer</li> <li>CommSlat4 = (</li> <li>CommSlat4 = (</li> <li>CommTimeLer</li> <li>CommSlat4 = (</li> <li>CommSlat4 = (</li></ul>	entry Loaded 504 504 1 IAV1 IAV2 COM1 COM2 gths = 8 Status = 0 n = (Loaded) Horizontal = 50	^
Schedule Current Conditio	on Delete Se erate Script	elected Condition		Schedule Custom I Event Time	Events	ription Task	Pause				F	eset Script Generator Utility	·
				~		Pa	Event Time	Confirm Se	ection	Sav Save & Co	ve Script File	Load Config or Scrip ATB Load & Continue to AF Not Updated	t File -MATB

Figure 274 – The Script Generator utility after the user enters an invalid value.

_	- Script Parameter Errors							
	Please enter a number between 0 and 299 to modify this time.	~						
		$\mathbf{v}$						
L '		_						

Figure 275 – Close-up of the "Script Parameter Errors" list after the user attempted to change an event onset time to an invalid value.

- Script Parameter Errors							
^							
• •							

Figure 276 – Another close-up of the "Script Parameter Errors" list after the user attempted to change an event onset time to an invalid value, in this case illustrating a different acceptable upper limit to account for an event's duration.

187

Distribution A: Approved for public release; distribution unlimited. 88 ABW Cleared 12/03/2014; 88ABW-2014-5667.

#### 9.2.4.2. Timeline Selector and Review Timeline Button

The Timeline Selector allows the user to select what information is displayed in the "Event Time," "Event Code," and "Event Description" lists will show all of the information for a script, regardless of how many conditions make up the script. However, this may not be ideal if the user is interested in the arrangement of events for a specific condition. As a result, by changing the selector from "Show Complete Timeline" to "Show Selected Condition," the "Event Time," "Event Code," and "Event Description" lists will be updated to show only the information for the condition selected in the "Conditions That Will Comprise This Script" component, allowing the user to look at only the desired information. In Figure 277, because "Test2" is selected in the "Event Time," "Event Code," and "Event Description" lists in the "Event Timeline Information" component only show from 300 seconds on, which is when the "Test2" condition starts. By clicking on "Test1" in the condition list, the "Event Time," "Event Code," and "Event Time," "Event Code," and "Event Time," show the "Event Time," "Event Code," and "Event Description" lists in the "Event Time Information" component only show from 300 seconds on, which is when the "Test2" condition starts. By clicking on "Test1" in the condition list, the "Event Time," "Event Code," and "Event Description" lists will update to only show up to 300 seconds, which is the end of that condition (see Figure 278).

![](_page_213_Figure_2.jpeg)

Figure 277 – The "Event Timeline Information" component showing a partial timeline after the user has selected the "Show Selected Condition" button.

![](_page_213_Figure_4.jpeg)

**Figure 278** – The "Event Timeline Information" component showing a partial timeline after the user has selected the "Show Selected Condition" button. In this figure, the user clicked on "Test1" in the condition list and then scrolled to the bottom of the event information lists to show that the lists are only showing the timeline for the first condition.

While the "Event Time," "Event Trigger," and "Event Description," lists allow the user to see how a script is constructed before saving it, sometimes the user needs to see the bigger picture when reviewing the event distributions, which is exactly what the *Review Timeline* button

188

affords the researcher. Pressing this button will launch a series of histograms designed to provide the user with a visual representation of how events are dispersed throughout time in the script. These histograms are constructed with the x-axis being the time of the condition or trial, and the y-axis being the number of events that occur in 5-second bins.

The order of the presentation for the histograms will follow the "Script Parameters" component of the Script Generator Utility. For instance, the Communications subtask histogram will be displayed first (see Figure 279), followed by the System Monitoring subtask's Lights (see Figure 280) and Gauges (see Figure 281) components, and then finally the Resource Management subtask's Pump Failures (see Figure 282) and Pump Shut-Offs (see Figure 283). However, the Script Generator Utility can use the information in the "Script Parameters" component to determine whether or not to skip certain histograms. For example, if the user does not schedule any CEs in a script, then the System Monitoring Lights component will be the first histogram presented. Also, if the user is reviewing a condition that has configured the System Monitoring subtask for Automated Mode, the Lights and Gauges histograms will be suppressed in lieu of a histogram that shows the users the automation "Restores" and "Failures" (see Figure 284). This would also be the case for the Resource Management subtask, if it was configured to be in Automated Mode using Automation Algorithm 2 (see Figure 285).

A histogram is displayed for each task component and a dialog box is displayed that allows the user to accept or reject the event distribution (see Figure 286). If the user rejects the event distribution, the Script Generator Utility will then generate a new event distribution for just that component. In other words, if the user approves the event distribution of the Communications subtask, but rejects the distribution of the Gauges component of the System Monitoring subtask, then the events in the Communications subtask and all other distributions will be preserved and the events in the Gauges component of the System Monitoring subtask will be redistributed. This example is illustrated in Figure 287. There is no limit to the number of times a user can reject a distribution.

![](_page_215_Figure_0.jpeg)

Figure 279 – The event distribution for the Communications subtask. Note the different colors used to indicate TCs and FCs.

![](_page_215_Figure_2.jpeg)

Figure 280 – The event distribution for the Lights component of the System Monitoring subtask.

Distribution A: Approved for public release; distribution unlimited. 88 ABW Cleared 12/03/2014; 88ABW-2014-5667.

190






Figure 282 – The event distribution of the Pump Failures for the Resource Management subtask.

191



Figure 283 - The event distribution of the Pump Shut-Offs for the Resource Management subtask.



Figure 284 – The event distribution of the System Monitoring subtask when in Automated Mode.

192



Figure 285 – The event distribution of automation failures for the Resource Management subtask when in Automated Mode using Automation Algorithm 2.



Figure 286 – The distribution approval dialog box.



Figure 287 – The event distribution of the Gauges component of the System Monitoring subtask's after the user disapproved of the initial distribution.

Finally, users should note that it can be difficult to use the *Review Timeline* function when the "Show Complete Timeline" option is selected and the script is comprised of multiple conditions. The distributions lose some resolution as the number of conditions and overall duration of the script increase, making them more difficult to read (see Figure 288). Additionally, when reviewing the complete timeline, rejecting any distribution will generate a new one for all conditions in that distribution (see Figure 289).

As an alternative, the user can review the event distributions and approve them on a conditionby-condition basis by using the "Show Selected Condition" option in the Timeline Selector. This can be accomplished by selecting the "Show Selected Condition" button from the Timeline Selector, clicking on the desired condition in the "Conditions That Will Comprise This Script" condition list, and then pressing the *Review Timeline* button.



Figure 288 - The event distribution for the Communications subtask over a 30-minute trial.



Figure 289 – The event distribution of the Pump Failures for the Resource Management subtask across two conditions. One condition shows a favorable distribution while the other does not.

195

#### 9.2.5. Schedule Custom Events

The "Schedule Custom Events" section (see Figure 290) allows the user to add unique events such as task pauses and the NASA-TLX into a script that was already generated, allowing users to control when and for how long the task will pause and when to launch a NASA-TLX.

			AF_MA	TB_ScriptGeneratorUtility_V300		
Script Parameters						
Condition Name Con	dition Length	Communications Subtask	Tracking Subtask	System Monitoring Subtask	Resource Management Sul	btask Subtask Component Visibility
Condition 1 🗸	Seconds	True Comms False Comm	s Difficulty	Automation Lights Gauges	Automation Failures	Shut-offs Task Window Hidde
		Schedu	le Custom Events	ription		System Monitoring V
Create New Condition	Delete Cu	Irrent Cond	^	↑ Task Pause ✓		
- Conditions That Will Compr	ise This Script		~	Pause Duration	Confirm Selection	Variables Currently Loaded           BixXii:ze = 300         CallerD = NGT504           CallerD = NGT504         CenterRange = 8           CommSlot1 = NAV1         CommSlot2 = NAV2           CommSlot3 = COM1         CommSlot3 = COM1           CommSlot4 = COM2         CommSlot4 = COM2           CommSlot4 = COM2         CommSlot4 = 0           CommSlot4 = 0         CommSlot4 = 0           CommSlot4 = 0         CommSlot4 = 50
Schedule Current Conditio	n Delete Se	elected Condition	- Schedule Custom Events Event Time Event Des	iption		CrosshairShiftVertical = 50 Reset Script Generator Utility
Gene	rate Script			Task Pause v Confirm Event Time Pause Duration Delete Se	election Save & Con Bection Defaul Cus	e Script File Load Config or Script File tinue to AF-MATE Load & Continue to AF-MATE It AF-MATB parameters used. tom parameters not loaded.

Figure 290 – The Script Generator Utility, highlighting the "Schedule Custom Events" component.

In order to schedule a pause in the task, the user must select the "Task Pause" option from the dropdown menu. Then, the user should enter the desired onset time into the "Event Time" field and the desired duration into the "Pause Duration" field (see Figure 291). Finally, the user should press the *Confirm Selection* button. The information will then be added to the "Event Time" and "Event Description" lists in the "Schedule Custom Events" component (see Figure 292).



Figure 291 – The user entering in information for a task pause before pressing the *Confirm Selection* button.

196

- Schedule Custom	Events		
Event Time	Event Description		
300 🔨	Task Pause Schedu 🔺	Task Pause 🗸	
			Confirm Selection
		Event Time	
			Delete Selection
		Pause Duration	
	× ×		
¥	× >		

Figure 292 – The user successfully entering information for a task pause.

Please note that a script must already be generated before users can schedule pauses in the task. Additionally, the value entered in the "Event Time" field must be within the duration of that generated script. If invalid values are entered in either the field, the user will be notified in the "Script Parameter Errors" portion of the "Script Parameters" component (see Figure 293), discussed in section **9.2.2.8 Script Parameter Errors**.



Figure 293 – The Script Generator Utility after the user unsuccessfully attempts to add a custom event.

In order to add a NASA-TLX to a script, the user must first select the "NASA-TLX" option from the dropdown menu, then enter the desired onset time into the "Event Time" field (see Figure 294), and then press the *Confirm Selection* button. The information will then be added to the "Event Time" and "Event Description" lists in the "Schedule Custom Events" component (see Figure 295) The "Pause Duration" label and field will disappear while the "NASA-TLX" option is selected.

Г	- Schedule Cu	ston	Events				
	Event Time	в	Event Description				
	300	^	Task Pause Schedu	^	NASA-TLX	¥	
							Confirm Selection
					Event Time		
					450		Delete Selection
				-			
		~	< >				

Figure 294 – The user entering information for a NASA-TLX before pressing the *Confirm Selection* button.

Γ	Schedule Cu	ston	Events			
	Event lime	Э.,	Event Description	٦.		
	300	^	Task Pause Schedu	^	NASA-TLX V	
	450		NASA-TLX Schedu			
						Confirm Selection
					Event lime	
						Dalata Calculia
						Delete Selection
				~		
		5				
		*				

Figure 295 – The user successfully entering information for a NASA-TLX.

In order to delete a custom event, the user should click on the desired item from either the "Event Time" or "Event Description" lists in the "Schedule Custom Events" component and then press the *Delete Selection* button.

### 9.2.6. Main Function Buttons

This last component in the AF-MATB Script Generator Utility contains the buttons that save and load Script Files into the utility, as well as the buttons that interface with the AF-MATB.

<b>0</b>	AF_MA	TB_ScriptGeneratorUtility_V300				
- Script Parameters						
Condition Name Condition Length Communications	Subtask Tracking Subtask	System Monitoring Subtask	Resource Management S	ubtask	Subtask Component Visibility	
Condition 1 V Seconds True Comms Fals	e Comms Difficulty	Automation Lights Gauges	Automation Failures	Shut-offs	Task Window Hidden	
	Tracking Low	None 🗸	None	Sy	vstem Monitoring 🗸	
Create New Condition Delete Current Condition	Script Parameter Errors				^ ~	
Conditions That Will Comprise This Script	Save Script File	Load Config or Script	t File	Variables Currently Loaded BoxStze = 300 CallerD = NGTS04 CenterRange = 8		
	Save & Continue to AF-I	MATB Load & Continue to AF-	матв	CommOption = 1 CommSlot1 = NA <sup>1</sup> CommSlot2 = NA <sup>1</sup> CommSlot3 = COI CommSlot4 = COI	in = 1 = NAV1 2 = NAV2 3 = COM1 4 = COM2	
· ·	Default AF-M	IATB parameters u	insed.	CommTimeLength CommsBlackoutS CommsEquation CrosshairShiftHo	ns = 8 status = 0 = (Loaded) rizontal = 50	
Schedule Current Condition Delete Selected Condition	Custom pa	rameters not loade	d.	CrosshairShiftVe	ertical = 50	
Generate Script	Event Time Event Desc	Task Pause V		Res	et Script Generator Utility	
		Event Time	lection Sa	ve Script File	Load Config or Script File	
		Pause Duration	Save & Co	Save & Continue to AF-MATB Load & Continue to AF-		
		v	Defat Cu	ilt AF-MA stom paran	TB parameters used. neters not loaded.	

Figure 296 – The Script Generator Utility, highlighting the utility's main function buttons and status text notifying the user of the utility's current state.

# 9.2.6.1. Save Script File

After the user has generated a script, is satisfied with the event distributions of that script, and has added any required custom events, the user simply needs to save the Script File. After pressing the *Save Button*, the user will be presented with a file saving GUI, which will allow the user to enter a name for the Script File (see Figure 297). Just like the saving procedure discussed in section **8.3.1 Save Values**, after entering a file name into the "File name" field and pressing the *Save* button in the GUI, the Script Generator Utility will automatically append a date-and-time stamp at the end of the filename. Additionally, when saving a script, the date-and-time stamp and \_"SCRIPT" are appended to the file name, just as Config Files are appended with "\_CONFIG" (see Figure 297).

After pressing *Save*, the GUI will close, and the Script Generator Utility will begin saving two files. The first is an Excel file (see Figure 298) that is discussed in section **9.3 Excel Script Details**. Once the Excel file has been successfully saved, a .mat file will also be saved, which contains all information stored in the utility, including all of the parameters and script-related information. After the .mat file has been successfully saved, the user will be notified and the utility will be ready to use (see Figure 299).

	Save Script As		×
Save in:	Scripts 💌	⊨ 🗈 💣 📰 -	
<u>Ca</u>	Name	Date modified	Туре
Recent places	AF-MATB System Files	9/8/2013 11:21 PM 9/8/2013 12:36 AM	File folder Microsoft
Desktop		5, 0, 2015 12150 f an	
Libraries			
Computer			
Network			
			,
	File <u>n</u> ame:	<u> </u>	Save
	Save as type: MAT-files (*.mat)	•	Cancel

Figure 297 – File saving GUI launched by the Script Generator Utility when the Save Script File or Save & Continue to AF-MATB buttons are pressed.

				AF_MA	FB_ScriptGene	ratorUtili	ty_V300					
Script Parameters												
Condition Name Cond	tion Length	Communication	ns Subtask	Tracking Subtask	System Mo	nitoring Sub	task	Resource N	fanagement S	Subtask	Subtask Component	Visibility
Test1 v S	econds	True Comms F	alse Comms	Difficulty	Automation	Lights	Gauges	Automation	Failures	Shut-offs	Task Window	Hidden
Test1	300	9	9	Tracking Low 🗸	None 🗸	9	9	None 🗸	9	1	System Monitoring	•
Create New Condition	Delete Cu	rrent Condition	Script Pa	arameter Errors								~
Conditions That VAII Compris Fest1 Fest2	e This Script		Ev 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.	Apple to Timeline Information           Event Time         Event Trigger           01         49         A           01         129         32           01         106         107           01         108         107           01         109         101           01         110         109           01         111         6	Event Des Tracking Difficulty System Monitoring Resource Manage Subtask Visible: T Subtask Visible: R Information Visible formation Visible C	cription Low Automation ment Autom vstem Monit acking ommunicatio seource Ma Scheduling Pump Stati >	R Shor Shor	Change Event	Time	Variables Cu BoxSize = 30 CallerID = NG CenterRange CommOption CommSlot1 = CommSlot2 = CommSlot4 = Co	rrently Loaded 0 1504 = 8 = 1 NAV1 NAV2 COM1 COM2 COM2 COM2 com4 com5 com4 com5 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com6 com	~
Schedule Current Condition	Delete Se	elected Condition	-9	Schedule Custom Events	intion					CrosshairShi	ftVertical = 50 Reset Script Generator Utili	ty V
Gener	ate Script			Event lime Event Desci 300 ^ Task Pause S 450 VASA-TLX Sc < <	iption ihedu Task P Ev Paus	ent Time	Confirm Se	lection	Sa Save & Co	ve Script File ontinue to AF-N Writin	Load Config or So IATB Load & Continue to g Excel File	cript File

Figure 298 - The Script Generator Utility after the Save Script File and Save & Continue to AF-MATB buttons are pressed.



Figure 299 – The Script Generator Utility after the script has been successfully saved.

# 9.2.6.2. Load Config or Script File

The *Load Config or Script File* button allows users to load files into the Script Generator Utility. By loading a Config File into the utility, the user introduces different governing parameters. These parameters, particularly those that determine exactly how many events can fit in a given length of time (see section **9.2.2 Script Parameters**), are critical to the creation of a script. Other parameters are used to ensure that the appropriate options are conveyed through the script when loaded into AF-MATB.

By loading a Script File, the user automatically loads all of the parameters in a Config File, and all of the information stored in each of the other four components of the Script Generator Utility. This provides users with the most detailed look at a script, as well as the ability to make modifications and resave the scripts, or to build new scripts based on previously-constructed conditions.

In order to load a file, users must first press the *Load Config or Script File* button and use the file selection GUI to select the desired file (see Figure 300). If the desired file was a Config File, the Script Generator Utility will appear as it does in Figure 301. If any information was stored in the "Script Parameters" or any other components of the utility, that information will be cleared. This is because the new parameters loaded via the Config File may change the values already deemed acceptable in the "Script Parameters" component, as well as invalidate any scripts that may have been generated. By purging all of this information, the utility ensures that scripts are generated with only the most current information. Therefore, it is recommended that before building any conditions in the "Script Parameters" component, the user should always load any custom parameters first.

-	Select File to Open		×
Look in:	Scripts	← 🗈 💣 📰 -	
Ca	Name	Date modified	Туре
Recent places	AF-MATB System Files Test_09-Sep-2013_00-09-20_SCRIPT	9/8/2013 11:21 PM 9/9/2013 12:09 AM	File folder Microsoft
Desktop	Test_09-Sep-2013_01-36-51_CONFIG	9/9/2013 1:36 AM	Microsoft
Libraries			
Computer Computer Network			
	<		>
	File name:	<b>•</b>	Open
	Files of type: MAT-files (*.mat)	•	Cancel

Figure 300 – File selection GUI launched if the user presses the *Load Config or Script File* or *Load & Continue to AF-MATB* buttons.

			AF_MA	TB_ScriptGeneratorL	tility_V300				
cript Parameters									
Condition Name Co	ondition Length	Communications Subtask	Tracking Subtask	System Monitoring	iubtask	Resource Ma	anagement Suk	otask	Subtask Component Visibility
Condition 1 🗸	Seconds	True Comms False Comms	Difficulty	Automation Lights	Gauges	Automation	Failures	Shut-offs	Task Window Hidde
			Tracking Low 🗸	None 🗸		None 🗸			System Monitoring 🗸 🗌
Create New Condition	Delete Cu	rrent Condition	Parameter Errors						
nditions That Will Com	orise This Script		Event Timeline Information Event Time Event Trigger	Event Description	R Sho Sho	Change Event T eview Timeline w Complete Timelini w Selected Condition	Fime e on	Variables Curri BoxSize = 300 CallerID = NGT5 CenterRange = CommSlot1 = N. CommSlot2 = N. CommSlot3 = C. CommSlot4 = C. CommSlac4 = C. CommSBlackou. CommsBlackou. ComssblarShift	ently Loaded 04 8 1 AV1 AV2 OM1 OM2 ths = 8 tStatus = 0 istatus = 0 istatus = 50
ichedule Current Condi Ge	tion Delete Senerate Script	elected Condition	Schedule Custom Events	ription Task Pause	~			Re	ertical = 50
			v	Event Time	Confirm Se Delete Se	ection	Save Save & Con	e Script File tinue to AF-MA Dnly Para	Load Config or Script File Load & Continue to AF-MATB meters Loaded!

Figure 301 – The Script Generator Utility after the user has loaded a Config File.

When a Script File is loaded, the Script Generator Utility should load the associated parameters, display the information in the appropriate fields, and notify the user that the file was successfully loaded (see Figure 302).

Please note that because of the changes made to this utility, Config or Script Files generated with previous versions of the Configuration or Script Generator utilities are not compatible with this version. Electing to load one of these files or any damaged file will produce an error (see Figure 303).

			AF_MA	TB_ScriptGener	atorUtility	/_V300					
- Script Parameters											
Condition Name Condition	on Length Corr	mmunications Subtas	Tracking Subtask	System Mor	itoring Subta	sk	Resource Ma	anagement S	ubtask	Subtask Component V	isibility
Test1 V Sec	onds True	e Comms False Comm	ns Difficulty	Automation	Lights	Gauges	Automation	Failures	Shut-offs	Task Window	Hidden
Test1 3	00	9 9	Tracking Low 🗸	None 🗸	9	9	None 🗸	9	1	System Monitoring	•
Create New Condition	Delete Current	t Condition	Parameter Errors								<b>^</b>
Test1 Test2	The Schol	~	Event Time         Event Tinger           0.01         0.01           0.01         129           0.01         106           0.01         107           0.01         108           0.01         109           0.01         108           0.01         108           0.01         108           0.01         108           0.01         108           0.01         108	Event Descr Tracking Difficulty L System Monitoring A Resource Managem Subtask Visible: Syr Subtask Visible: Tra Subtask Visible: Tra Subtask Visible: Re Information Visible: Information Visible:	iption www.kutomation ent Autom stem Monit cking mmunicatio source Ma Scheduling Pump Stati >	Re Show Show	Change Event T wiew Timeline v Complete Timelini v Selected Condition	ime a n	BoxSize = 30 CallerID = NG CenterRange CommSlot1 = CommSlot2 = CommSlot3 = CommSlot4 = CommSlot4 = CommSlot4 = CommSlot4 = CommSlot4 = CommSlot4 =	0         0           7504         5           7504         5           8         1           NAV1         0           COM1         COM1           COM2         COM2           rgths = 8         14/t5tatus = 0           on = (Loaded)         Horizontal = 50	^
Schedule Current Condition	Delete Selecter	ed Condition	Schedule Custom Events Event Time Event Dest 300 ASA-TLX S	cription	use V				CrosshairShi	tVertical = 50 Reset Script Generator Utility	·
				Eve	nt Time	Confirm Sel	ection	Sava 8 Cr	ve Script File	Load Config or Scri	pt File
			v <	> Pause	Duration	Delete Sele		Save & Cl	Scr	ipt Loaded!	r-maib
											V

Figure 302 – The Script Generator Utility after a Script File was successfully loaded.



Figure 303 – The Script Generator Utility after the user attempt to load a corrupt file.

205

### 9.2.6.3. Save & Continue to AF-MATB

Just as with the AF-MATB Configuration Utility (discussed in section **8.3.6 Save and Continue to AF-MATB**), this function allows the user to rapidly test a Script File by passing that file directly into the AF-MATB. After being notified that the Script File was successfully created, as was discussed in section **9.2.6.1 Save Script File**, a status text will also notify the user that AF-MATB will launch (see Figure 304). The Script Generator will then disappear, and after entering in a Participant ID, the user will be notified that AF-MATB has already successfully loaded a file, as discussed in section **7.2.1 Phase 1: Loading Config and Script Files**.

A				A	F_MA	TB_ScriptGene	eratorUtili	ty_V300					
Script Parameters													
Condition Name Condit	ion Length	Communications	Subtask	Tracking Subtas	sk	System M	onitoring Subt	ask	Resource M	anagement S	Subtask	Subtask Componer	nt ∀isibility
Test1 V Se	conds	True Comms Fal	se Comms	Difficulty		Automation	Lights	Gauges	Automation	Failures	Shut-offs	Task Window	Hidden
Test1	100	9	9	Tracking Low	*	None 🗸	9	9	None 🗸	9	1	System Monitoring	<b>~</b>
Create New Condition	Delete Cur	rrent Condition	- Script Pa	arameter Errors									<b>^</b>
— Conditions That Will Comprise Test1 Test2	This Script-	~	Ev 0.0 0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1	vent Timeline Information Event Time Event 01 49 01 129 01 106 01 107 01 107 01 109 01 101 01 109 01 101 01 109 01 101 01 109 01 101 01 109 01 101 01 109 01 101 01 101 01 109 01 100 01 101 01 109 01 101 01 109 01 101 01 109 01 100 01 109 01 100 01 100 00 100 01 100 00	Trigger	Event Des Tracking Difficuty System Monitoring Resource Manage Subtask Visible: T Subtask Visible: T Subtask Visible: C Subtask Visible: A Information Visible Information Visible Information Visible	cription Low / Automation ment Autom ystem Monit racking ommunicatio esource Ma c Scheduling Pump Statu >	R Shor Shor	Change Event T eview Timeline w Complete Timelin w Selected Condition	Fime	Variables Cr BoxSize = 3 CallerID = NC CenterRang CommSlot1 = CommSlot2 = CommSlot3 = CommSlot4 = CommSlot4 = CommSlot4 = CommSlac4 CommSlac4 CommSlac4 CommSlac4	urrently Loaded 00 37504 = 8 = 1 NAV1 NAV2 = cOM1 = cOM2 engths = 8 coulStatus = 0 tion = (Loaded) tion = 50	
Schedule Current Condition General	Delete Se te Script	elected Condition	-	Schedule Custom Event: Event Time Eve 300 Task 1	s ent Descr Pause Sc	ription	ause V	]			CrosshairSh	iffVertical = 50 Reset Script Generator Ut	v
				450 NASA	L-TLX Sc	hedu Ev	ent Time	Confirm Se Delete Sel	lection	Save & Co	ve Script File ontinue to AF-I Loadin	Load Config or 1	Script File o AF-MATB

Figure 304 – The Script Generator Utility, notifying the user that AF-MATB is loading via the status text.

### 9.2.6.4. Load & Continue to AF-MATB

Just as with the AF-MATB Configuration Utility (discussed in section **8.3.7 Load and Continue to AF-MATB**), this function allows the user to load a previously created Script File from the Script Generator utility into AF-MATB. After loading the file, as discussed in section **9.2.6.2** Load Config or Script File, a status text will notify the user that AF-MATB will launch (see Figure 304). Following this message, the Script Generator Utility will disappear. Just as was the case in section **9.2.6.3 Save & Continue to AF-MATB**, after entering a Participant ID, the user will be notified that a file has been successfully loaded, as discussed in section **7.2.1 Phase 1:** Loading Config and Script Files.

Please note that users can only use this function with Script Files. Unlike the loading functionality discussed in section **9.2.6.2 Load Config or Script File**, this functionality restricts the ability to load Config Files. If a Config File is loaded, a status text will notify the user that the parameters from that file will be loaded, but the utility will not launch AF-MATB (see Figure 305)

4			AF_MA	TB_ScriptGenerator	Itility_V300					- 🗆 🗙
Script Parameters										
Condition Name	Condition Length	Communications Subtask	Tracking Subtask	System Monitoring	Subtask	Resource M	lanagement Subl	task	Subtask Component ∀i	sibility
Condition 1 🗸	Seconds	True Comms False Comm	s Difficulty	culty Automation Lights Gauges			Automation Failures SI		s Task Window I	
			Tracking Low 🗸	None 🗸		None 🗸		Sy	stem Monitoring	•
Create New Condi	ion Delete Cu	urrent Condition	Parameter Errors							<b>^</b>
Conditions That Will C	omprise This Script	elected Condition	Event Timeline Information Event Time Event Trigger	Event Description	F Sho Sho	Change Event	Time CC	Variables Current IoxSize = 300 AllerID = NGT504 EnterRange = 8 commOption = 1 commSlot1 = NAV commSlot2 = NAV commSlot2 = NAV commSlot3 = COM commSlot3 = COM commSlackout5 commSlackout5 commSlackout5 commsSlackout5 commsSlackout5 commsSlackout5 commsSlackout5 commsSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout5 commSlackout	tly Loaded /1 /2 12 12 13 14 15 15 16 16 16 16 16 16 16 16 16 16	~
	Generate Script		Event Time Event Desc	Task Pause	~			Rese	et Script Generator Utility	
				Event Time	Confirm Se	lection	Save Save & Conti	Script File	Load Config or Scrip	T-MATB
		_	<b>v</b>	Pause Durati	on		O AF	nly Paran 7-MATB v	neters Loaded! vill not launch	V. 3.0

Figure 305 – The Script Generator Utility after the user attempts to use the *Load & Continue to AF-MATB* button to load a Config File.

# 9.3. Excel Script Details

As discussed in section **9.2.6.1 Save Script File**, whenever a Script File is saved from the Script Generator Utility, the first file that is created is an Excel Workbook. This workbook contains two sheets designed to allow the researcher to review all relevant information pertaining to the Script File without requiring MATLAB or the Script Generator Utility. The first sheet, labeled "Actual Timeline," is comprised of 5 columns of data:

- a. <u>Event Time (Seconds)</u>: The values in this column correspond to the onset time of the events. These values should be identical to the values contained in the "Event Time" list of the "Event Timeline Information" component of the Script Generator Utility.
- b. <u>Script ID Code:</u> These values are used by the Script Generator Utility to understand and organize events. These codes translate to the following:
  - i. 12: Communication Event
  - ii. 61 & 62: Lights component of the System Monitoring subtask
  - iii. 71 74: Gauges component of the System Monitoring subtask
  - iv. 91 98: Resource Management subtask Pump Failures and Automation Failures
  - v. 101 108: Resource Management subtask Pump Shut-Offs
  - vi. 111-114: System Monitoring Automation Restores
  - vii. 121 124: System Monitoring Automation Failures
  - viii. 199: NASA-TLX
  - ix. 200: Task Pause
  - x. 345: Tracking subtask difficulty
  - xi. 678: System Monitoring subtask Automation status
  - xii. 890: Resource Management subtask Automation status
  - xiii. 123456: Subtask Visibility Events

Please note that these numbers do not necessarily translate to the items for subtasks that contain more than one object. For example, 71 does not necessarily equate to Gauge 1, 72 to Gauge 2, etc. Rather, these numbers are placeholders that allow the Script Generator Utility's algorithms to group events into their respective subtasks, while still ensuring that each subtask maintains its own channel of events when applicable.

- c. <u>Item Identifier:</u> Used to provide detail about exactly which object each number in the <u>Script ID Code</u> field is assigned to. For example, if an event's <u>Item Identifier</u> is 71, but its <u>Script ID Code</u> is 4, then 71 actually translates to the Gauge 4 (F4). These codes can also be used to tell the user which item in the Tracking, System Monitoring, and Resource Management dropdown lists was selected. The codes also represent the 6 windows of AF-MATB (System Monitoring, Tracking, Communications, Resource Management, Scheduling, and Pump Status) for the numbers 1-6. Finally, this number can represent the duration of a task pause if scheduled through the script.
- d. <u>Event Trigger:</u> The code read by the AF-MATB that instructs the task to execute the desired behavior.
- e. <u>Stop Time:</u> Where applicable, this field shows the time at which an event is scheduled to time-out. For events that have no associated time-out, such as Pump Shut-Offs or FCs, this number is 0.

The second sheet, titled "Event Translation Key," provides a description for behaviors that occur in the AF-MATB when the number from the <u>Event Trigger</u> field of the "Actual Timeline" sheet is read by the task.

#### References

- Comstock, J.R., & Arnegard, R.J. (1992). The Multi-Attribute Task Battery for human operator workload and strategic behavior research. NASA TM-104174, National Aeronautics and Space Administration, Langley Research Center, 1992.
- Hart, S.G., & Staveland, L.E. (1988). Development of a multi-dimensional workload scale: Results of empirical and theoretical research. In P. A. Hancock & N. Meshkati (Eds.), *Human mental workload* (pp. 139-183). Amsterdam: North-Holland.
- Molloy, R. & Parasuraman, R. (1996). Monitoring an automated system for a single failure: Vigilance and task complexity effects, *Human Factors*, *38*(2), 311-322.
- Phillips, C.A., Kinsler, R., Repperger, D.W., Mandal, J., Neidbard-Doll, A., Kender, D. (2013). A human-machine interaction strategy function: information throughput and weighting with application to Multiple-Attribute-Task-Battery, *Theoretical Issues in Ergonomic Science*, 14(4), 379-401.
- Phillips, C., Repperger, D., Kinsler, R., Bharwani, G. & Kender, D. (2007). Quantitative model of the Human-Machine Interaction and Multi-Task Performance: A strategy function and the Unity Model Paradigm, *Computers in Biology and Medicine*, *37*(9), 1259-1271.
- Phillips, C.A. & Walters, C.M. (2013). User's Guide: Air Force Multiple Attribute Task Battery (AF-MATB-IT). AFRL-RH-WP-TR-2013.
- Phillips, C.A., Walters, C.M., & McKinley, A. et al. (2013). Information throughput model for the resource management component of the Multiple-Attribute Task Battery (MATB). *Aviation, Space, and Environmental Medicine*, 84(4), 336.
- Walters, C.M. (2012). Application of the human-machine interaction model to Multiple Attribute Task Battery (MATB): Task component interaction and the strategy paradigm. Unpublished master's thesis. Wright State University, Dayton, OH.

# Glossary

711 <sup>th</sup> HPW/RHCPA	711 <sup>th</sup> Human Performance Wing/Human Effectiveness Directorate,
	Warfighter Interface Division, Applied Neuroscience Branch
AFRL	Air Force Research Laboratory
AF-MATB	Air Force Multi-Attribute Task Battery
CE	Communications Event
DAQ	Digital Acquisition
DPI	Dots Per Inch
FC	False Communication
GUI	Graphical User Interface
HMI	Human-MATB Interaction
HOIM	Human Operator Informatic Model
MCR	MATLAB Compiler Runtime
NASA	National Aeronautics and Space Administration
OS	Operating System
RAM	Random Access Memory
RMSD	Root Mean Square Deviation
TC	True Communication