ELECTRONIC COMBAT SIMULATION IN A NETWORKED, FULL MISSION REHEARSAL, MULTI-SIMULATOR ENVIRONMENT

David W. Galloway, Patrick G. Heffernan, E. Allen Nuss, Charles M. Summers
TRW Avionics & Surveillance Group, Warner Robins Avionics Laboratory
Warner Robins, Georgia

ABSTRACT

The Integrated Electronic Combat Simulation System (IECSS) has been developed for the MH-53J and MH-60G Weapon System Trainers (WSTs) and is under development for the HC-130P Aircrew Training Device (ATD). This system provides dynamic simulation of the closed loop Electronic Combat (EC) environment to support multiship operations for eight networked training systems. The IECSS simulates: (1) The electromagnetic and infrared environment; (2) Threat weapons dynamics and engagement including basic C3 characteristics; (3) Electronic warfare (EW) defensive system processing and environment interaction; (4) Countermeasures effectiveness calculations; and (5) EW systems audio and video interface to the aircrew. The level of fidelity for this simulation is sufficient to accommodate mission rehearsal for qualified aircrews in addition to programmed, repeatable training to qualify or upgrade new aircrew members.

The IECSS real-time software for one WST is hosted on a single VME chassis with multiple 68030 CPUs, and these general purpose processors communicate with the simulation host computer through shared memory. The IECSS has been developed using a building-block approach which separates threat modeling. In this way, enhancements can be made to any model without significant impact to other existing modes. The software suite also includes off-line editors and diagnostic tools in addition to the real-time functions. Off-line threat setup involves populating a file structure which contains threat laydown and characterization data. New threats are easily added to the database through menu-driven editors.

An Inter-Simulation Network (ISN) connects up to eight IECSS-equipped trainers through a fiber-optic based reflective memory technique. All eight players share a common electromagnetic simulation but individually process the environment based upon position, occulting, and defensive models assigned. This enables the WSTs/ATDs to mission rehearse or fly in formation through a consistent environmental laydown.

ABOUT THE AUTHORS

David Galloway is currently a lead hardware engineer at TRW. He was the Systems Engineer whose responsibilities included technical management of the software and hardware team which designed and implemented the IECSS. As a Research Engineer at Georgia Tech, Mr. Galloway's responsibilities included designing and implementing real-time digital signal processing, data acquisition/control hardware and software for a phased array receiver. Mr. Galloway received a M.S.E.E. in 1988 and a B.S.E.E. in 1986 from Auburn University.

Pat Heffernan is currently a project manager at TRW. He was the lead engineer for the IECSS threat simulation. Mr. Heffernan was a USAF Electronic Warfare Officer for six years. He flew operationally with the 8th Special Operations Squadron (MC-130E). Mr. Heffernan received a B.S. in 1985 from the US Air Force Academy and another B.S. in 1990 from the University of West Florida.

Al Nuss is currently a project manager at TRW. He was the lead engineer for the IECSS EW systems simulations. Mr. Nuss was an EW engineer for the USAF and was responsible for all aspects of the ALQ-187, ALQ-101 and QRC 80-01 electronic countermeasures systems. Mr. Nuss received his B.S. in Aerospace Engineering from Purdue University and an M.S. in Management from Troy State University.

Chad Summers is currently a project engineer at TRW. He was the lead engineer for the IECSS off-line support system (database management system). Mr. Summers was a software communications engineer responsible for digital communications of LAN devices. He received a B.S.E.E. in 1986 from Auburn University.

ELECTRONIC COMBAT SIMULATION IN A NETWORKED, FULL MISSION REHEARSAL, MULTI-SIMULATOR ENVIRONMENT

David W. Galloway, Patrick G. Heffernan, E. Allen Nuss, Charles M. Summers
TRW Avionics & Surveillance Group, Warner Robins Avionics Laboratory
Warner Robins, Georgia

INTRODUCTION AND SYSTEM OVERVIEW

The Integrated Electronic Combat Simulation System (IECSS) provides a full electronic combat (EC) simulation capability for the MH-53J Pave Low and MH-60G Pave Hawk Weapon System Trainers (WSTs). A follow-on effort currently under development will install the same capability on the HC-130P Aircrew Training Device (ATD). All three of the aircrew training simulators are installed at the USAF 542nd Crew Training Wing, Kirtland Air Force Base, Albuquerque, New Mexico.

The basic system requirement for IECSS was to provide the government with the capability of realistically simulating a full EC scenario in real-time using a cockpit mockup as the user interface for the aircrew. An extension to this basic requirement was to provide the capability of supporting multiple WSTs "playing" in the same gaming scenario. In other words, the threat simulation software would have the capability of encountering any one of up to eight WSTs/ATDs.

The basic architecture of the WSTs/ATDs, shown in Figure 1, implements the common EC environment as a shared memory data structure

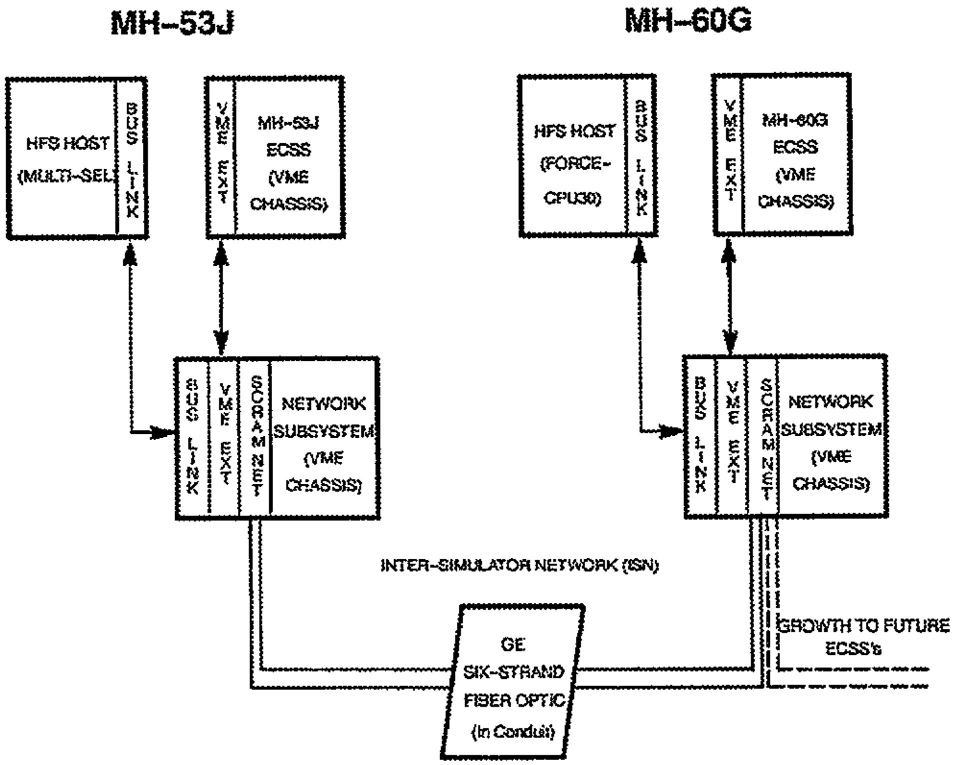
MH-53J
MH-60G
HFS HOST (MULTI-SEL
BUS LINK
VME EXT
MH-53J ECSS (VME CHASSIS)
HFS HOST (FORCE-CPU30)
BUS LINK
VME EXT
MH-60G ECSS (VME CHASSIS)
BUS LINK
VME EXT
SCRAMNET
NETWORK SUBSYSTEM (VME CHASSIS)
BUS LINK
VME EXT
SCRAMNET
NETWORK SUBSYSTEM (VME CHASSIS)
INTER-SIMULATOR NETWORK (ISN)
GE SIX-STRAND FIBER OPTIC (In Conduit)
GROWTH TO FUTURE ECSS's

Figure 1 Weapon System Architecture

between WSTs/ATDs via the Inter-Simulator Network (ISN); a fiber optic based reflective memory. It should be noted that the IECSS nomenclature implies the simulation of multiple, integrated Electronic Combat Simulation Systems (ECSS). In a multiship, networked simulation, one of the ECSSs is declared as master of the entire threat simulation environment. It is the master's responsibility to assimilate all scenario information from the slave ECSSs to derive an active threat list. During the generation of the active threat list, the master is also responsible for determining targeting assignments against all active aircraft on the network.

Each ECSS also provides its respective WST/ATD with the ability of simulating all of the electronic warfare (EW) avionics installed on the aircraft in addition to providing a local threat simulation capability. The threat simulation will support a scenario of 400 unique threats with a maximum of 64 active at any given instant in time. The simulation also provides a closed looped simulation of countermeasures effectiveness using the best available effectiveness data.

The ECSS software architecture, shown in Figure 2, is based on the concept of simulating the EC environment using a shared memory data structure modeled as the electromagnetic environment. This data structure, referred to as the Common Environment Data (CED), contains the information pertinent to describing the battlefield environment to the level of fidelity required for a training simulator. The CED includes structures for storing active threats, active emitters, active weapons, and any active countermeasures (CM) being applied to the scenario. The CM data structures provided include RFCM, IRCM, and chaff and flare expendables.

To fully support the expansion and flexibility of the IECSS design, a full suite of off-line

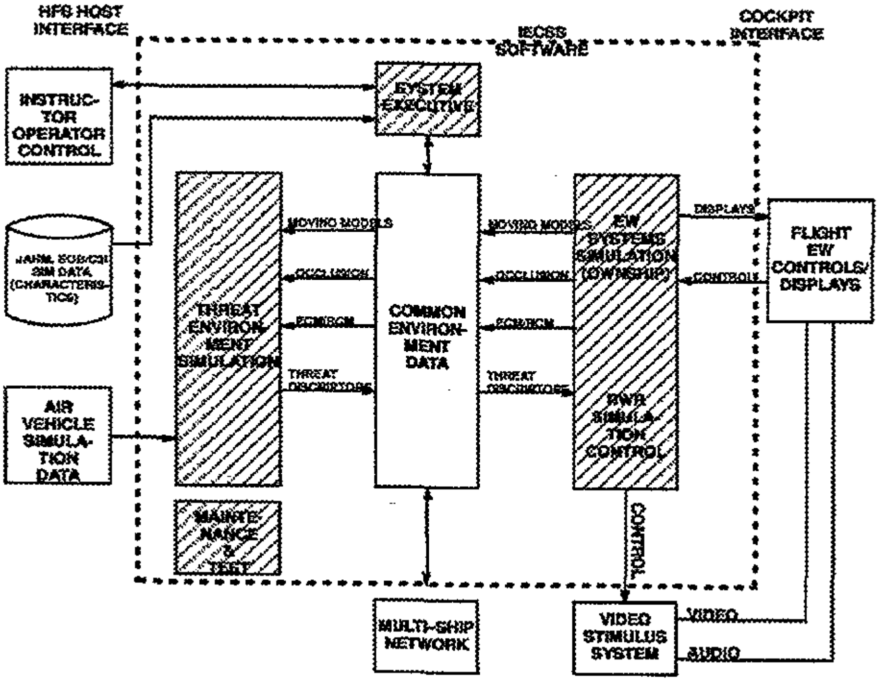

Figure 2 ECSS Software Architecture

database support tools were developed to provide the end-user with the capability of maintaining the databases required for IECSS. These tools provide the user with the capability of generating new and updating existing threat simulation parametrics. It also provides the ability for generating a scenario mission load. Each mission load contains up to 25 different mission scenarios or laydowns. Each laydown defines the location of a maximum of 400 unique threats and defines the desired Command, Control, and Communications (C3) network for each scenario.

THREAT SIMULATION

The threat simulation models activate and engage the WSTs/ATDs in a controlled manner in order to simulate actual battlefield threat environments. Although there are many threat simulation models available, the unique nature of networked aircraft operating over large areas and distances compounds the threat simulation problem because each WST/ATD must contend with its own subset of threats. In addition, the threat simulation model must be independent of other EW simulation functions so that it can be used for any WST/ATD. The IECSS threat simulation model is one approach to solving the networked problem while providing a generic model.

The threat simulation models begin with a mission load which is built off-line by the user. The mission load contains information such as threat scenario laydowns, threat parametrics, weapon parametrics, C3 information, and other EW-related data. The mission load is read into real-time data structures during initialization. An important real-time data structure is the threat laydown. The threat laydown data structure contains threat identification and positional data. This data structure is available to every WST/ATD via the ISN, thus, the threat laydown serves as the starting point for all real-time threat functions.

When the network is being operated in the master-slave role, the master WST/ATD is responsible for generating a master threat environment from the slaves' local threat environments, executing tactics algorithms for the threats in the master threat environment, generating airborne interceptor flight paths, and managing certain commands from the instructor operator station (IOS). The slave WSTs/ATDs are responsible for generating their own local threat environments, producing weapon flyouts, and reacting to certain inputs from the IOS. When the WST/ATD is operating in the standalone mode, then all functions are accomplished by the standalone ECSS.

Figure 3 describes the general approach for production of the master electronic warfare

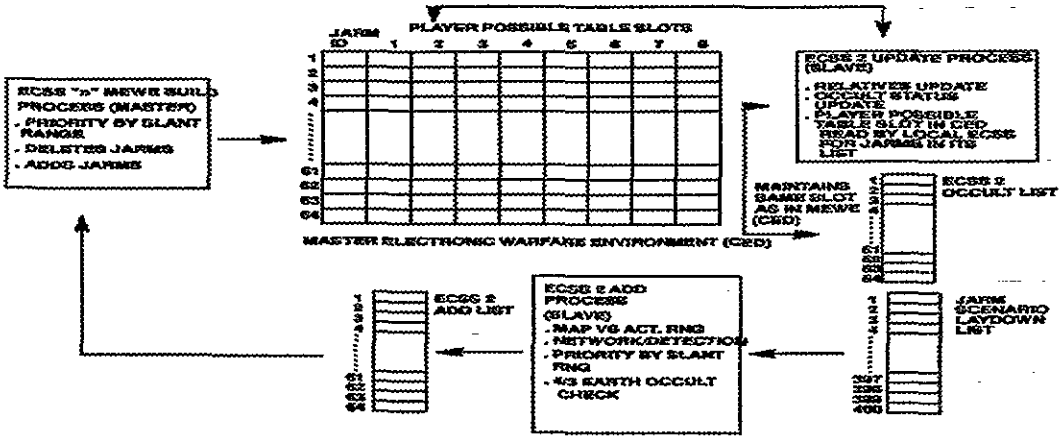
ECSS "n" MEWE BUILD PROCESS (MASTER)
. PRIORITY BY SLANT RANGE
. DELETES JARMS
. ADDS JARMS
JARM ID
PLAYER POSSIBLE TABLE SLOTS
1 2 3 4 5 6 7 8
1 2 3 4 61 62 63 64
MASTER ELECTRONIC WARFARE ENVIRONMENT (CED)
ECSS 2 UPDATE PROCESS (SLAVE)
. RELATIVES UPDATE
. OCCULT STATUS UPDATE
. PLAYER POSSIBLE TABLE SLOT IN CED READ BY LOCAL ECSS FOR JARMS IN ITS LIST
MAINTAINS SAME SLOT AS IN MEWE (CED)
ECSS 2 OCCULT LIST
1 2 3 4 61 62 63 64
JARM SCENARIO LAYDOWN LIST
1 2 3 4 397 398 399 400
ECSS 2 ADD PROCESS (SLAVE)
. MAP VS ACT. RNG
. NETWORK/DETECTION
. PRIORITY BY SLANT RNG
. 4/3 EARTH OCCULT CHECK
ECSS 2 ADD LIST
1 2 3 4 61 62 63 64

Figure 3 Master Electronic Warfare Environment

(threat) environment (MEWE). Each slave WST/ATD produces a local threat environment consisting of up to 64 threats from a maximum threat laydown of 400 threats during the add process. It was necessary to reduce the number of potential threats to a manageable limit to meet real-time limitations in processing. The reduction of potential threats from the laydown to the local threat environment (add list) is accomplished by comparing map ranges from the WST/ATD to each threat in the laydown to a threat activation range, checking initial occulting (terrain masking) information, and exercising initial C3 data. In addition, threats already in the MEWE are ignored by the add process. The resulting add list is prioritized by slant range, from the threat to the WST/ATD, and the excess threats over 64 are eliminated from further consideration.

The add lists are obtained by the master over the ISN. The master takes each add list and prioritizes each threat in the add lists by slant range. Only the 64 highest priority threats are slotted into the MEWE, which is part of the overall CED structure. The master also deletes threats from the MEWE when they are occulted from their contributing WST/ATD for a predetermined time period or when the WST/ATD leaves the threat's activation range.

After the MEWE is constructed, the master is responsible for target assignment. A WST/ATD is listed as a player possible when a threat affects it. From this player possible list, the master determines, based upon slant range, which WST/ATD will be targeted by a specific threat. Threats continue to target a WST/ATD until it is shot down, occulted, or leaves the activation range of the threat.

After target assignment, the targeted WST/ATD ECSS calculates the relative positional information (relative azimuths, relative elevations, etc.) for the threat and requests occulting information from the Digital Radar Land Mass Simulator (DRLMS) during the update process. This information is then transmitted back to the MEWE for use by the master threat simulation. The process is then repeated at a resolution required to support the overall system fidelity.

Each threat is built off-line using a series of tactics algorithms which dictate how the threat operates in the simulation. The master executes these algorithms for each threat in the MEWE depending upon positional factors, ECM/ECCM factors, and probability of detection. Atmospheric factors, such as range attenuation, rain, visibility, and terrain clutter affect the probability of detection. The master also determines when a threat fires/launches at the targeted aircraft. The goal was to make this part of the threat simulation as representative of the actual threat system operation, so that there was little to no perceivable difference to the aircrew between the actual and simulated threat environments.

An essential part of the threat simulation is the C3 system. The threats can be defined in the scenario as either autonomous or networked. An autonomous threat operates by itself based upon its own activation range and parameters. Networked threats rely upon other threats in the network, called controllers, to activate them. The controllers are defined in the mission load. Networked threats rely on the controllers to detect the incoming targets and activate them. Activation, engagement, and weapon firing are dependent upon the level of conflict selected for a threat scenario. The instructor can define the level of conflict off-line and then modify it during the mission rehearsal session through the IOS.

The master also generates the airborne interceptor (AI) flight paths. During the mission load build, AIs can be defined for the simulation. These AIs, during real-time, automatically activate and engage a targeted WST/ATD. The flight path generation routines are six DOF models and the information they provide is not only used in the threat simulation, but transferred to the Computer Image Generator (CIG) so that the AI can be presented visually in the simulation.

Each WST/ATD ECSS calculates, as stated above, relative positional information for a threat that could affect it (player possible). The relatives consist of aircraft azimuth and elevation relative to the WST/ATD X,Y,Z coordinate system, relative azimuth and elevation relative to true north (using a Cartesian coordinate system), map (X,Y) and slant (X,Y,Z) ranges, aircraft radar cross section (RCS) and infrared (IR) signature, and threat radar/IR field of view checks and power densities at the WST/ATD. The relatives are used throughout the threat and EW equipment simulations.

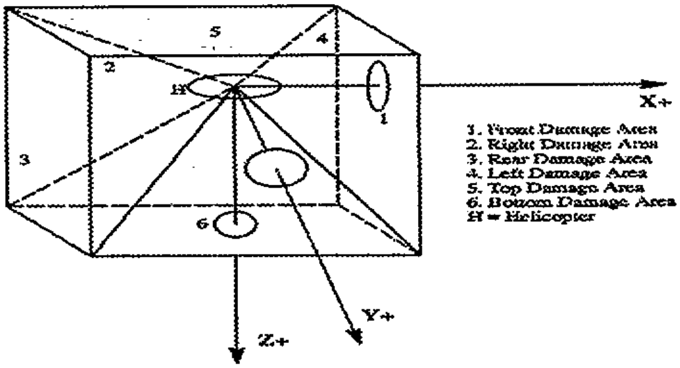

Figure 4 Damage Area Scheme

Each WST/ATD ECSS also calculates the weapon flyout trajectories and damage assessment for a weapon that has been fired/launched at it. The flyout trajectories are six degree of freedom (DOF) trajectories and are used by the CIG to present the flyout visually. There are five trajectories simulated. The pursuit trajectory determines the target line-of-sight from the weapon and directs the weapon velocity vector toward the target. The three-point trajectory determines the target's line-of-sight from the weapon, computes the weapon's time-to-impact from the line-of-sight, calculates the target velocity and the weapon speed, and directs the weapon velocity vector toward the predicted impact point. The proportional navigation trajectory determines the target line-of-sight from the weapon, computes the line-of-sight angle rate of change and vector direction of change from the line-of-sight, tracks weapon velocity history data, limits and filters the line-of-sight angle rate of change, and modifies the weapon's velocity vector direction from the filtered line-of-sight angle rate of change and vector direction of change. The beam rider trajectory determines the target line-of-sight from the weapon's control point and modifies the weapon's velocity vector direction to remain along this line-of-sight. The ballistic trajectory determines the weapon's velocity vector in a straight line path.

The damage assessment model for a WST/ATD initially begins at the termination of a weapon flyout. A Monte Carlo technique is used to determine if the weapon makes a direct impact upon the target. Direct impacts result in direct kills. If the weapon missed but still detonated, a damage assessment model is used to determine the amount of damage the target took. This model breaks the target up into six areas that surround the target. The damage area is determined at weapon termination. The amount of fragmentation/blast damage is calculated for the damage area and scaled. Each damage area has a set of critical equipment that is susceptible to damage and rated as to how much scaled fragmentation/blast it would take to "kill" that component. The amount of scaled fragmentation/blast damage is used to determine which components are "killed" or not "killed" and those that are "killed" are automatically malfunctioned. The aircrew member must then respond to the affects of a damaged aircraft which could ultimately become a "kill". Figure 4 illustrates the damage area concept for the damage assessment model.

EW DEFENSIVE SYSTEMS AND COUNTERMEASURES SIMULATION

The design approach used for the EW defensive systems models decoupled the threat models from the operation of the defensive models. Early EW simulations routinely structured their threat models to drive specific reactions in the defensive systems models. Parameters selected for threat simulations were specifically tailored to drive a required radar warning receiver display and audio response, or dictate a specified countermeasures effectiveness. As part of the threat structure, for a given threat operating mode and countermeasures systems technique, selection of a single exact effectiveness factor would be applied to the threat for the entire engagement. It must be emphasized that these effectivity factors were placed in the threat structure and not part of the defensive systems model. The defensive systems models, in large part, only processed switch changes and lamp displays.

To accommodate this decoupling, the design for radar warning receiver (RWR) models must begin with an in- depth analysis of the operational flight program (OFP). For each module of the OFP, a trade analysis is required to determine whether: (1) the actual OFP code can be used line by line; (2) an emulation of the module be created; (3) the module be functionally simulated; or (4) the particular module may not be required for simulation/training. Of course, the simulation host language and ease of translation to the host language plays a critical role in use of the OFP line for line. Programmability of modern RWRs is a critical factor in the rehost/emulation analysis. The approach used in the IECSS was driven by the design decision to incorporate user update capability to the emitter identification data (EID). When a change is made to the aircraft's EID data, the same change can be loaded in the simulation's EID since the same data structure has been maintained. The modules of the OFP that directly access the EID are therefore rehosted to the simulation's host language. This design approach creates a simulation near emulation of the aircraft's RWR. As a result, the RWR simulation operates independently of the threat environment. The RWR simulation samples the threat environment, reads all threat parametric data, and from this point on processes the data, displays the threat symbology and creates the associated threat audio without further direct interaction with the threat environment until a new processing cycle is initiated.

For active countermeasures systems models, all switchology and display simulation, threat processing, and countermeasures assignment are accomplished internal to the model. Data structures independent and decoupled from the threat structure are used to store and maintain the effectivity factors used by the countermeasures effectiveness module of the IECSS simulation. For electronic and infrared countermeasures systems, the technique type and parametrics associated with the technique are determined by the defensive system model and transferred to the countermeasures effectiveness routines. For expendable countermeasures, the timing between releases and the quantity for each release is controlled by the defensive system model. The quantity for each dispense event is transferred to the countermeasures effectiveness routines as they occur.

The key link to our decoupled threat and system design is the countermeasures effectiveness (CME) subroutine. The CME receives inputs from the IECSS EW defensive systems and the threat related data from the software partitioned common environment data (CED). The CME scans all operating defensive systems for valid countermeasures entries in the effectiveness data tables. A list of possible countermeasures is stored for further processing. For example, if a chaff cloud is active, the active threats are tested for matching operating mode and stored in a countermeasures possible list. Once all the active threats and defensive systems countermeasures are determined, each possible countermeasure is dispatched to the appropriate type of countermeasure effectiveness subroutine (RF, IR, chaff, or flares). These subroutines compare the ownship's radar cross section and IR signature to the countermeasure's chaff and flare signatures respectively as well as the jammer-to-signal ratio and IR intensity for RF and IR jammers. The tabulated effectiveness factors are then adjusted for probability of success based upon real world flight and simulation test data. The ownship's RCS and IR signature is updated every effectiveness call based upon current threat/ownship engagement geometry. The CME module is executed at a one hertz rate, and calculated threat aim point adjusted every cycle based upon the current geometry and environmental conditions.

DATABASE SUPPORT

The purpose of the IECSS Database Support System is to provide an off-line editing capability of the information needed to support a real-time weapon system simulation. The data is structured using relational database techniques and is available for update via menu driven interactive screen input and standard SQL commands. The IECSS Database Support System uses the ORACLE Relational Data Base Management System and ORACLE products such as SQL*FORMS and SQL*MENU to accomplish this task. The database supp.ort system is divided into two logical categcries; Library Maintenance and Mission Preparation.

Library Maintenance tasks involve the editing of data which is used to support the following simulation capabilities:

1) Threat Parametrics
2) Site and Platform Maintenance
3) Aircraft EW Configuration
4) Electronic Order of Battle/C3 (EOB/C3) Parameters
5) EW System Characteristics and Emitter Identification Data

Threat Parametrics are used by the real-time threat models to perform an accurate simulation of threats as they interact with the electronic warfare environment. The Site and Platform library is essentially an association of one or more threats with a Site or Platform name. The Sites and Platforms built can be positioned in a mission scenario causing all associated threats to be incorporated. Aircraft EW Configuration parameters define specific system setup data for a particular aircraft. This includes the positioning of countermeasure dispense hardware onboard the aircraft. The EOB/C3 library defines engagement modifiers and communication delay times based on a particular conflict level. The EW System characteristics and Emitter Identification library contains the data used by the various EW system models to perform an accurate simulation of the system against threats in the environment. The data maintained in each of the Library Maintenance libraries is read during IECSS initialization at the beginning of a mission training or rehearsal session.

Mission Preparation involves the building of mission scenarios, or laydowns, based on Library Maintenance threat data, and the generation of these scenarios and supporting data sets into a mission load. A mission load may contain up to 25 different scenarios and includes the threat parametrics, weapon parametrics, EW system characteristics, Emitter Identification data and threat positional data required to support the simulation. The mission load files are read during IECSS initialization.

The mission scenario is the EW environment defined by the aircrew instructor to accomplish a particular training or rehearsal objective. The scenario is built by positioning threats using the latitude/longitude coordinate system. Each threat is assigned other field information such as altitude, speed, heading visual identification, communication delay times and engagement modifiers. Threats within a single mission scenario may be networked together and assigned corresponding EOB/C3 parameters.

SUMMARY/CONCLUSIONS

The IECSS supports a full EC aircrew training capability providing for training in the usage of the installed EW avionics and the proper techniques required for countering the simulated threat sites. By supporting the ability of simulating multiple aircraft in a single gaming scenario, the IECSS provides a mission rehearsal capability that is unparalleled in the simulator world.

Another major achievement in the IECSS software implementation is the modularity of the design allowing for future software upgrades. This design approach allows for new threat simulations to be added to the environment via the off-line editors while the real-time design allows for insertion or deletion of EW avionics models based on changes to aircraft configuration. Also, by keeping in mind the desire to add new SOF aircraft to the ISN, the design also supports adding new aircraft configurations to the software development environment. Finally, by having a single data area (CED) for all EC environment information, the IECSS design can be easily integrated with other dissimilar simulations/simulators by using the Distributed Interactive Simulation (DIS) standards and protocols.