Real-time IR/RF Missile Simulation Leveraging FPGA Technology

Dr. Stephen C. Jones, Robert L. Hall, and Dr. Mark Brown
Scientific Research Corporation
103 Quality Circle, Suite 220, Huntsville, AL 35806
sjones@scires.com, rhall@scires.com, mdbrown@scires.com
(256) 971-9880, (256) 971-9890 (Fax)

Distribution Statement A: Approved for Public Release

Abstract

This paper presents an associated technology developed to verify/demonstrate the closed-loop functionality of the Multi-Spectral Stimulator (MSS) Injection Test Method (MSSITM). The Digital Missile Simulation (DMS) started as a concatenation of non real-time models that represent a missile fly-out: A Seeker Response Model (SRM), an IR Tracker, and a generic missile 6-DOF / Autopilot model. After the successful closed-loop capability was achieved a parallel effort was initiated to stimulate sensors in real-time. This effort followed two paths in an effort to reduce the risk in locating hardware for stimulation. One path included stimulating a dual-mode seeker in a HWIL environment and the other was to adapt the present DMS to run at real-time rates. The DMS effort and capabilities are emphasized in this paper.

The real-time DMS development ported the computationally expensive model aspects to a Field Programmable Gate Array (FPGA) PCI board housed in a PC and interconnected to the MSS via Reflected Shared Memory (RSM). The FPGA modeled components work interactively with the DMS main simulation located on the host PC and can access updates to functional and operational parameters through the RSM network during scenario runs. Examples of RSM data passed are: range information, supplied by the RF scene generator, providing data for range gate sizing, and the 6-DOF and track error information. This paper provides development, details, and benchmarks produced by this successful effort.

1. Introduction

The DMS initial components, Aviation & Missile Command Research, Development, and Engineering Center (AMRDEC) (previously) MICOM's InfraRed Seeker Analsis Toolbox (MIRSAT), the SRM, an IR imaging tracker, and a generic 6-DOF / Autopilot models were concatenated to form a PC-based non-real-time missile simulation developed to verify closed-loop capability of the phase I proof-of-concept MSS developed. Under phase II the emphasis was to stimulate actual sensors in real-time. However, dual-mode seekers were not readily available for integration and testing and as a risk reduction effort an alternative demonstration was proposed. This demonstration was to distribute the processing over several PCs connected via RSM. This was accomplished by porting the computationally expensive portions of the non-real-time missile simulation to a FPGA Peripheral Component Interconnect (PCI) board and by having the host PC function interactively with the FPGA. The host PC houses a SCRAMNet GT RSM board and Gidel's PROCStar II PCI board. This PC is interconnected to the MSS via RSM. The MSS Scenario Controller controls scene generation timing and simulation initialization and processes the 6-DOF/Autopilot models.

2. MSS Background

The MSS prototype system produces synchronous IR and RF outputs for either injection or projection to stimulate multi-mode sensor systems or simulations. The scenes generated are temporally and spatially registered and generated from a three-dimensional database. A common Scenario Controller (SC), utilizing a real-time operating system, coordinates terrain, target and sensor data to Scientific Research Corporation's Adaptable Radar Environment Simulator (ARES) and the AMRDEC's Multi-Mode Scene Generator (MMSG). The MSS prototype contains a native Test and Training ENabling Architecture (TENA) interface which acts as the conduit to outside systems thus enabling the stimulator to interact with live, virtual, or constructive entities (i.e. battlefield scenarios, live targets / threats, simulations). The stimulator can run in either real-time or stepped mode, providing signals on demand. The stimulator has also been integrated to a high fidelity missile simulation that consists of an IR seeker response model, IR imaging tracker, and a 6-DOF/Autopilot missile model. The stimulator design can be modified to stimulate multiple passive sensors, active laser systems, multi-mode systems, multiple radar systems, or almost any combination of sensors. Figure 1 provides an overview of the MSS, the component block diagram of the closed-loop system, and the resulting prototype and portable rack assembly.

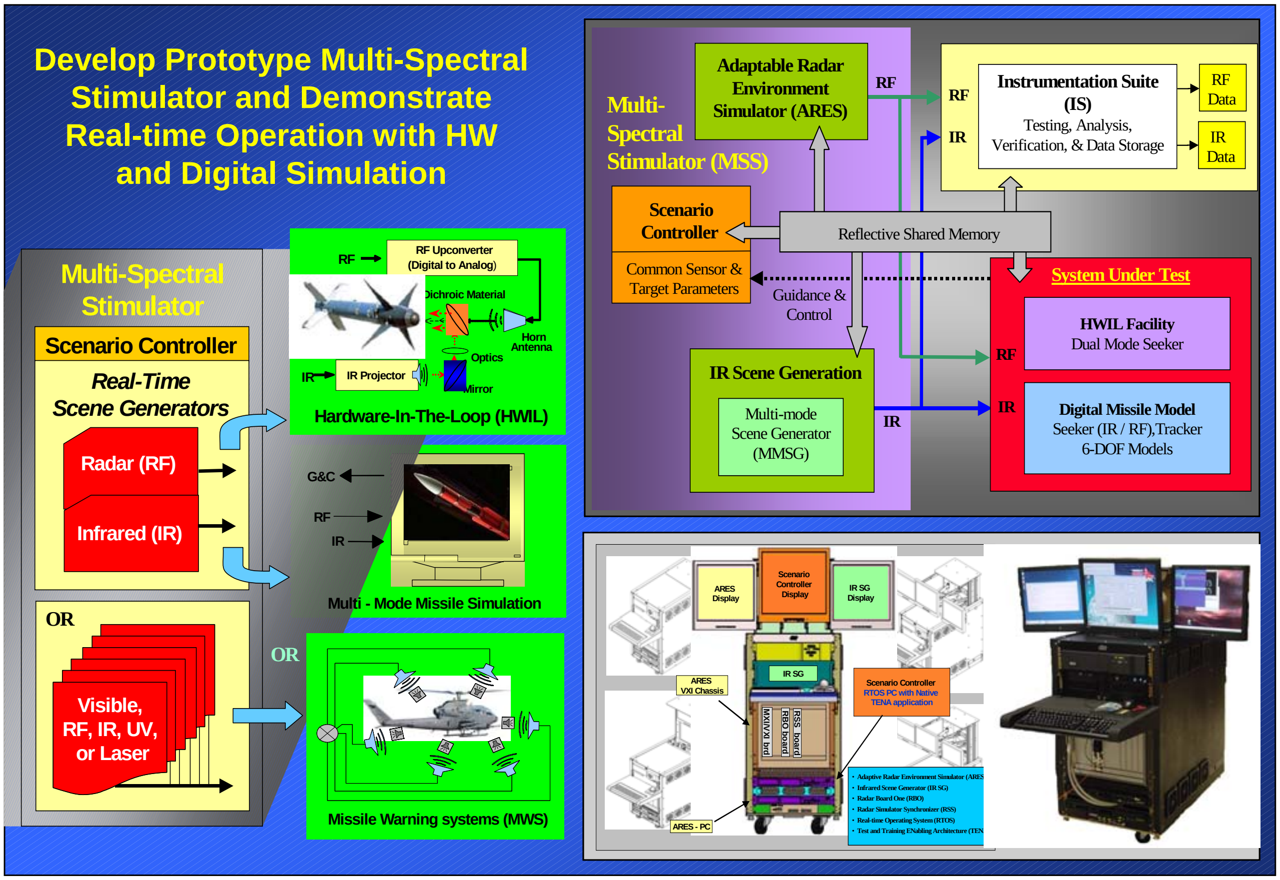


Figure 1. System Overview, Functional Diagram, and Prototype

3. DMS Demonstration Configuration

The DMS closed-loop demonstration configuration is shown in Figure 2. The MSS contains the SC which hosts the 6-DOF/Autopilot models and controls the overall simulation timing, control, and parameter transfer for the scene generators (SG), data recording on the Instrumentation Suite, and setup and initialization of the DMS. The RF and IR data are recorded in real-time to the Instrumentation Suite (IS), in separate files along with their corresponding frame by frame sensor and scenario data. The IS not only captures the RF data to disk but transfers the RF I/Q components to streaming data across RSM. This allows the real-time display of the RF data on any system connected to RSM which in turn can provide a target range calculation usable by the DMS for range gate sizing or alternately for an RF seeker/tracker or sensor fusion application. The MSS IR SG graphics engine has a DVI output and provides 16-bit gray-level imagery whose values are located in the Red and Green channels of the RGB signal. The DVI output is sent through a DVI splitter and then through two DVI to Cameralink converters feeding, in parallel, the IS and DMS systems. The DMS processes the imagery through the SRM and the IR tracker and outputs the track error through RSM to the 6-DOF/Autopilot models on the SC, which then updates sensor viewing angles for the SGs with respect to the given 6-DOF/Autopilot results.

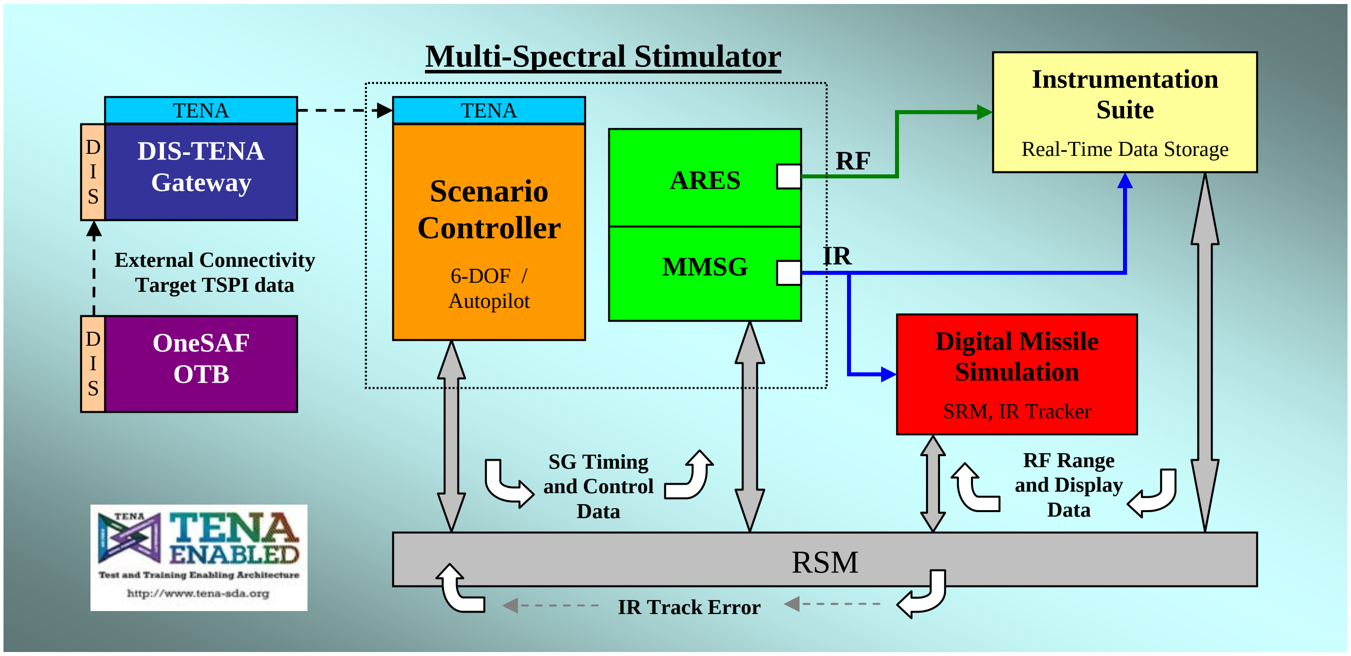


Figure 2 MSSITM Digital Missile Simulation configuration

The DMS configuration also shows the Live, Virtual, and Constructive (LVC) capability provided by the MSS through its native TENA application. External connectivity is provided to the SC through TENA, a DIS-TENA gateway and from One Semi-Automated Forces (OneSAF) that provides target TSPI data. Figure 3 shows the TENA configuration used to obtain target state data used to verify an external connectivity capability.

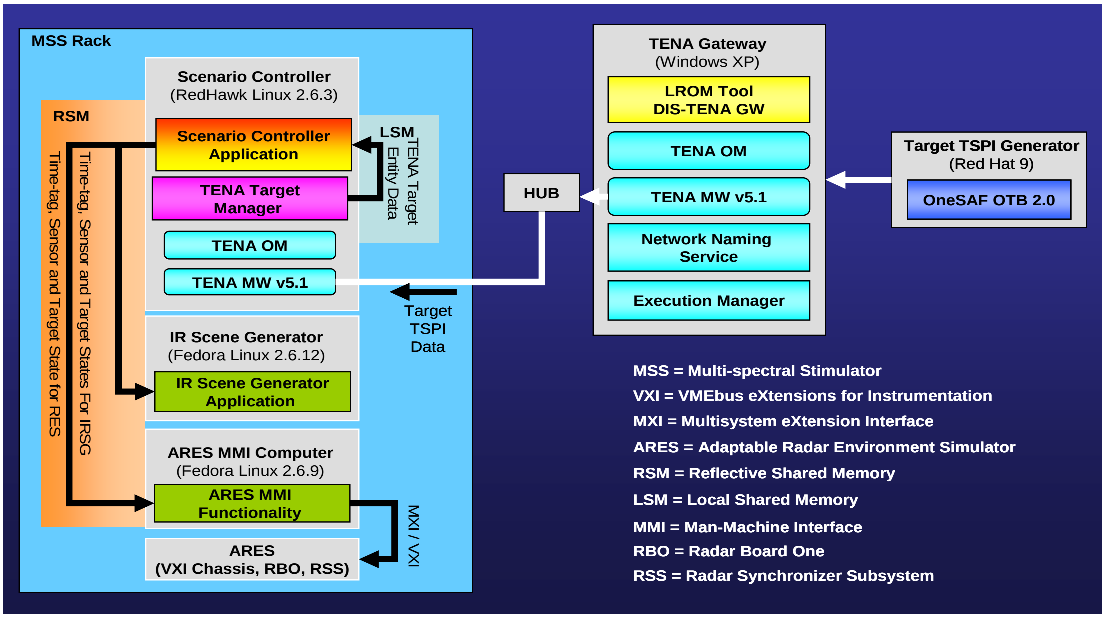
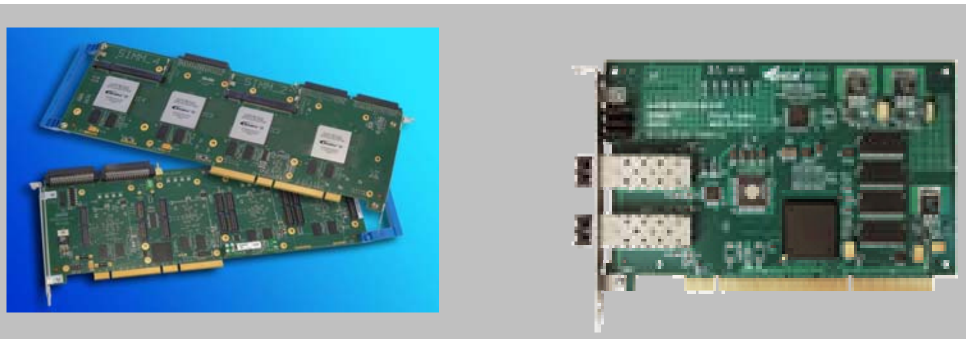


Figure 3. MSS TENA interface

3.1 Hardware, Simulation, and Software Models

The DMS hardware consists of the following: The DMS PC contains dual Opteron processors with 2GBs of RAM and the PCI bus is populated with the RSM and FPGA boards shown in Figure 4. The Gidel PROCStar II contains four Altera Stratix II 60 FPGAs and is capable of processing ~240 logical elements (LE). The DMS simulation utilizes approximately 70% of the boards processing capability. The FPGA board presently holds a Cameralink daughter card to acquire the IR imagery from the MSS IR SG. The second PCI board is the RSM, SCRAMNet GT, configured with 128MB of shared memory and is capable of 210 MB/Sec of data throughput. The simulation control parameters, track error, and RF data are passed to the DMS through RSM.



Gidel ProcStar II FPGA board

SCRAMNet GT RSM board

Figure 4. DMS PCI hardware

3.2 Real-time Generic Digital Missile Model

The missile simulation model functions primarily on the IR data being generated by the MMSG but can leverage the range data provided by the RF component or the true range provided by the SC. The missile simulation consists of three models: an IR SRM, an IR imaging tracker, and a 6-DOF / Autopilot model. These combined models represent the essential components within a missile simulation. The current effort developed a digital missile model, running at real-time rates, by porting the computationally expensive algorithms within the separate IR SRM and IR tracker to a FPGA PCI board. The FPGA board feeds the resulting track errors to the host PC system which passes the results through RSM to the 6-DOF / Autopilot model hosted on the Scenario Controller.

The original FPGA-based DMS processed a 640x480 IR image but only modeled the seeker through the convolution stage of the SRM using 24-bit fixed-point numbers. The IR tracker ran on the host PC and was restricted to a 256x256 ROI. This provided approximately 8 frames per second (fps) in processing speed for a closed-loop simulation. The FPGA Seeker model now receives 1024 x 768 IR imagery and processes the SRM and tracks on a 512x512 Region of Interest (ROI). This latest version modeled the entire SRM with 32-bit fixed-point convolution and included the port of the IR tracker subsystem. The simulation now has a handshaking approach with the SRM and tracker, where for the tracker; the host PC passes a “track” ROI to the FPGA and FPGA passes Kalman filter output, (track errors), to the host PC. These modifications increased the processing rate to a 60-89 fps capability. The throughput is dependent on the host PC display rate. Table 1 shows the incremental steps and progress of the software port.

The SRM components include the following models: Radiance Look-Up-Table (LUT), Optical Excitance, Convolution, Focal Plane Array (FPA), Electronics and the Imaging IR tracker, a “Most Predominant Feature” or “Hot Spot” tracker.

[Note: The throughput capability can be expanded to a 1024x1024 ROI by obtaining a PCI board with larger footprint FPGA or through re-design and programming distribution between FPGAs. Instead of using the PROCStar II with 60 LE per FPGA, the versions containing either 130 or 180 LE would allow convolution to be performed on the larger ROI without compromising frame rate throughput.]

Table 1: Digital Missile Simulation port to FPGA Progress

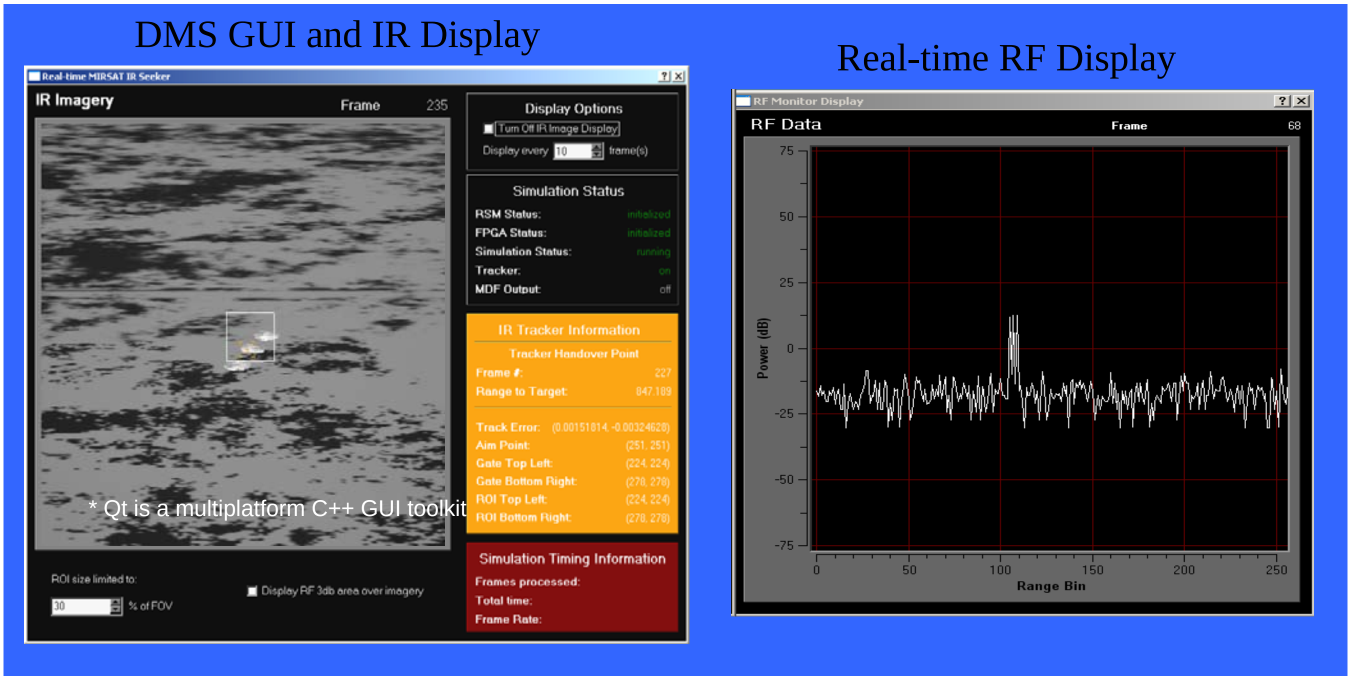
DMS description	Region of Interest	Frames per Second	Processing Environment	Displayed
IR SRM and IR tracker running on Host PC only	256 x 256	~3 fps	Linux OS, using SFE	All frames
IR SRM and IR tracker on Host PC - **Optics ported to FPGA**	256 x 256	~8 fps	Linux OS, using SFE	All frames
Entire IR SRM ported to FPGA, IR tracker running on Host PC	256 x 256	~27 fps	Linux OS, using SFE	All frames
Entire IR SRM ported to FPGA, IR tracker running on Host PC	**512 x 512**	**50 → 60 fps**	Linux OS, **No SFE, using Open GL**	Every 10th frame → No Display
IR SRM and IR tracker processed on FPGA controlled from Host PC	512 x 512	**60 → 89 fps**	Linux OS, **No SFE, using Open GL**	Every 10th frame → No Display

* Simulation Framework Environment (SFE)

Table 1 illustrates the overall performance improvement provided by creating an application that interacts directly with the FPGA board where the computationally expensive processing is streamlined onto dedicated processors. It shows the continual processing increases that started by porting the SRM radiance Lookup Table (LUT) and optics models, written in C++, to the FPGA board processors. The initial increase is from 2-3 fps to 7-8 fps, easily doubling the speed. The processing speed increased again to 27 fps by porting the remainder of the SRM models. The SRM software models were initially wrapped within a Simulation Framework Environment (SFE). Unfortunately this environment contains limitations to image display and hardware control. Greater simulation control and significant processing speedup, 50-60 fps, (depending on display output), was achieved by extracting the SRM models from the SFE. This frame rate also includes an increased ROI processing, from 256 x 256 pixels to 512 x 512 pixels. Next, the development of a Digital IR Missile Simulation Driver, that optimizes FGPA control, allowed the application to run 136% faster. Finally, porting the IR tracker code to the FPGA resulted in a further 12% increase, reaching 60-89 fps; again, the frame rate results are output display dependant.

3.3 Real-time IR GUI and RF Displays

The initial DMS Graphic Users Interface (GUI) resided within a SFE called OmeChron which served as a flexible wrapper for the MIRSAT algorithms and offered interactive setup and display capability. However, a new driver application and GUI was required to provide a platform independent application for testing and to address the need for greater hardware control and a more a flexible host-FPGA interface. This was accomplished by porting the remainder of the SRM models to the FPGA board and while using OmeChron/MIRSAT as the driver that interacted with the FPGA and the remainder of the system (MSS, IR Tracker, and 6-DOF/Autopilot). A simplified driver application based on Qt, a cross-platform GUI software development kit**,** was developed to provide greater hardware control and displays meaningful user information, shown in figure 5. This new driver replaced the SFE version and also serves as a tracker debugging tool, displaying frame by frame status. The Qt application also allowed the use of OpenGL providing greater graphics display flexibility and removing the flicker prevalent in the SFE driver.



Frame 235

Display every 10 frame(s)

RSM Status: initialized
FPGA Status: initialized
Simulation Status: running
Tracker: on
MDF Output: off

Frame #: 227
Range to Target: 847.189

Track Error: (0.00151814, -0.00324628)
Aim Point: (251, 251)
Gate Top Left: (224, 224)
Gate Bottom Right: (278, 278)
ROI Top Left: (224, 224)
ROI Bottom Right: (278, 278)

Frames processed:
Total time:
Frame Rate:

ROI size limited to: 30 % of FOV

☐ Display RF 3db area over imagery

Frame 68

Figure 5. DMS GUI, IR Display, and Real-time RF Display

The Real-time RF Display application, shown in Figure 5, was developed in support of the MSS. It computes and displays the magnitude of the baseband I&Q output from the Adaptive Radar Environment Simulator (ARES) in real-time and can be run on any PC with Windows that has Qt, Qwt, and a SCRAMNetGT Reflective Shared Memory (RSM) card. It is presently run under Windows but will run on Linux once the source code is rebuilt for the Linux environment. The RSM data that this application receives in real-time originates from the serial I&Q data output from the ARES, which is then collected by the FibreXtreme SL100 card hosted on the Instrumentation Suite (IS). Video Savant (VS) is run remotely by the VS Remote Process Control (VSRPC) application on the IS, which is activated by the Scenario Controller via Ethernet, in order to synchronize this operation with the MSS simulation. The VS Stream Filter, which is a customized dll that operates in conjunction with VS, copies the first 400 pairs of the in-coming I&Q serial stream from the FibreXtreme to RSM established by the SCRAMNetGT card on the IS. The Real-time RF Display acts as a monitor, computing and displaying the magnitude of the I&Q that is currently in RSM. The basic RSM fiber ring configuration for the DMS simulation is shown below in Figure 6. The portion of the configuration that is relevant to RF output is highlighted in color and in bold font. Similarly, a standalone driver was developed that populates RSM with RF data from a stored data file.

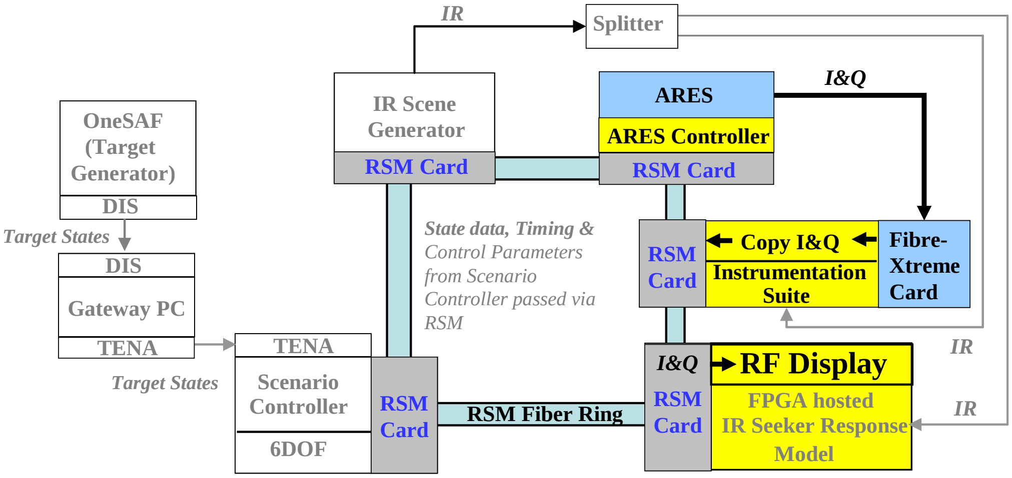


Figure 6. DMS configuration w/ Real-time RF Display emphasized

3.4 IR Seeker Response Model (SRM)

A generic high-fidelity IR SRM was modified and ported to work under the Windows and Linux OS and utilizes Qt as the GUI interface. The original SRM resides within a Simulation Framework Environment (SFE) called OmeChron, which consists of: an Object-oriented simulation development environment, an application that supports the visual assembly of component models into composite models of arbitrary complexity, provides the capability to set / store object properties and relationships, and supports event-based or process-based simulation. The component models represented within the SRM are: dome, optics, focal plane array (FPA), multiplexer, amplifier, analog to digital (A/D) converter, scan converter, and nonuniformity correction. Figure 7 shows the components represented within the SRM.

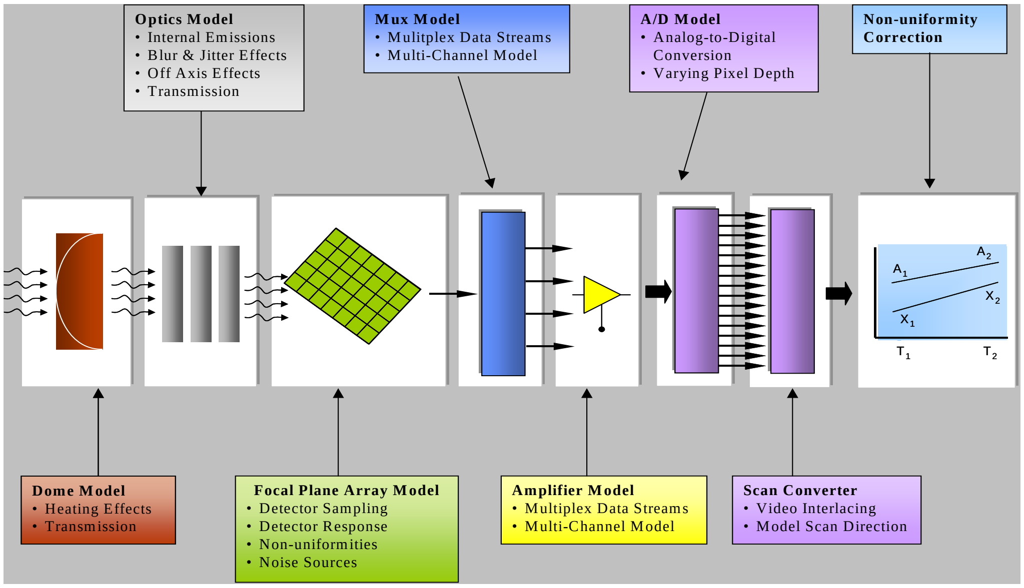


Figure 7. SRM – IR Seeker Component Models

The model algorithms representing the SRM ported to the FPGA board are shown in Figure 8. The controls and setup parameters are housed on the host PC and enable the FPGA to process exclusively on the input imagery provided by the MMSG.

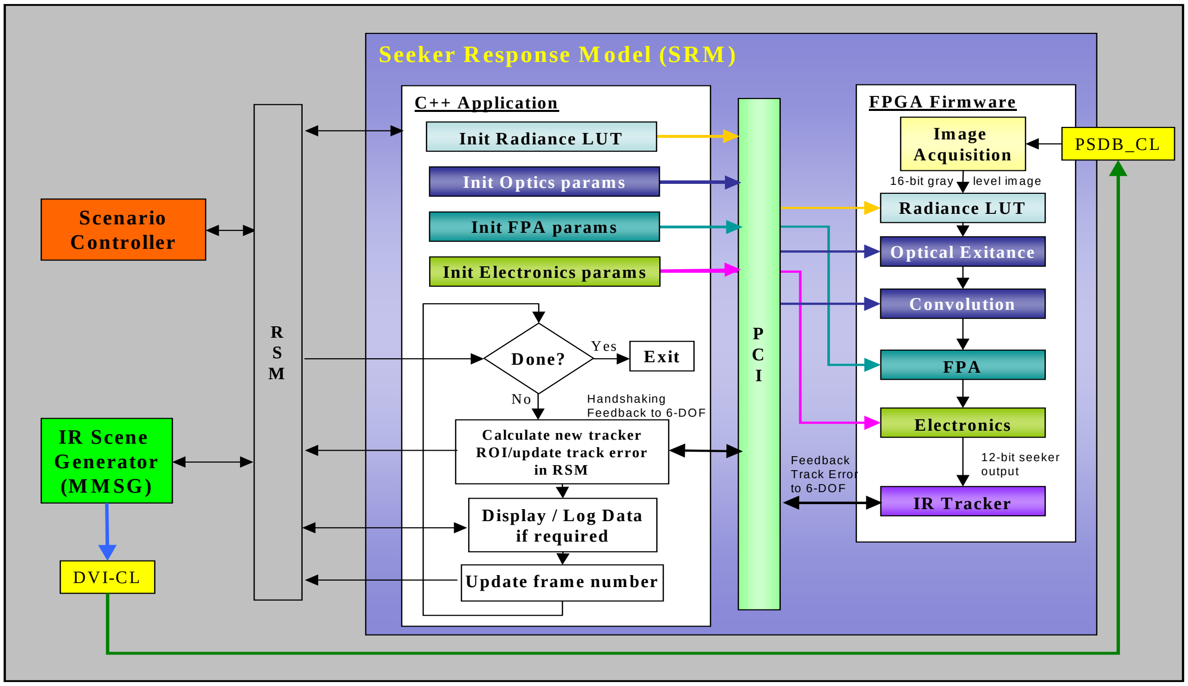


Figure 8. SRM – IR Seeker Component Models ported to FPGA

3.5 Imaging IR Tracker

The IR tracker ROI is dependent on either true or estimated range; this ROI is determined on the host PC and passed to the FPGA. A Most Predominant Feature (MPF) tracker processes the imagery by determining the spatial location of the most predominant peak, which is then smoothed by a 2D Kalman Filter to determine the final track errors. These errors are passed back to the SC through RSM to the 6-DOF/Autopilot model closing the loop for the digital missile simulation.

The IR Tracker is a rather simple, but expandable, image processing model that searches an initially small ROI for a MPF. The tracker has three modes of operation: Off-line open-loop incorporated into the MSS Multi-Spectral Player analysis tool, closed-loop mode that is fed raw 16-bit gray-level data captured from IR Scene Generator during run-time, and in closed-loop mode fed 12-bit A/D gray-level output from the SRM.

3.6 6-DOF / Autopilot (Open-Loop & Closed-loop Modes)

The SC has been developed for use in both open and closed-loop modes and can utilize a 6-DOF model either from the SC or from an external source (e.g. SUT). The open-loop SC mode can drive the ARES and MMSG (at the same or different rates), with target state and sensor position information. Under the open-loop scenario the user specifies: missile trajectory, ground range(s), launch azimuth, delta altitude(s), nominal seeker LOS, and scan rate / extent.

During Closed-loop runs, the MSS Scenario Controller (SC) requires that a 6-DOF model process is running, either internally or externally. Target states are generated internally or from a standalone 6-DOF model. Target state information can also be received from the TENA Target Manager, a SC native application. In all cases, the SC is responsible for translating the external target information. Under the closed-loop scenario, the user provides the target location and the launch location and azimuth.

4. Summary

The DMS was initiated as a tool to verify the MSS proof-of-concept closed-loop simulation capability and grew into a parallel effort to demonstrate MSS real-time stimulation capability. The use of FPGA technology in coordination with high-end PCs brings a High Performance Computing (HPC) capability a PC simulation functioning at real-time rates. This opens up the potential to create test beds with the versatility for testing both sensors and scene generators. This technology can be readily expanded for use with alternate sensor inputs and used in parallel to develop sensor fusion test beds that can create and test novel discrimination techniques for trackers, ATR, or a variety multi-sensor systems that could populate UAV or UGV systems. By installing parallel FPGA PCI boards in a single system, multiple sensors, seekers and their trackers could be coordinated and tested simultaneously. Providing a facility to test disparate sensor components and develop novel discrimination techniques at real-time speeds. These test beds could be used in a distributive fashion allowing the use across differing ranges modeling actual threats or weapons without compromising real assets or endangering the warfighter during testing.

5. Acknowledgements

The authors would like to thank the Test Resource Management Center (TRMC) Test and Evaluation/Science and Technology (T&E/S&T) Program for their support. This effort is funded by the Multi-Spectral Test Focus Area under the TRMC's T&E/S&T Program through the Scientific Research Corporation's contract number DAAD05-02-C-0048. Additionally, the AMRDEC's Systems Simulation and Development Directorate, Hardware In The Loop Functional Area provided access to its Multi-Mode Scene Generator (MMSG) and support for the RF4 HWIL integration.