



Design Enhancements of GUI Based Host Hardware Acceptance Test Software

Shridhar.B.Devamane¹, Amulya.K², Deepika. K. N³, Pooja.R⁴, Vinay Kumar. N⁵
Assistant Professor¹, BE Student^{2,3,4,5}
Department of Electronics & Communication
APS College of Engineering, Bangalore, Karnataka, India

Abstract:

Hardware acceptance test software is a software application package for testing the system hardware and the EW (electronic warfare) processing algorithms of the system in test mode. Hardware Acceptance Test software consists of two components: Target software and Host software. Target software is the main software which will be executing in the system hardware, Host software is PC based GUI application software, which will interact with the Target software for tests or operations. Scope of this project is to enhance the capabilities of Host component of hardware acceptance test software for incorporating the following features such as operations for verifying the pulse descriptor word generations of wide band and narrow band receiver channels for the integration rig setup, batch file execution of Pulse descriptor word reception operations, different types of file conversion operations.

Keywords: Radar warning receiver, Pulse descriptor word, Direction finding, Radar, Electronic warfare.

I. INTRODUCTION

Radar Warning Receiver (RWRs) has been an important part of electromagnetic Warfare (EW). The RWR systems detect the radio emissions of radar systems. Their primary purpose is to issue a warning, when a radar signal that might be a threat is detected. The warning can then be used manually or automatically, to evade the detected threat. RWR systems can be installed in all kind of airborne, sea-based assets. In the case of a fighter aircraft the RWR is designed to detect incoming radar signals, identify and characterize these signals to a specific threat, and alert the aircrew through the tactical electronic warfare system (TEWS) display, and the pilot can take necessary countermeasures. Electronic Countermeasures (ECM) are the actions taken to prevent or reduce the enemy's effective use of electromagnetic spectrum. Two major actions of ECM are jamming and deception. From 1990s last century, Radar electronic War has played a more and more important role in the local wars. The radar reconnaissance equipment, named electronic support measure (ESM) is one of the most important parts in the Electronic War, performing threat detection and area surveillance to determine the bearing and identity of the surrounding radar emitters. In the dense electromagnetic environments encountered in warfare, the large number of surrounding emitters will cause an ESM to receive a seemingly total random pulse stream consisting of interleaved pulse chains with large noise. Only when we segregate the different radar pulse chains from the pulse stream, can we proceed with the measurements, analysis, identification of the signal parameters and types, and the impose all kinds of interference on the radar threatening emitters, including blanket jamming and deceptive jamming, etc. A typical system is composed of a passive antenna and a receiver which down converts the high frequency RF signal received by the antenna to a lower intermediate frequency (IF) which can be processed more easily. The IF signal is then brought into the digital domain using an A/D converter. Operating on the digital samples and the pulse measurements block produces PDWs which are collection of measurements

that describe the characteristics of the signal (TOA, PW, Frequency, and PA). The PDWs are fed to the processing block whose block job is to sort out the received signals and identify the types of radars in its signal environment and their parameters. Here the number of pulse trains which arrive and leave with different pulse characteristics to the block of de interleaving process in major concern. The HOST-HAT software is executable software in a host PC. The HOST-HAT software communicates with the TARGET software through Ethernet. The HOST-HAT software includes the test procedure to verify and validate the interface, logical and functional testing of the hardware. The HOST-HAT software is required to send the commands to the target and target performs the related test/ operations and replies the results to the HOST software the results are to be displayed in the HOST GUI.

II. PROPOSED SYSTEM

A RWR converts real radio frequency pulse signals into digitized radio frequency pulse descriptor word (PDW) data and a pulse processor is used for filtering PDW data, The continuous streams of supplied radio frequency PDW data is inserted into the electronic surveillance system for the purpose of simulating a radio frequency pulse environment. An electronics unit is configured for integration into the electronic surveillance system between the receiver and pulse processor for inserting supplied radio frequency PDW data files into the pulse processor, computer including hardware for storing the PDW data files and to system software for providing a graphical user interface for the received PDW data files. The hats software comprises at least two databases for radio frequency PDW data files, first PDW file database for storing real radio frequency PDW data previously received by the electronic surveillance system, and a second PDW file database for storing files created by reformatting data files, whereby the fabricated files maintained in said second pulse descriptor word file database may be selected for transmission to said electronics unit and insertion into the electronic surveillance system. A database for pulse descriptor word data

files, containing the data about angle of arrival, amplitude, frequency, and time of arrival pulse descriptor word data and storing the files on the database and wherein said sub-step of creating pulse descriptor word data files comprises downloading pulse descriptor word data files containing actual amplitude, threshold and rfdca pulse descriptor word data for radio frequency signals previously received by the electronic surveillance system and recording and reformatting the downloaded file, fabricating pulse descriptor word data, and combining the fabricated pulse descriptor word data with previously stored pulse descriptor word data files. Direction finding (DF) is used for finding the angle of arrival of the received signal over the whole (360°) direction. This facility makes these systems important in military and civilian applications. To simplify the amplitude comparison analysis, the ratio between signal levels which are fed from receiving antennas of the proposed DF is used. This method makes the system simple in analysis and implementation and gives a good accuracy and resolution over a wide range of frequencies. The proposed system operates over a wide band of frequencies and covers the whole (360°). Purpose of direction finding operations is to detect the position of the emitter, which can be calculated from the bearings of several direction finders. The characteristics of the radar signal that are generally estimated include the signal's

- Amplitude
- Frequency
- Duration
- Time-of-arrival (TOA)
- Angle-of-arrival (AOA)
- Intra-pulse modulation characteristics,
- Inter-pulse characteristics.

Once the characteristics of all radar signals have been estimated, radar receivers use the estimated parameters to sort the collection of intercepted signals to identify all of the intercepts that belong to a single radar emitter. These parameters will be stored in the database in encrypted format. Attenuation is a general term that refers to any reduction in the strength of a signal. Attenuation occurs with any type of signal, whether digital or analog. By finding the attenuation losses at the required frequency range, the loss values will be stored and required to compensate the loss the cable attenuation files should be stored in the system database. Calibration files are the files which hold some system input values to perform some operations, basically calibration file will be encrypted binary file stored in the database it must be decrypted for using it decrypted.

2.1 Software design description:

1. Narrow band BIT Integration test.
 - Input DST Filter.
 - Record PDWS in excel sheet.
2. Wide band BIT Integration test.
 - Record PDWS in Excel sheets.
3. Binary file generation for DF table.
4. Excel file to binary file conversion for cable attenuation files.
5. Binary file to excel sheet conversion for CAL tables.
6. Binary file generation for PDW/EMITTER record option.
7. Batch files execution for PDW reception operations and plot PDW Parameters and Record the PDWS in Excel sheets.

8. DST file operation for amp file, threshold file and rfdca file.

III. IMPLIMENTATION

Communication steps between host and target:

- Load DMP HATS HOST SOFTWARE.
- Initialize communication between host and target.
- Select the test/operation to perform.
- Frame the data packets with test/operation parameter
- Frame the command packets with command ID, size of transmission.
- Transmission is done through Ethernet cable.
- Wait for synchronization packet from the target.
- Verification of synchronization packet and check for errors.
- Wait for acknowledgement.
- Display result if acknowledgement is received

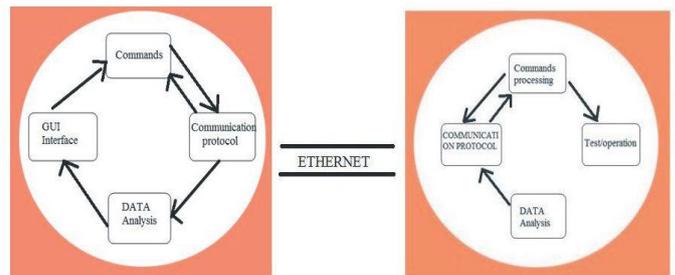


Figure.1. Block diagram of hats software design.

3.2 Bit Integration Test

The purpose of this test is to perform the BIT test for narrow band in integration testing. This test is conducted by receiving and analyzing the PDWs received for all 6 channels and for all 5 bands for both option of 100 MHz and 500 MHz bandwidth of DST.

3.2.1 Implementation algorithm to input DST files

1. Begin.
2. Initialize the new file.
3. Get the directory path of the DST files(thrfile.bin, ampfile.bin and dstfile.bin).
4. Open the file in read mode.
5. Check the file status, if is NULL.
6. Copy all the values from the DST file to the array.

3.2.2 Implementation algorithm to record PDWs in excel sheet

1. Begin.
2. Test whether the test is fail or not, if the test is not fail.
3. Copy the resultant PDWs to an array.
4. Initialize the new file.
5. Get the directory path of the fail.
6. Copy the resultant PDWs values in the array to the file.
7. The copied values will be displayed in the excel sheet.

3.2.3 Batch files execution of PDW reception operations

The purpose of this operation is execute the entire PDW reception operations through commands using batch file The PDW operations are

- system operation.
- configure test mode.

- configure CW/pulse.
- configure threshold.
- configure delay.
- configure mux.
- configure NB receiver.

3.3 Software design description of direction finding table

1. Begin.
2. Select input text file.
3. Load the file.
4. Count number of lines.
5. Read the file line by line.
6. Open output binary file and name it same as input.
7. Encrypt the data and write it to the output binary file.
8. End.

3.4 Software design description of cable attenuation file

1. Begin.
2. Select input excel file.
3. Open the excel app.
4. Load the workbook and worksheet.
5. Select the data to copy (it is called as ranging).
6. Open the output binary file.
7. Read the file tab by tab.
8. Encrypt the data and write it into the output binary file.
9. End.

3.5 Software design description of calibration file

1. Begin.
2. Select input binary file.
3. Open the output excel file.
4. Load the workbook and worksheet.
5. Count number of lines.
6. Read the file line by line.
7. Select the data to copy.
8. Open the output excel file.
9. Read the file tab by tab.
10. Encrypt the data and write it into the output excel file.
11. End.

IV. RESULT

The operations for verifying the pulse descriptor word generations of wide band and narrow band receiver channels for the integration rig setup is done, the binary file from Host software is compared with the target values.

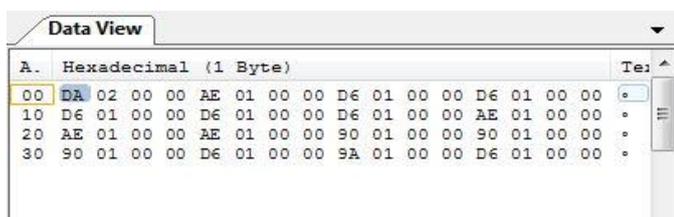


Figure. 2. Amplitude(encrypted binary file).

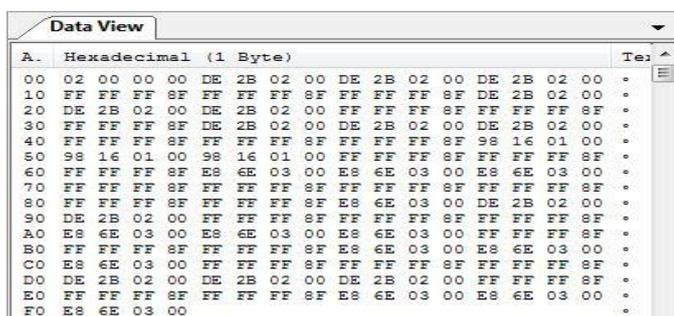


Figure. 3. Threshold(encrypted binary file).

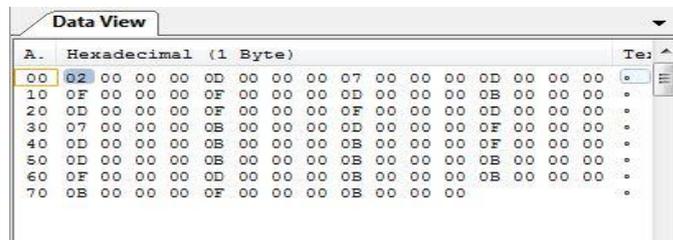


Figure. 4. Rfdca(encrypted binaryfile).

The Direction finding(DF) table data are encrypted and stored in a binary file.

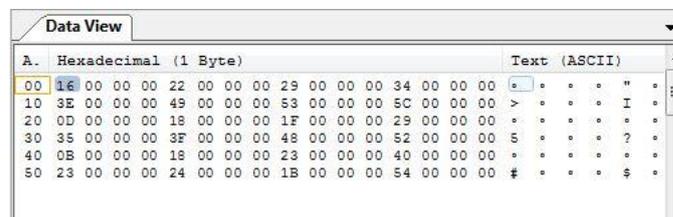


Figure. 5. DF(encrypted binary file).

The cable attenuation file data from excel sheets are encrypted and stored in a binary file.

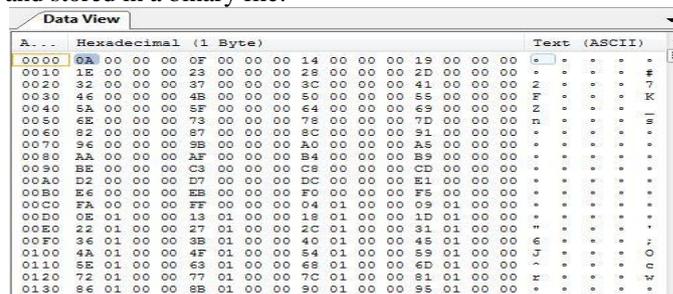


Figure. 6. Cable attenuation(encrypted binary file).

The calibration values are decrypted from a binary file and are stored in excel sheets.

V. CONCLUSION

A scheme of testing the system hardware and EW (Electronic Warfare) by software application called Hardware Acceptance Test software based on GUI (graphical user interface) application software. we have enhanced the capabilities of the host component of the hardware acceptance test software by incorporating the operations like verifying the pulse descriptor word generation of wide band and narrow band receiving channels, that is by taking the DST files from the system file directly for the input files for comparison with the received values instead of declaring the values in the code and record the resultant pulse descriptor word values in the excel sheet after the completion of the test. The batch file execution of PDW reception operations by the single command using batch files instead of the testing the every individually operations manually. File conversion of DF table, Calibration file and Cable attenuation file is also successfully completed.

VI. REFERENCE

- [1]. HAT software design description at Defense Research Development Organization.
- [2]. Mark Fewster & Dorothy Graham, "Software Test Automation", Addison-Wesley, 1999.

[3]. Tim Koomen & Martin Pol, "Test Process Improvement", Addison-Wesley, 1999.

[4]. Shari Lawrence Pfleeger, "Software Engineering", Practice Hall, Inc., 1998.

[5] Robert V. Binder, "Testing Object-Oriented System", Addison-Wesley, 1999.

[6]. Ian Sommerville, "Software Engineering", AddisonWesley, fifth edition, 1995.

[7]. Patrik Rosenquist & Kristina Adelswärd Bruck, Master Thesis: "Automated Testing Of Software With GUI", 1998.

[8]. Edward Kit, "Software Testing (in the real word)", Addison-Wesley, 1995.

[9]. Björn Regnell, Per Beremark, Ola Eklundh, "Requirements Engineering For Packaged Software", 3(2): 121- 129, 1998.

[10]. James A. Whittaker, "What Is Software Testing? And why Is It So Hard?", 70-79, IEEE Software January/ February 2000.

[11]. Gem Kaner, Jack Falk, Hung Quoc Nguyen, "Testing Computer Software", John Wiley & Sons, inc , second Edition, 1999.