

Analysis Tool for Radar Data Using C' Oriented Approach

S.Manasseh¹ G. Rama Krishna² J.Madan Kumar³

Assistant Professor¹

Department of Electronics and Communication Sri Venkateswara College of Engineering and Technology, RVS Nagar, Chittoor, A.P, India

Email ID:prasanth416@gmail.com¹

Professor²

Department of Electronics and Communication Sri Venkateswara College of Engineering and Technology, RVS Nagar, Chittoor, A.P, India

Email ID: svec_grk@rediffmail.com²

Assistant Professor³

Department of Electronics and Communication Sri Venkateswara College of Engineering and Technology, RVS Nagar, Chittoor, A.P, India

Email ID:madansoft@gmail.com³

Abstract—The main objective of our project Radar Data Analysis Tool an 'C' oriented approach scan the tracking data sent to mission computers by Radars like PCMC'S and RADAR'S. The Radar emits signals continuously towards the target. These signals strike the target and echoes (reflects) back towards the radar. The Radar generally equipped with Data System. This system gathers tracking information of the target such as Range, Elevation, and Azimuth individually from various sub systems of radar. This data is transmitted at 10 Hz rate to the mission computers for processing. Check for the validity of scanned tracking data. Arranged data into various data formats such as 22-Bytes, 18-Bytes etc according to various radars. Display the data formats in Hexadecimal form. Plot the graph for prime parameters acquired through data packers.

Index Terms— RADAR'S, Range, Elevation, Azimuth, Target, Echoes, data formats, data packers

I. INTRODUCTION

Two computer systems provide redundant real time chain support for the launch. Data from various tracking and telemetry stations are acquired on these systems and processed. The results are displayed on various types of terminals like Graphics, PCs, Alphanumeric terminals and wide screen display etc., in range safety room. Computers designate mode data for antenna pointing is provided to all tracking and telemetry stations. PC based local area network (SDS-LAN) provides real time support for various specialist displays located in launch control center (LCC), in Mission Control Center (MCC) and Specialist Display System (SDS).

Scan the tracking data sent to Mission Computers by Radars like PCMC'S and RADAR'S. Check for the validity of scanned tracking data. Arranged data into various data formats such as 22-Bytes, 18-Bytes etc according to various Radars. Display the data formats in Hexadecimal form. Plot the graph for prime parameters acquired through data packers.

The data processing is organized on each computer by three separate processes. The first (TM process) is meant to handle chamber pressures, Attitudes, Error functions and INS state vectors with respect to Launcher at 2Hz. The second process (SB process) handles S-Band (Range, Range Rate, Azimuth and Elevation) data and supplies state vector with respect to launcher at 8 Hz/4 Hz. The third process is the tracking process (TR process) which handles 10Hz data of all radar's, TVM (if

available). This also looks for availability of processed data from process 1 and 2, before formatting the output for CDM and various displays [7].

Both processes (SB and TR) check data for validity and reasonableness and later filtering is carried out for each source data to obtain state estimates. Process 1, handles TM data, validates data and through suitable data conversion algorithm brings output with respect to launcher. All processed state vectors are in top centric relative coordinate system [5].

After third stage, a separate process with differential correction (DC) algorithm is activated. The DC algorithm refines and provides an accurate state vector at 380 Sec. The refine state vector is used to generate look angles during the long coast phase, if tracking data is not available during the above period, the INS data is used for CDM data computation from T+400 Sec onwards. But with INS data, CDM transmission is at 2Hz.

II. RADAR (Radio Detecting and Ranging)

This is used mainly to estimate the distance between the station and the target .The Radar emits signals continuously towards the target. These signals strike the target and echoes (reflects) back towards the Radar. By this process the Radar calculates the Slant Range by dividing the total distance covered by the signal with two.

The Radar generally is equipped with Data System. This system gathers tracking information of the target such as Range, Elevation, and Azimuth individually from various sub systems of radar. This data is transmitted at 10 Hz rate to the Mission Computers for processing [4].

1 2 3	4	5 6 7	8 9 10	1112 13	1415 16	17	18
xxxxxx	xx	xxxxxx	xxxxxx	xxxxxx	xxxxxx	xx	xx

TIME: It is count down time in **HHMMSSCS**.

RANGE: Range or Slant Range is the distance between the station (Radar) and the target. Radar measures this range in Kilometers.

ELEVATION: Elevation is the angle in the vertical plane at which the target is observed. It is the angle between the slant range the horizontal plane in the Top centric Coordinate System. This is measured in Centi Mils (1 Mil = 0.0573degrees).

AZIMUTH: Azimuth is the angle between the true north and the horizontal axis which is the projection of the slant range in the Topocentric Coordinate System. It is measured in Centi Mils.

AZIMUTH / ELEVATION ERRORS:

Azimuth and Elevation Errors are correction factors for Azimuth angle and angle of Elevation respectively.

QUALITY INFORMATION:-

SKIN / TRP MODE: The tracking is said to be in Skin mode when the signals from the Radar touches the surface of the target and reflects back.

The tracking is said to be in TRP mode when the signals from the Radar reaches the transponders present in the rocket gets amplifies and transmitted back. Two transponders namely TRP1 and TRP2 are present on the rocket.

OVERALL (AUTO / NONAUTO) MODE:

The data is said to be good or is in Auto mode when the Radar tracks automatically on its own by following the target.

The data is said to be in Non-Auto mode if the Radar is operated manually to track the data of the target.

STATION CODE:

There are various stations with codes such as **PCMC 1, PCMC 2, PCMC3, RADAR 1 RADAR 2, RADAR 3.**

Take the transmitted data from Radar as input to the program and compute the various parameters explained above separately into their respective units as defined in data formats.

Convert them into hexadecimal values and arrange them in order and length (number of bytes that the parameter occupies) as defined in the data formats for different Radars. Some of the data formats are as shown below. Obtain the outputs for the data packets in hexadecimal form.

Finally generate the graphs for parameters Range, Azimuth and Elevation with respect to Time and display them.

Graph 1 : Plot the graph for Range with respect to Time.

Graph 2.: Plot the graph for Azimuth with respect to Time.

Graph 3 : Plot the graph for Elevation with respect to Time.

- 12 – 14 BYTES: ELEVATION – CENTI MILS 3 bytes
- 15 – 17 BYTES: RANGE RATE – METER/SEC – 3 bytes
- 18 – 22 BYTES: QUALITY INFORMATION – 5 bytes

Tracking Data Transmission Formats for RADAR 2, RADAR3

The length is 18 bytes

- 1 – 3 BYTES :TIME – MM SS CS – 3 bytes
- 4 BYTES :STATION CODE – 1 byte
- 5 – 7 BYTES :RANGE – TENS OF METERS 3 bytes
- 8 – 10BYTES :AZIMUTH – CENTI MILS – 3 bytes
- 11 – 13 BYTES :ELEVATION – CENTI MILS 3 bytes
- 14 – 16 BYTES :QUALITY WORD – 3 bytes
- 17 BYTES :AZ RF – COUNT - 1 byte
- 18 BYTES :EL RF – COUNT - 1 byte

III SYSTEM ANALYSIS

System Analysis is the process of gathering and interpreting facts, diagnosing problems and using this information to recommend improvements to the system.

The existing Radar Data System tracks data about the target and transmits to the mission control systems. This system has the following features only. The Radar Data System collects all information of the target in counts of time, range, azimuth, elevation AGC, RF errors and quality information. This data is sent to mission control systems. The mission control systems checks for the validity of the existing software. The Radar Data system transmits 22/18 Bytes of tracking data at 10 Hz to mission computers. There is no provision for simulating tracked data till now [9].

IV PROPOSED SYSTEMS

Instead of the existing Rader Data System we have developed a Radar Data Analysis tool to simulate the Radar Data. The Radar data is stored as a file and is taken as input to our software.

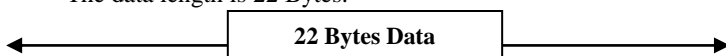
The data contains countdown time, range, azimuth, elevation, AGC, RF errors and quality information. The data is computed to the respective engineering units.

The data is converted into Data Formats in Hexadecimal form. The data frames are generated at 10 Hz into 22/18 Bytes data formats. These data frames can be transmitted to mission systems to check the validity of the software. As shown in fig

1 2 3 4	5 6 7 8	9 10 11	2 13 14	5 16 17	18 19 20 21 22
xxxxxxxx	xxxxxxxx	xxxxx	xxxxx	xxxxx	xxxxxxxx

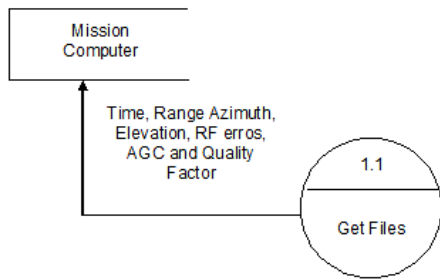
Tracking Data Transmission Format For PCMC 2

The data length is 22 Bytes.

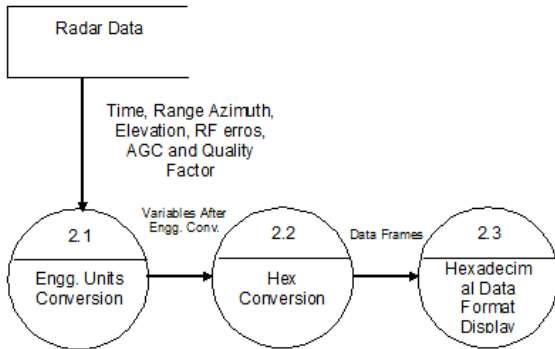


- 1 – 3 BYTES: TIME – MM SS CS – 3 bytes
- 4 BYTES: STATION CODE – 1 byte
- 5 – 8 BYTES : RANGE – METERS 4 bytes
- 9 – 11BYTES : AZIMUTH – CENTI MILS – 3 bytes

Second Level DFD for Collecting Files



Second Level DFD for Processing Data



V FUNCTIONAL DESCRIPTION

The following are the functions of the tool to simulate radar data. Reading the raw data from file. Converting all parameters into their respective engineering units. Converting into hexadecimal values. Display these hexadecimal values as 22/18 bytes data formats of radars.

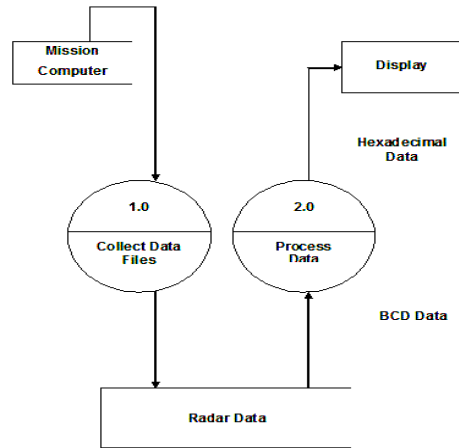
VI SYSTEM TESTING

System testing is the usage of implementation, which is aimed at ensuring that system works accurately and efficiently before live operation commences. Testing is vital to success of any system. System testing makes a logical assumption that if all the parts of the system are correct, the goal will be successfully achieved. The candidate system is subject to a variety of tests, on line response, volume, stress, recovery and security and usability tests. A series of testing are performed for the proposed system before the system is ready for user acceptance testing.

The testing steps are:

- Unit testing
- Code walk-through
- Integrated testing
- Validation
- Output testing
- System Testing
- User acceptance testing

First Level Data Flow Diagram of Tool to Simulate Radar Data



VII IMPLEMENTATION

During implementation stage, the system is physically created. Necessary programs are coded, debugged and documented. Operating procedures are developed. The implementation normally ends with a formal system test involving all exponents and procedures. The system is developed in Turbo C++ and successfully tested with real time data[1].

SPECIFICATION

Hardware Specifications	:System Installations
Main memory	:16 MB RAM
Hard disk	:2.1 GB
Floppy disk	:1.44 MB
Monitor	:SVGA
Cache memory	:512 KB
Printer	:HP DeskJet 1000C

Software Specifications	:
Operating System	:MS-DOS 6.22
Language	:Turbo C++

VIII DESIGN DESCRIPTION OF MODULES

The following functions are used in the software to get the resultant output:main(), flytime(), slanrange(),elevation(), azimuth(), difr(), difa(), difele()

8.0: main()

Input: Raw radar data from file

Functional Description:

Main() reads radar data from file. Functions main(), flytime(), slanrange(),elevation(), azimuth(), difr(), difa(), difele()are all called. These called functions in turn calls system function setf() which converts all the computed engineering units of the respective variables used in the above function to hexadecimal values. Then using the system functions cout(), fwrite() the converted hexadecimal values are displayed on the screen and written in the output file. The display is in the form of 22/18 bytes data formats for the PCMCs and RADARs respectively. Then the graphs for the parameters Range, Azimuth and Elevation with respect to Time are plotted.

8.1 Interface with other modules

main(), flytime(), slanrange(),elevation(), azimuth(), difr(), difa(), difele()

Out Put:The display is in the form of 22/18 bytes data formats for the PCMCs and RADARs respectively. Then the graphs for the parameters Range, Azimuth and Elevation with respect to Time are plotted.

Flytime ()

Input: Gets the time value from the already read input file.

Functional description:This function reads the time in seconds from the file and converts into HHMMSSCS format in 4 bytes and displays in the same fashion on screen and writes in the output file.

Output: Integer value is displayed in BCD format.

Slant range

Input: Gets range value from the already read input file.

8.2 Interface with Other Modules:

Setf () system function is used to convert the values into hexadecimal output.

Functional Description:This function reads range in kilometers from the file and converts into meters and converts into hexadecimal value using setf() into 4bytes of length and displays in the same fashion on screen and writes in the output file.

Out Put:

Displays in 4bytes length on the screen and writes in the output file in hexadecimal format.

Azimuth ()

Input: Gets azimuth and elevation values from the already read input file.

8.3 Interface with Other Modules:

Setf() system function is used to convert the values into hexadecimal output.

Functional DescriptionThis function reads azimuth and elevation values in degrees from the file and computes into centi mils and converts into hexadecimal value using setf() into each 3 bytes of length and displays in the same fashion on screen and writes in the output file.

Output: Displays each in 3 bytes length on the screen and writes in the output file in hexadecimal format.

Input: Gets range rate value from the already read input file which is fixed as 00.

8.4 Interfaces with Other Modules:

Setf() system function is used to convert the values into hexadecimal output.

Functional Description:This function reads rangerate in meters per sec from the file and converts into hexadecimal value using setf() into 3 bytes of length and displays in the same fashion on screen and writes in the output file.

Output:Displays in 3 bytes length on the screen and writes in the output file in hexadecimal format.

Functional Description:This function reads az error as correction factor from the file and converts into hexadecimal value using setf() into 1 byte of length and displays in the same fashion on screen and writes in the output file.

Output: Displays in 1 bytes length on the screen and writes in the output file in hexadecimal format.

difr():

Input: Getting the differences in slant range

Output: displaying the differences in slant range in 1 byte length and write the output

difaz ():

Input: Getting the differences in azimuth

Functional Description: This function reads az error as correction factor from the file and converts into hexadecimal

value using setf() into 1 byte of length and displays in the same fashion on screen and writes in the output file.

difele ()

Input: Getting the differences in elevation

Output: displaying the difference output of the elevation

8.5 Interfaces with other modules;

The system files initgraph() is used for initializing graphical mode and graphical drive. System function line() is used for drawing coordinate axis and defining scale. System functions setcolor(), settextstyle(), moveto(), lineto() are used for setting colours for the graph, for changing the text fonts and size, to move to particular point and draw line to the destination point respectively.

Functional Description:The coordinate axis is defined and scales are measured. The graph is plotted between the Time and Range using *moveto()* and *line()* functions according to the defined scale and axes.

Output: Graph 1

tazchange()

8.6 Inter faces with other modules:

The system files initgraph() is used for initializing graphical mode and graphical drive. System function line() is used for drawing coordinate axis and defining scale. System functions setcolor(), settextstyle(), moveto(), lineto() are used for setting colours for the graph, for changing the text fonts and size, to move to particular point and draw line to the destination point respectively.

Functional Description:The coordinate axis is defined and scales are measured. The graph is plotted between the Time and Azimuth using *moveto()* and *lineto()* functions according to the defined scale and axes.

Output: Graph 2

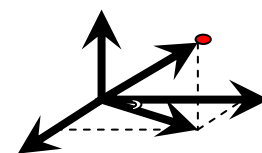
Telechange ()

Input: Reads Time and Elevation values from file.

8.7 Inter faces with other modules:

The system files initgraph() is used for initializing graphical mode and graphical drive. System function line() is used for drawing coordinate axis and defining scale. System functions sector(), settextstyle(), moveto (), lineto() are used for setting colours for the graph, for changing the text fonts and size, to move to particular points and draw line to the destination point respectively.

Functional Description:The coordinate axis is defined and scales are measured. The graph is plotted between the Time and Range using *moveto()* and *lineto()* functions according to the defined scale and axes.



Output: Graph 3.

8.7 Coordinate System

The Topocentric Relative Coordinate System is a Coordinate system with station as the origin on the Earth. Coordinate System with station as the origin

on the earth. The three quadrants are towards, East, North, and Vertical axis towards north is the horizontal and towards west is the vertical axis

IX `RESULTS AND CONCLUSION

In this paper we have discussed a simple fundamental analysis using Radar data by C oriented approach. By this we can do radar data analysis in an effective manner without an error and the advantage is decoding process is easy and faster execution and also we can develop analysis tools in other software technology also.

slant range

REFERENCES

- [1] Balaguru Swamy ,Programming in ANSCII 'C' - TMH Publishers
- [2] Yashavant Kanetkar ,Graphics Under C-BPB Publishers
- [3] James A. Senn, System Analysis and Design - MGH Publishers
- [4] M.L. Stone and G.P. Banner, "Radars for the Detection and Tracking of Ballistic Missiles, Satellites, and Planets," Linc Lab. J., in this issue.
- [5] Dixon,Michael,GerryWiener, 1993: TITAN: Thunderstorm Identification, Tracking, Analysis, and Nowcasting—A Radar-based Methodology. J. Atmos. Oceanic Technol., 10, 785–797.
- [6] P.A. Ingwersen and W.Z. Lemnios, "Radars for Ballistic Missile Defense Research," Linc. Lab. J., in this issue.
- [7] James, Curtis N., Stacy R. Brodzik, Harry Edmon, Robert A. Houze, Sandra E. Yuter, 2000: Radar Data Processing and Visualization over Complex Terrain*. Wea. Forecasting, 15, 327–338.
- [8] Waldteufel, P., H. Corbin, 1979: On the Analysis of Single-Doppler Radar Data. J. Appl. Meteor., 18, 532–542.
- [9] M.I. Skolnik, Introduction to Radar Systems, 2nd ed. (McGraw-Hill, New York, 1980), p. 432.[10] R.K. Avent, J.D. Shelton, and P. Brown, "The ALCOR CB and Imaging Radar," IEEE Antennas Propag. Mag. 38 (3),1996, pp. 16–27.
- [10] K.M. Cuomo, J.E. Piou, and J.T. Mayhan, "Ultra-WidebandCoherent Processing," Linc. Lab. J. 10 (2), 1997, pp. 203–222.
- [11] Schaffner, M.R.Computers, IEEE Transactions on Volume: C-27 , Issue: 11

