<u>10th June 2015. Vol.76. No.1</u>

 $\ensuremath{\mathbb{C}}$ 2005 - 2015 JATIT & LLS. All rights reserved $^{\cdot}$

ISSN: 1992-8645

www.jatit.org



SAYED CHHATTAN SHAH

Assistant Professor

Department of Information Communications Engineering

Hankuk University of Foreign Studies, South Korea

E-mail: shah@hufs.ac.kr

ABSTRACT

Korean wide area differential global navigation satellite system augments global navigation satellite system by broadcasting additional signals from geostationary satellites and providing differential correction messages and integrity data for the GNSS satellites. It includes a network of wide area reference stations, wide area master station, ground earth station and geostationary satellites. Wide area reference stations are widely dispersed GNSS data collection sites that monitor and process satellite data to determine satellite orbit and clock drift plus delays caused by atmosphere and ionosphere. This information is then transmitted to wide area master station which creates and broadcasts correction messages through geostationary satellites. The user segment receives and applies correction messages to improve position accuracy and reliability. This study presents a flexible and robust software design and data processing algorithms of user segment of Korean wide area differential global navigation satellite system. The user segment software performs numerous functions such as calculation of ionosphere and troposphere delays, processing of correction messages, and data quality monitoring. It implements numerous tropospheric, ionospheric and position models, supports RINEX and BINEX data exchange formats, and is designed to work in real time and post processing modes. It can also be used in precision and non-precision approach modes. The software is divided into several layers such as Data Processing and Visualization, and can be easily extended to support various interfaces such as web interface and mobile device interface. The current version processes global positioning system and wide area differential global navigation satellite system data but can be easily extended to support various global navigation satellite systems such as GLONASS and Galileo.

Keywords: Global Navigation Satellite System, Global Positioning Systems, GPS Augmentation System

1. INTRODUCTION

Korean wide area differential global navigation satellite system is a large scale research project funded by Ministry of Land. Transport and Maritime Affairs [1]. It augments global navigation satellite system [4] by broadcasting additional signals from geostationary satellites and providing differential correction messages and integrity data for the GNSS satellites. It includes a network of wide area reference stations, wide area master station, ground earth station and geostationary satellites. Wide area reference stations are widely dispersed GNSS data collection sites that monitor and process satellite data to determine satellite orbit and clock drift plus delays caused by atmosphere and ionosphere. This information is then transmitted to wide area master station which creates and broadcasts correction messages through geostationary satellites. The user segment receives and applies correction messages to improve position accuracy and reliability [2].

E-ISSN: 1817-3195



Figure 1: Korean WA-DGNSS Architecture design and data processing algorithms of user

<u>10th June 2015. Vol.76. No.1</u>

© 2005 - 2015 JATIT & LLS. All rights reserved

ISSN: 1992-8645	www.iatit.org	E-ISSN: 1817-3
10011.1772 0010	in in in . Juill. OIL	E 10011. 1017 0

segment of Korean wide area differential global navigation satellite system [3]. The user segment software performs numerous functions such as calculation of ionosphere and troposphere delays, processing of correction messages, and data quality monitoring. It implements numerous tropospheric, ionospheric and position models, supports RINEX and BINEX data exchange formats, and is designed to work in real time and post processing modes. It can also be used in precision and non-precision approach modes. In the precision approach mode, fast, slow, ionospheric and tropospheric corrections are applied while in the non-precision approach mode only fast and slow corrections are applied [4].

The software is divided into several layers such as Data Processing and Visualization, and can be easily extended to support various interfaces such as web interface and mobile device interface. The current version processes global positioning system and wide area differential global navigation satellite system data but can be easily extended to support various global navigation satellite systems such as GLONASS and Galileo [4].



Figure 2: User Segment Software Block Diagram

2. ARCHITECTURAL DESIGN

Architecture of user segment software is given in Figure 3 which shows the relationship between key components. For a detailed description, readers are referred to [5].

3. DATA PROCESSING ALGORITHMS

Wide area master station broadcasts three types of correction messages through geostationary satellites. 1) Fast Corrections for satellite clock's rapid and short-term errors, 2) Slow Corrections for satellite clock's slow drift errors and slow ephemeris errors, and 3) Ionospheric Grid Point Corrections for the estimated ionosphere signal propagation delays [2]. A block diagram of corrections message processing is given in Figure 4. In this section, key algorithms required to process and apply fast and slow corrections messages are described. For a detailed description of data processing algorithms, readers are referred to [6].



Figure 4: Corrections Message Processing Block Diagram

3.1 Processing of Fast Corrections

Fast corrections are used to correct fast-changing parameters such as satellite clock errors, and are included in message type 2-5 and 24. The fast corrections message contains pseudorange corrections and user differential range correction indicator. PRC is used to remove pseudorange error whereas UDREI is satellite health parameter which takes values in range of 1 to 15.

<u>10th June 2015. Vol.76. No.1</u>

© 2005 - 2015 JATIT & LLS. All rights reserved



<u>10th June 2015. Vol.76. No.1</u>

 $\ensuremath{\mathbb{C}}$ 2005 - 2015 JATIT & LLS. All rights reserved $^{\cdot}$



Notation	Description
IODP	Issue of Data PRN
MT	Message Type
PA	Precision approach
NPA	Non precision approach
PRC	Previously received pseudorange
previous	corrections
I _{fc}	Time-out interval for PRC
RRC	Range rate corrections
PRC	Pseudorange corrections
t.	Time of applicability of the most
of ,current	recent received pseudorange
	correction.
t.	Time of applicability of the
of , previous	previously received pseudorange
	correction.
FC	Total fast correction
t	User time
t	Time of applicability of the most
of	recent fast correction

3.2 Application of Fast and Slow Corrections

Fast corrections are used to correct fast-changing parameters whereas slow or long term corrections are used to provide error estimates for slow varying satellite ephemeris and clock errors, and are included in message type 24 and 25 [7]. The slow corrections message contains satellite position corrections, velocity corrections and satellite clock corrections. Message type 25 may be broadcast in two forms as per requirement. The Figure 6 shows the schematic of each form.



Figure 6: Slow Corrections Message Type 25

10th June 2015. Vol.76. No.1

 $\ensuremath{\textcircled{\sc 0}}$ 2005 - 2015 JATIT & LLS. All rights reserved

0 2000 2010	e, a. 2201,	JITAL
ISSN: 1992-8645	www.jatit.org	E-ISSN: 1817-3195
The algorithm for application of correction	IF (NPA mode == true)	
messages is given below.	IF (Fast and slow correcti Apply fast corrections	ons are available) s:
Check Integrity	$PR = PR_{measure}$	$_{ed}$ + FC + CC
IF (UDREI==15)	Apply slow correction	is:
Satellite = unhealthy	$\begin{bmatrix} \mathbf{X}_{\text{corrected}} \end{bmatrix} \begin{bmatrix} \mathbf{Z}_{\text{corrected}} \end{bmatrix}$	\mathbf{x}_{GPS} $\delta \mathbf{x}$
IF (UDREI==14 DGNSS data == invalid IOD! =IODE)	$\begin{bmatrix} \mathbf{y}_{\text{corrected}} \\ \mathbf{z}_{\text{corrected}} \end{bmatrix} = \begin{bmatrix} \mathbf{z}_{\text{corrected}} \\ \mathbf{z}_{\text{corrected}} \end{bmatrix}$	$ \begin{vmatrix} y_{\text{GPS}} \\ z_{\text{GPS}} \end{vmatrix} + \begin{vmatrix} \delta y \\ \delta z \end{vmatrix} $
Satellite = unmonitored	IE (DGPS iono correct	ions are available)
<pre>IF (PA mode == true) IF (UDREI in MT 2-5 & 24 >= 12) Satellite Status = Do not use IF (PA mode==true) IF (Correction message is missed & Previously received correction data == Apply degradation models Apply fast, ionospheric, tropospheric an clock corrections: PR = PR + FC - ICorrect + TCorrect</pre>	F (DGPS iono correct Apply DGPS ionos PR = PR measured + 1 ELSE IF (GPS iono corr Apply GPS iono PR = PR measured PR = PR measured PR = PR measured PR = PR measured + 1 Do not apply ion d IF (DGPS tropo correct Apply DGPS tropo PR = PR measured + F CC ELSE	Tons are available) pheric corrections: $FC - IC_{SBAS} + CC$ rections available) ospheric corrections: $d + FC - IC_{GPS} + CC$ to spheric corrections tions are available) ospheric corrections: $FC - IC_{SBAS} + TC_{SBAS} + CC$
TR = TR measured + TC = TC SBAS + TC SBAS +	Do not apply tr	opo corrections
Apply long term corrections:	ELSE Do not use the satellit	te
$\begin{bmatrix} x_{corrected} \\ y_{corrected} \\ z_{corrected} \end{bmatrix} = \begin{bmatrix} x_{GPS} \\ y_{GPS} \\ z_{GPS} \end{bmatrix} + \begin{bmatrix} \delta x \\ \delta y \\ \delta z \end{bmatrix}$ IF (Correction messages are not missed &	3.3 Calculation of Pseudora Pseudorange residual impl obtained after eliminating e clock bias, ionospheric dela and clock error from the sm	ies the remaining error errors such as satellite ay, tropospheric delay, oothed pseudorange of
fast, slow, ionospheric & tropospheric corrections are available)	satellite j. An algorithm to residual is given below.	calculate pseudorange
Apply fast, ionospheric, tropospheric clock corrections:	and a) Calculate Distance betw and Satellite j	een Reference Station

$$\label{eq:pressure} \begin{split} PR &= PR_{\rm measured} + FC - IC_{\rm SBAS} + TC_{\rm SBAS} + CC \\ \text{Apply long term corrections:} \end{split}$$

$$\begin{bmatrix} \mathbf{x}_{corrected} \\ \mathbf{y}_{corrected} \\ \mathbf{z}_{corrected} \end{bmatrix} = \begin{bmatrix} \mathbf{x}_{GPS} \\ \mathbf{y}_{GPS} \\ \mathbf{z}_{GPS} \end{bmatrix} + \begin{bmatrix} \delta \mathbf{x} \\ \delta \mathbf{y} \\ \delta \mathbf{z} \end{bmatrix}$$

ELSE

PA not possible Exit _____

$$D_j = \sqrt{(x - x_j)^2 + (y - y_j)^2 + (z - z_j)^2}$$

 $\left(x,y,z\right)$ Reference Station Coordinates

 $\left(\boldsymbol{x}_{j},\boldsymbol{y}_{j},\boldsymbol{z}_{j}\right)$ Satellite j Coordinates,

Satellite ephemeris error is included

b) Calculate Pseudorange

 $PR_{j} = D_{j} + d_{iono} + d_{tropo} - \Delta t_{sv} + B + e_{p}$

 e_{p} Measurement noise

B is receiver clock error

<u>10th June 2015. Vol.76. No.1</u>

 $\ensuremath{\mathbb{C}}$ 2005 - 2015 JATIT & LLS. All rights reserved $^{\cdot}$

ISSN: 1992-8645	www.jatit.org	E-ISSN: 1817-3195

c) Smooth Pseudorange

Pseudorange $\mathbf{p1}_k$, $\mathbf{p2}_k$ of satellite j is available from the navigation message.

Smoothed pseudorange is obtained by

$$\begin{split} &\rho\widehat{1}_{k} = \frac{K-1}{K} [\rho\widehat{1}_{k} + \bigtriangleup \Phi 1_{k} - \frac{2}{\gamma-1} \varDelta (d\Phi_{k})] + \frac{1}{K} \rho 1_{k} \\ &\rho\widehat{2}_{k} = \frac{K-1}{K} [\rho\widehat{2}_{k} + \bigtriangleup \Phi 2_{k} - \frac{2\gamma}{\gamma-1} \varDelta (d\Phi_{k})] + \frac{1}{K} \rho 2_{k} \\ &d\phi_{k} \equiv \phi 2_{k} - \phi 1_{k} \\ &\bigtriangleup (d\phi_{k}) = d\phi_{k} - d\phi_{k-1} = \bigtriangleup \phi 2_{k} - \bigtriangleup \phi 1_{k} \\ &\bigtriangleup \Phi_{k} = \Phi_{k} - \Phi_{k-1} \end{split}$$

K: Averaging smoothing constant, K=300

k: Epoch

d) Calculate Pseudorange Residuals

 $PRR_{i} = SPR_{i} - D_{i} - d_{iono} - d_{tropo} + \Delta t_{sv} - B$

4. CLASS DIAGRAMS

Class diagram is used to model the static view of a system. It includes classes, their attributes, operations and relationships with other classes [8]. Rectangles are used to represent classes.

The most important classes of user segment software of Korean wide area differential global navigation satellite system are shown in Figure 7. The classes are divided into five groups, each represented with a different color. The classes in light green color are used to represent raw and preprocessed data whereas classes in light purple color are used to represent fast and slow corrections data. Classes in sky blue and orange colors are data processing classes that are used to process raw global positioning system data and wide area differential global navigation satellite system correction messages, respectively. The classes in tan color are used to parse RINEX navigation and observation files and BINEX files. For a detailed description of each class, readers are referred to [5].

5. SOFTWARE MODULES

User Segment software provides an interface to access numerous modules in real time and post processing modes. The key modules of user segment software are briefly described below.

Data Handlers

Data Handlers are used to hold raw and preprocessed data during the execution. Some classes such as RawPre-processedDataBuffer, GlobalData and Time are used to buffer raw and preprocessed data for further processing and provide utility functions to convert between time standards and coordinate systems.

Data Processor

Data Processor is used to process raw data. The raw data is accessed from data handler classes and after processing either is made available to user interface classes for display or transmitted across the network. Numerous tropospheric, ionospheric and position models are implemented in this module. The diagram of data processing classes is given in Figure 8.



Figure 8: Data Processing Classes

<u>10th June 2015. Vol.76. No.1</u>

© 2005 - 2015 JATIT & LLS. All rights reserved

ISSN: 1992-8645

www.jatit.org

RINEX Parser

RINEX Parser is used to parse RINEX navigation and observation files. It includes two key classes: RINEXNavigationParser which is used to parse navigation files and RINEXObservationParser which is used to parse observation files. Both classes are accessed through RINEXParser class which provides an interface to rest of the components in the system. It can be used in real time or post processing mode. The class diagram of RINEX Parser is given in Figure 9.

BINEX Parser

BINEX Parser provides an interface to parse BINEX files [10] in real time or post processing modes. It is designed to support:

Record 0x01 = 1 for GNSS navigation information

Record 0x7f Subrecord 0x05 for GNSS Observables

The class diagram of BINEX Parser is given in Figure 10.

Correction Messages Parser

Correction Messages Parser module is used to parse wide area differential global navigation satellite system's correction messages broadcast by geostationary satellites in a binary format. A list of correction messages is given below.

T	able	2:	Α	list	of	COP	rectio	on	messages
---	------	----	---	------	----	-----	--------	----	----------

Message Type	Contents	Related Message	Relationship Parameter
0	Don't use for safety applications (Testing)		
01	PRN Mask Assignments	02-05, 07, 24, 25, 28	IODP
02~05	Fast Correction	06	IODF
06	Integrity Information	02-05, 24	IODF
07	Fast Correction Degradation Factor	01	IODP
09*	Geo Navigation Messages (X,Y,Z Time etc)		
10	Degradation Parameters		
12	SBAS Network Time, UTC Offset Parameters		
17	GEO SV Almanac		
18	Ionospheric Grid Point Mask	26	IODI
24	Mixed Fast Corr. / Long time SV Error Corr.		IODF
25	Long time satellite error corrections	01	IODP
26	Ionospheric Delay Corrections	18	IODI
27	SBAS Service Message	27	IODS
28	Clock Ephemeris Covariance Matrix Message	01	IODP
29~61	Reserved for Future Messages		
62	Internal Test Message		
63	Null Message		

User Interface

User interface module provides an access to user segment and allow user to visualize, monitor and analyze system's performance. It includes several screens that are used to display raw and preprocessed data and numerous static and dynamic graphs related to raw and preprocessed data. Key user interface elements are described below.

E-ISSN: 1817-3195

User Segment Software Screen given in Figure 11 allows user to launch software either in real time processing mode or post processing mode.



Figure 11: User Segment Software Screen

User Segment Software Real Time Processing Mode screen given in Figure 12 allows user to use various components of User Segment software in real time processing mode.

Receiver Activity Screen given in Figure 13 is used in a real time processing mode to display several important items to help user to understand how the receiver is being used and its current operating conditions.

6. SOFTWARE CHARACTERISTICS

User Segment software of Korean wide area differential global navigation satellite system is made of several independent and reusable modules that can be modified and tested separately. The modules can be easily added or replaced into user segment software. The current version processes global positioning system and wide area differential global navigation satellite system data but can be easily extended to support various global navigation satellite systems such as GLONASS and Galileo.

<u>10th June 2015. Vol.76. No.1</u>

© 2005 - 2015 JATIT & LLS. All rights reserved

www.jatit.org

JATIT
E-ISSN: 1817-3195

7. CONCLUSION

ISSN: 1992-8645

The user segment software performs numerous functions such as calculation of ionosphere and troposphere delays, processing of correction messages, and data quality monitoring. It is made of several independent and reusable modules that

In this study a detailed design of user segment software of Korean wide area differential global navigation satellite system is described in terms of UML architectural and class diagrams. The study also describes key algorithms for processing correction messages and key user interface elements.

can be modified and tested separately.

ACKNOWLEDGEMENT

This research paper was supported by Hankuk University of Foreign Studies Research Fund of 2015.

REFRENCES:

- Yun, H., C.D. Kee and D.Y. Kim, 2011. Availability performance analysis of Korean wide area differential GNSS test bed. J Korea Navigation Inst., Vol. 15, pp. 510-516.
- [2] FAA Document, 1999. Specification of the Wide Area Augmentation System (WAAS). Report No. FAA-E-2892B, April 1999.
- [3] Grewal, M.S., L.R. Weill and A.P. Andrews, 2007. Global Positioning Systems, Inertial Navigation and Integration. 2nd Edition, John Wiley and Sons, New York, ISBN: 9780470099711, Pages: 416.
- [4] Blanch, J., T. Walter, P. Enge, S. Wallner and F.A. Fernandez et al., 2013. Critical elements for a multi constellation advanced RAIM. Navigation, Vol. 60, pp. 53-69.
- [5] Shah S.C., Choi W.S., Han W.Y., Yun H., Kee C., 2013. Korean WA-DGNSS User Segment Design. TransNav, Vol. 7, No. 1, pp. 69-74.
- [6] Shah S.C., W.S. Choi and W.Y. Han, 2012. Detailed design of Korean WA-DGNSS user segment software. Report No. 6060-2012-042, Electronics and Telecommunications Research Institute.
- [7] RTCA SC-159, 2001. Minimum operational performance standards for global positioning system and wide area augmentation system airborne equipment, RTCA/DO-229C. November 28, 2001, Washington, DC.

- [8] Tegarden, D.P., A. Dennis and B.H. Wixom, 2012. Systems Analysis and Design with UML. 4th Edition, Wiley, New York, ISBN: 9781118092361, Pages: 592.
- [9] Yun, H., C. Kee and D. Kim, 2011. Korean wide area differential global positioning system development status and preliminary results. Int. J. Aeronautical Space Sci., Vol. 12, pp. 274-282.
- [10] http://binex.unavco.org, accessed on 1-10-2014.

Journal of Theoretical and Applied Information Technology <u>10th June 2015. Vol.76. No.1</u>

© 2005 - 2015 JATIT & LLS. All rights reserved.



ICCNI- 1002 0645 intit are E ICCNI 1017 2105 日 Navigation Message Decoder 別 Navigation Message Reciever Data Handler nanac Data SBAS GEO SV Data Handler GPS SV Clock Data Handler GPS Ionospheric Data Handler 8 8 8 ata SBAS Slow Corr Data Handi SBAS Fast Cor Data Han GPS Ephemeris Handler CDS Ale SBAS Io Correct H Q C 訇 Integrity Monitor Data Quality Monitor SBAS Integrity Monitor GPS SV DQM Ð GPS Epher E 罰 GPS SV Inte tion Lev GPS lonos, DQM C Calcula Ó Ð Data Processor SBAS Slo rrections Data Processor AS GEO SV Data PS SV Clock Data «device» Tropospheric Data ric Delay 割 GPS Ep SBAS Fast GPS Data Calculator Processor dorange Calculator V AL Ind Elr Cal Posit Calcu ą Independent Data Verifier 0 d Pre-ni ed Data B ata Disp Data Logg Unit

Figure 3: User Segment Software Architecture

Journal of	Theoretica <u>1</u>	1 and Applie 0 th June 2015. Vo	ed Inform	ation Technology	′
	© 2005 - 20	15 JATIT & LLS	. All rights res	erved.	JATT
ISSN: 1992-8645		www.jatit.org	2	E	E-ISSN: 1817-3195
Observation Set	Time	Receiver P	osition	IGP Ionospheric	Corrections
SatelliteAzimuthElevation	Coordinat	ies Cos	r.stants	lonospherizCo.	rrections
Ephemeris L'ata	KawPrej	processed Datab	Juffer	DGPSNavigatio	n Message
Tropospheric⊃ata	DGPSC	orrection;	Metecn	ologicalParameters	XYZ
IonosphericData	Time ou	tInterval;	Degra	dationP grameters	DataLoger
Preprocessed Data	FaitCo	rrections	Lat:	itudeLongitude	CRC
Satell:tePosition	S1ow Co	rrections	Pseudor	range⊃ataProcessor	Matrix
IonosphericData Processor	Integri	ityMonitor	Ephem	ierisDataPrccessor	Complex
Tropospheric DataProcess of	r WA-D	GNSSU serSeg:	ment	PositionCalculator	BINEXParter
FastCorrection; Processor	SlowC	orrections Proc	esscr	DGPSNavigationMess	ageProcessor
EGFSTropo spheric DataP	rocessor	DG?SI5n	csphericDat	aProcessor R	INEXParser
GlobalDeta		RINEXNavig	ationParser	RINEXObse	rvationParser

Figure 7: Key Classes of User Segment Software

<u>10th June 2015. Vol.76. No.1</u>

© 2005 - 2015 JATIT & LLS. All rights reserved

ISSN: 1992-8645

www.jatit.org



RINEXNav igationParser ephemerislonoDataBuffer :RawPre-processedDataBuffer navigationFile :File addEphemeris(ephemerisData :EphemerisData) :void close() :void findEphemeris(utcTime :long, satelliteID :int) :EphemerisData getEphemerisSize() :int getIonosphericData(utcTime :long) :lonosphericData open() :bool parseData() :void RINEXParser + parseHeader() :void «constructor» navigationFile :File RINEXNavigationParser(fileName :File) :void observationFile :File RINEXNavigationParser() :void rawPre-processedData :RawPre-processedDataBuffer rinexNavigationParser :RINEXNavigationParser rinexObservationParser :RINEXObservationParser **RINEXObservationParser** parse() :bool numberOfObsTypes :int «property» numberOfSatellites :int navigationFile() :File observationDataBuffer :RawPre-processedDataBuffer observationFile() :File observationFile :File «constructor» orderOfObservationData :int ([]) RINEXParser() :void satelliteOrder :int ([]) RINEXParser(navigationFile :File, observationFile :File) :void timeOfFirstObservation :Time assignType(line :String, k:int, j :int, i :int) :void close() :bool getApproxPosition() :Coordinates getCurrentObservations() :ObservationSet + hasMoreObservations() :bool nextObservations() :ObservationSet + open() :bool parseAntDelta(line :String) :void parseApproxPos(line :String) :void + parseData() :void + parseHeader() :void + parseTimeOfFirstObservation(line :String) :void + parseTypes(line :String) :void «property» + observationFile() :File «constructor» + RINEXObservationParser(observationFile :File) :void RINEXObservationParser() :void

Figure 9: Class Diagram of RINEX Parser

Journal of Theoretical and Applied Information Technology <u>10th June 2015. Vol.76. No.1</u>

© 2005 - 2015 JATIT & LLS. All rights reserved

ISSN: 1992-8645

www.jatit.org



BINEXParser	crc	DataHandlers::Raw PreprocessedDataBuffer
GPS_PI :double = 3.1415926535898 {readOnly} _binexFile :String binaryReader :BinaryReader isBinexLittleEndian :bool forwardReadable :bool reverseBytes :bool ephemerisData crcBytes :int = 0 BINEXParser(String) openFile() :bool	 crc161ab ::ushort ([]) = new ushort[256] crc32tab ::uint ([]) = new uint[256] cks08(byte[]) ::byte crc16(byte[]) :ushort crc32(byte[]) :uint crc38(byte[], byte[]) :int crc32(byte[], byte[]) :ushort crc32(byte[], byte[]) :uint 	 ephemerisDataList :List<ephemerisdata></ephemerisdata> ionosphericDataList :List<ionosphericdata></ionosphericdata> observationsList :List<ionosphericdata></ionosphericdata> preprocessedDataList :List<preprocesseddata></preprocesseddata> troposphericDataList :List<toposphericdata></toposphericdata> satellitteHistoryList :List<statellittepreprocesseddata></statellittepreprocesseddata> smoothPseudorangeHistoryList :List<statellittepreprocesseddata></statellittepreprocesseddata> _wmsInformation :WMSInformation _wmsPostion :WRSPostion _inexFileHeader :RINEXObsFileHeader
open mich () sool parseBinex() :bool parseBinex() :bool checkCRC(byte[], int) :bool reverseByteArray(byte[]) :byte[] Combine(byte, byte) :ushort msecToDateTime(fong) :DateTime parseObData(BinaryReader) :bool Bytes2Int(byte []) :int GetBit(byte, int, int) :int GetBit(byte, int) :long GetBit(byte, int) :bool combineBytesbyte[]) :String	Crc16 - polynomial :ushort = 0xA001 - table :ushort ([]) = new ushort[256] + ComputeChecksum(byte[]) :ushort + ComputeChecksumBytes(byte[]) :byte[] + Crc16()	 RawPreprocessedDataBuffer() addSatelliteHistory(SatellitePreprocessedData) :bool getSatelliteHistory(SatellitePreprocessedData> addSmoothPseudorangeHistory(SmoothPseudorangeHistory) :bool getSmoothPseudorangeHistory(Int) :List<satellitepreprocesseddata></satellitepreprocesseddata> addSmoothPseudorangeHistory(SmoothPseudorangeHistory) :bool getSmoothPseudorangeHistory(SmoothPseudorangeHistory) addRinexFileHeader(RNEXObsFileHeader) :void getRinexFileHeader(I) :RINEXObsFileHeader) :void addEphemerisData(EphemerisData) :bool addObservations) :bool addObservations) :bool addObservations) :bool addPre_processedData(PreprocessedData) :bool addPre_processedData(PreprocessedData) :bool
SignExtention(Sting, int): Sting combineBytes BE(byte[]): long parseEphemerisData(BinaryReader): bool readSat(Dount(BinaryReader): int readSat(D_SatSystem(BinaryReader): satID_System read_ubnxi_LE(BinaryReader): read_ubnxi_results read_ubnxi_BE(BinaryReader): read_ubnxi_results GetBigEndianBytesFromDouble(double): double GetBigEndianBytesFromSingle(float): double GetBigEndianBytesFromSingle(float): double SwapInt16(short): short SwapUnt16(short): subort SwapInt13(unt): uint SwapInt13(unt): uint SwapInt164(long): long	HashAlgorithm Crc32 + DefaultPolynomial :UInt32 = 0xetb88320 + DefaultSeed :UInt32 = 0xfffffff - hash :UInt32 - seed :UInt32 - table :UInt32 ([]) - defaultTable :UInt32 ([]) + Crc32() + Crc32(UInt32, UInt32)	 cleanBufferedData() :bool getCurrentEphemerisData() :EphemerisData getCurrentDosephericData() :IonosphericData getCurrentPre_processedData() :ProprocessedData getCurrentPre_processedData() :ProprosphericData getEphemerisData() :TroposphericData getEphemerisData(Irime) :EphemerisData getEphemerisData(Time) :EphemerisData getEphemerisData(Time) :IonosphericData getObservations{Time) :IonosphericData getObservations[Time) :Observations getObservations[Time] :PreprocessedData getPre_processedData(Time) :PreprocessedData getProposphericData(Time) :PreprocessedData getProposphericData(Time) :TroposphericData
<u>SwapUInt64(ulong) :ulong</u>	 Initialize() :void HashCore(byte[], int, int) :void HashFinal() :byte[] Compute(byte[]) :UInt32 Compute(UInt32, byte[]) :UInt32 InitializeTable(UInt32, UInt32, I) CalculateHash(UInt32[], UInt32[] CalculateHash(UInt32[], UInt32] :byte[], int, int) :UInt32 UInt32ToBigEndianBytes(UInt32) :byte[] «property» HashSize() :int 	 getEphemerisDataList() :List<ephemerisdata></ephemerisdata> getIonosphericDataList() :List<ionosphericdata></ionosphericdata> getObservationsList() :List<observations></observations> getPre_processedDataList() :List<toposphericdata></toposphericdata> getTroposphericDataList() :List<toposphericdata></toposphericdata> logBufferedData() :bool epropertys wmsInformation() :WMSInformation wmsPosition() :WRSInformation wmsPosition() :WRSPosition

Figure 10: Class Diagram of BINEX Parser

Journal of Theoretical and Applied Information Technology <u>10th June 2015. Vol.76. No.1</u>

© 2005 - 2015 JATIT & LLS. All rights reserved



ISSN: 1992-8645	

www.jatit.org

Vser Segment Information Name ETRI GNSS User Segment IP Address 127.0.0.1 3000	Binex File C:\teqc\DOKD_120702 Relet Log File usLog.txt select	
Start Stop	Reciever Activity Ephemeris Data SBAS Parser	
of Visible Satellites: 9 cliver Time: 105222354000 msec caiver Date Time: 7/1/2012 11:59:14 PM caiver Clock Bias: 0 t Tropospheric Delay: 0 opospheric Delay: 0 opospheric Delay: 0 frequencies of the satellite Di 14 ag: False DE: 0 Residuals: 0 PR Residuals: 0 PR Residuals: 0 P Residuals: 0 P Longitude: 0 2 longitude: 0 2 longitude: 0		* E
tellite ID: 22 Sea Sea Proventional: 0 PR Residuals: 0 PR Residuals: 0 Ionospheric Delay: 0 Plongitude: 0 Plasitude: 0 Plasitude: 0		
tellite ID: 31 hg: False DE: 0 Residuals: 0 PR Residuals: 0 nospheric Delay: 0 Ionospheric Delay: 0 P Longitude: 0 P Lotitude: 0 : 2712.453125		
tellite ID: 12 ag: False DE: 0		-

Figure 12: User Segment Software Real Time Processing Mode Screen



Figure 13: Receiver Activity Screen