SoftwareX 6 (2017) 209–216

Contents lists available at ScienceDirect

SoftwareX

journal homepage: www.elsevier.com/locate/softx

Original software publication

BabelFish—Tools for IEEE C37.118.2-compliant real-time synchrophasor data mediation

M.S. Almas[a,*], L. Vanfretti[a,b], M. Baudette[a]

[a] SmarTS Lab, KTH Royal Institute of Technology, Stockholm, Sweden
[b] Research and Development Division, Statnett SF, Oslo, Norway

ARTICLE INFO

Article history:
Received 22 June 2016
Received in revised form 29 June 2017
Accepted 3 August 2017

Keywords:
IEEE Std C37.118.2-2011
LabVIEW
Phasor measurement unit
Phasor data concentrator
Standard implementation
Synchrophasors
WAMS
WACS
WAPS
WAMPAC

ABSTRACT

BabelFish (BF) is a real-time data mediator for development and fast prototyping of synchrophasor applications. BF is compliant with the synchrophasor data transmission IEEE Std C37.118.2-2011. BF establishes a TCP/IP connection with any Phasor Measurement Unit (PMU) or Phasor Data Concentrator (PDC) stream and parses the IEEE Std C37.118.2-2011 frames in real-time to provide access to raw numerical data in the LabVIEW environment. Furthermore, BF allows the user to select "data-of-interest" and transmit it to either a local or remote application using the User Datagram Protocol (UDP) in order to support both unicast and multicast communication.

In the power systems Wide Area Monitoring Protection and Control (WAMPAC) domain, BF provides the first Free/Libre and Open Source Software (FLOSS) for the purpose of giving the users tools for fast prototyping of new applications processing PMU measurements in their chosen environment, thus liberating them of time consuming synchrophasor data handling and allowing them to develop applications in a modular fashion, without a need of a large and monolithic synchrophasor software environment.

Code metadata

Current code version	*v1.1.0*
Permanent link to code/repository used of this code version	https://github.com/ElsevierSoftwareX/SOFTX-D-16-00050
Legal Code License	*GPLv3*
Code versioning system used	*GIT*
Software code languages, tools, and services used	*BabelFish V1: C++, Active X, LabVIEW* *BFE: LabVIEW*
Compilation requirements, operating environments & dependencies	*BabelFish V1: Visual Studio 2010 Ultimate Edition* *BFE: LabVIEW 2013 or newer*
If available Link to developer documentation/manual	https://github.com/SmarTS-Lab-Parapluie/BabelFish/tree/master/BabelFish%20V1/Documentation
Support email for questions	msalmas@kth.se

1. Introduction

Synchrophasor measurements from Phasor Measurement Units (PMUs) are currently being utilized to deploy various Wide Area Monitoring, Protection and Control (WAMPAC) Applications [1,2]. PMUs provide high resolution, time-synchronized voltage phasors, current phasors and frequency measurements that conform to the IEEE Std C37.118.1-2011 requirements [3]. PMUs stream out these synchrophasor measurements by using the synchrophasor data transmission format defined by the IEEE Std C37.118.2-2011 [4].

1.1. Motivation

The synchrophasor data transmission protocol governed by IEEE Std C37.118.2-2011 [4] is an application level protocol [5]. Synchrophasor measurements, once packaged as IEEE Std C37.118.2-2011 frames [4] and transmitted using Transmission Control Protocol (TCP)/IP [6], appear as bytes of data at the receiving end. In order to translate these bytes of data into useful and computable information, an IEEE Std C37.118.2-2011 protocol parser is required.

[*] Corresponding author.
E-mail addresses: msalmas@kth.se (M.S. Almas), luigiv@kth.se, luigi.vanfretti@statnett.no (L. Vanfretti), baudette@kth.se (M. Baudette).

http://dx.doi.org/10.1016/j.softx.2017.08.002

Currently synchrophasor measurements are available internally in PDC software, which may have built-in processing tools adaptors or extension capabilities. OpenPDC [7] for example allows development of new algorithms that exploits PMU data, but requires high proficient programming (.Net, Java, C++) [8] and development skills that the target audience (electrical power systems engineers, prominently MSc level students) are not equipped with.

The synchrophasor data mediator presented in this article named BabelFish Engine (BFE), is a real-time IEEE Std C37.118.2-2011 protocol parser developed entirely in LabVIEW environment [9]. BFE is a part of BabelFish (BF) tools which are developed in-house by SmarTS-Lab [10] at KTH Royal Institute of Technology.

The BabelFish Engine (BFE) brings the PMU measurements into LabVIEW, which provides a very user friendly development interface, enabling fast prototyping of algorithms based on live measurements by researchers/engineers without a strong background in programming and software development. BFE liberates the synchrophasor application developers from tedious communication protocol data handling procedure.

1.2. BabelFish (BF) tools

BF is made available in two versions. The first version named as BabelFish version 1 (BFv1) [11] implements low-level protocol and data handling in C++, while Active X provides coupling between C++ routines and the user through the LabVIEW environment. Its use is thus limited to an Operating System (OS) that can support Active X. The second version named as BabelFish Engine (BFE) aims to address this limitation while at the same time making it suitable for its use in the National Instruments embedded control platforms, mainly the NI-cRIO [12].

1.3. BabelFish Engine (BFE) significance

The BFE only requires the least amount of configuration to access real-time synchrophasor data. The user need only to provide IP address, port number and Device ID of the PMU or Phasor Data Concentrator (PDC) [13] stream to establish connection and initiate the real-time IEEE Std C37.118.2-2011 protocol parsing. BFE unwraps the PMU/PDC stream and provides access to the raw phasors, analogs, digitals, frequency and Rate of Change of Frequency (ROCOF) measurements available within the stream. BFE also provides other information such as data rate, nominal frequency, phasor type and phasor/analog/frequency data format of the PMU/PDC stream. Furthermore, BFE facilitates monitoring of important statuses of the PMU/PDC stream such as “PMU Time Quality”, “Unlocked Time”, “PMU Sync Error”[1] and “Configuration Change” as specified in IEEE Std C37.118.2-2011. BFE also allows the user to make polar plots of all the available phasor measurements in the PMU/PDC stream.

In addition to real-time data mediation, BFE allows the user to select particular “data-of-interest” from the overall parsed synchrophasor measurements to locally utilize them in a particular application. Furthermore, BFE facilitates to transmit this “data-of-interest” to a remote unicast or set of multi-cast devices for their utilization in WAMPAC applications. In this case, the end user only requires a UDP client application that can be independent of platform, Operating System (OS) and computer programming language, to receive this “data-of-interest”.

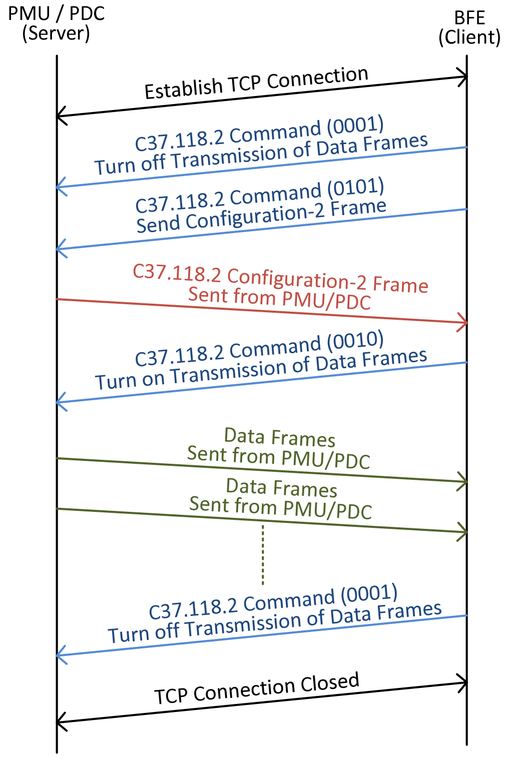


Fig. 1. Messages exchange between BFE and PMU/PDC for real-time IEEE Std C37.118.2-2011 parsing.

2. Software description

The BFE can connect to any PMU or PDC that is streaming data compliant to the IEEE Std C37.118.2-2011 [4]. BFE enables the user to configure three simple parameters, i.e. IP address, port number and Device ID of the PMU/PDC stream to establish a TCP/IP connection. As shown in Fig. 1, upon successfully establishing a TCP/IP connection, the BFE exchanges different messages with the PMU/PDC as specified in the IEEE Std C37.118.2-2011: to either turn on/off data transmission or requesting a configuration frame. The only messages that are sent from the BFE to a PMU/PDC are command messages. Using the data extracted from Configuration-2 frame [4] received from the PMU/PDC, BFE populates its GUI with meta-data, such as names of different PMUs available within the stream, the number of different phasors, analog and digital signals available in each PMU. At this stage, BFE also populates its GUI with information related to time synchronization, time accuracy and the configuration change status for each PMU available in the stream. The user can then turn on real-time data transmission from the PMU/PDC through the “Real-Time ON” option of the GUI. This results in real-time parsing of the incoming IEEE Std C37.118.2-2011 data frames and displays the parsed information through different indicators configured in the GUI. The phasor measurements are displayed using numerical values both in rectangular and polar coordinates. In addition, the BFE provides functionality to make polar plots of all the available phasor measurements in the PMU/PDC stream to provide holistic view of the power system to the user. The digitals are presented in the form of LEDs which turn “on” or “off” depending on the value of the digitals being received.

2.1. Software architecture

The BabelFish repository includes all associated software developed by KTH SmarTS-Lab to support the two versions of the

[1] The readers/users are requested to refer to the supplementary material (see Appendix A) submitted with this manuscript for complete documentation on BFE functionality of displaying “PMU Time Quality/Accuracy / Synchronization”.

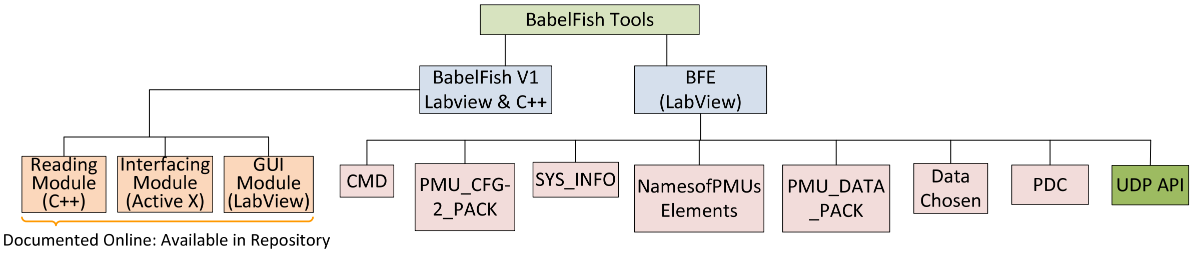


Fig. 2. Structure of the BabelFish repository and software modules.

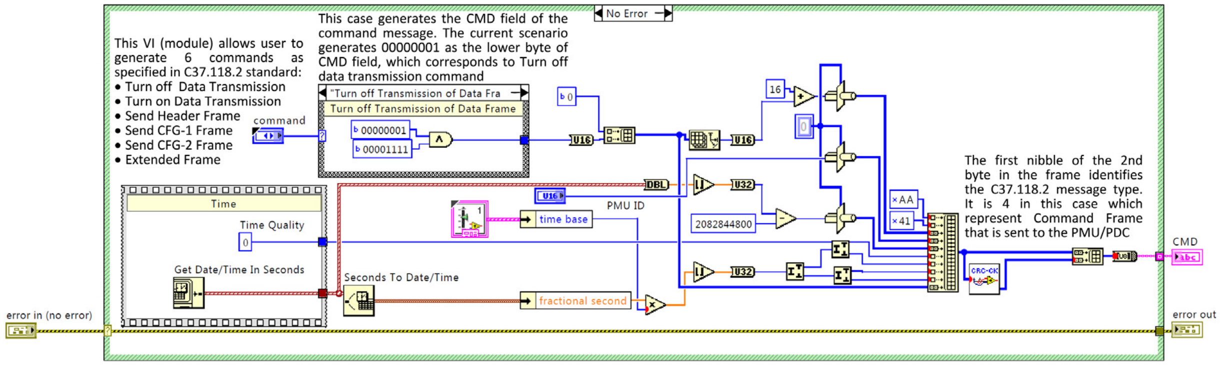

This VI (module) allows user to generate 6 commands as specified in C37.118.2 standard:
• Turn off Data Transmission
• Turn on Data Transmission
• Send Header Frame
• Send CFG-1 Frame
• Send CFG-2 Frame
• Extended Frame
This case generates the CMD field of the command message. The current scenario generates 00000001 as the lower byte of CMD field, which corresponds to Turn off data transmission command
No Error
command
"Turn off Transmission of Data Fra
Turn off Transmission of Data Frame
00000001
00001111
16
U16
U32
DBL
PMU ID
time base
2082844800
Time
Time Quality
Get Date/Time In Seconds
Seconds To Date/Time
fractional second
AA
41
The first nibble of the 2nd byte in the frame identifies the C37.118.2 message type. It is 4 in this case which represent Command Frame that is sent to the PMU/PDC
CRC-CK
CMD
error in (no error)
error out

Fig. 3. CMD VI code snippet.

real-time data mediator (see Fig. 2): (i) BabelFish V1, developed using both C++ and LabVIEW; and (ii) BFE, which is a LabVIEW project. BabelFish V1 is documented in [11] and online (available in the repository https://github.com/SmarTS-Lab-Parapluie/BabelFish). In this paper, BFE is described in more detail.

BFE was implemented entirely in LabVIEW, which is a visual programming language from National Instruments [9]. LabVIEW was selected because it enables to easily create graphical user interfaces (GUIs), facilitates smooth integration with other programming languages (e.g. Python and MATLAB), and because it is possible to compile LabVIEW programs for their use in National Instruments embedded platforms (i.e. NI-cRIO [12]). Moreover, LabVIEW provides numerous toolkits for advance analysis, hardware integration, data logging and report generation.

BFE was developed as a LabVIEW project with different functionalities organized in small code modules referred to as Virtual Instruments (VIs) in LabVIEW [14]. The structure of the BabelFish repository is shown in Fig. 2. A brief description of relevant VIs of BFE and their functionality are discussed in the next section.

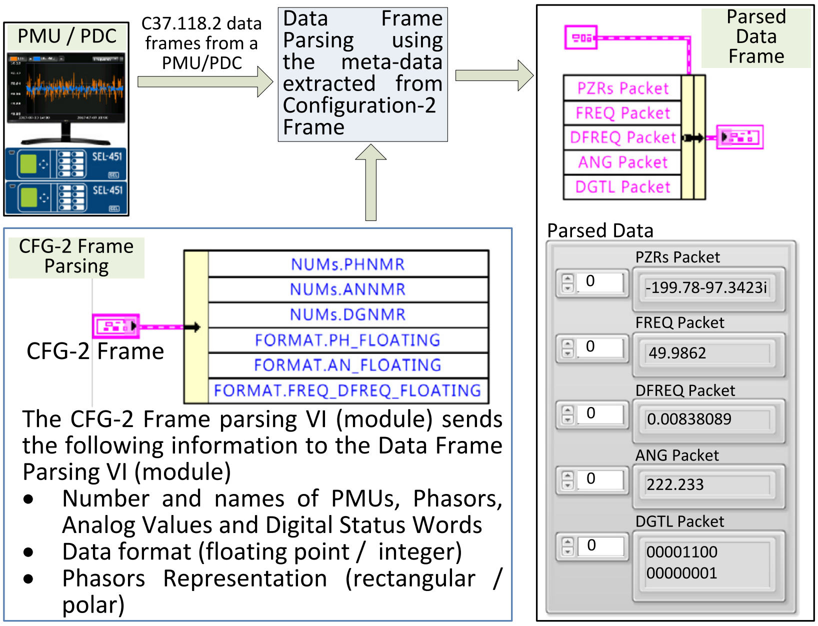
2.2. Software functionalities

As shown in Fig. 2, BFE consists of different VIs that perform specific tasks in communicating and parsing the PMU/PDC streams. The description of these VIs and the overall functionalities of BFE are discussed below.

1. CMD VI: This VI generates different command messages as specified by the IEEE Std C37.118.2-2011 [4]. When the user configures the IP Address, Port Number and Device ID of the PMU/PDC stream to be parsed, CMD VI is the first to execute after a successful TCP/IP connection. This VI sends command frames to the PMU/PDC to either turn on or off data transmission, requests Configuration-2 frame, etc., as specified in the standard. A code snippet of the CMD VI is shown in Fig. 3 and its functions are briefly explained within the figure.
2. PMU_CFG-2_PACK: If the synchrophasor stream consists of more than one PMU, this VI retrieves each Configuration-2 packet associated with each PMU available in the stream. The BFE automatically handles the incoming synchrophasor data format by parsing the "FORMAT" field of the "CFG-2" Frame.
3. SYS_INFO: This VI parses each Configuration-2 packet to reveal the time stamp, frame size, data rate, ID Code and time base for the PMUs available in the synchrophasor stream.
4. NamesofPMUsElements: This VI retrieves the names of different PMUs and their respective phasors, analogs and digitals as configured in the incoming synchrophasor stream.
5. PMU_Data_Pack: When the user enables the data transmission from a PMU/PDC through the GUI, this VI performs real-time parsing of the incoming IEEE Std C37.118.2-2011 data frames. The information from Configuration-2 frame is used to identify the number and data types for phasors, analogs and frequency measurements. The overall functionality of this VI is shown in Fig. 4. Inside BFE all the data is treated in the form of "string" data type which can be easily converted to other data-types for data manipulation/visualization.
6. DataChosen: This VI allows the user to select the "data-of-interest" either to use locally or to send it to a remote application using the UDP transmission protocol.
7. PDC: The GUI of the BFE is accessed through the PDC VI.
8. UDP API: This UDP Client application developed in LabVIEW establishes a UDP connection with the BFE and receives the "data-of-interest". Once the "data-of-interest" is selected, the user can configure the UDP ports of BFE to transmit this data to remote application as simple UDP messages.

2.3. Software testing

As a first step towards testing of the BFE, the network tool analyzer "Wireshark" [15] is utilized to ensure that the sequence of messages exchanged between the BFE (client) and the PMU/PDC (server) are correct i.e. according to the IEEE Std C37.118.2-2011



The CFG-2 Frame parsing VI (module) sends the following information to the Data Frame Parsing VI (module)

- Number and names of PMUs, Phasors, Analog Values and Digital Status Words
- Data format (floating point / integer)
- Phasors Representation (rectangular / polar)

Fig. 4. Real-time IEEE Std C37.118.2-2011 data frame parsing through the PMU_Data_Pack VI.

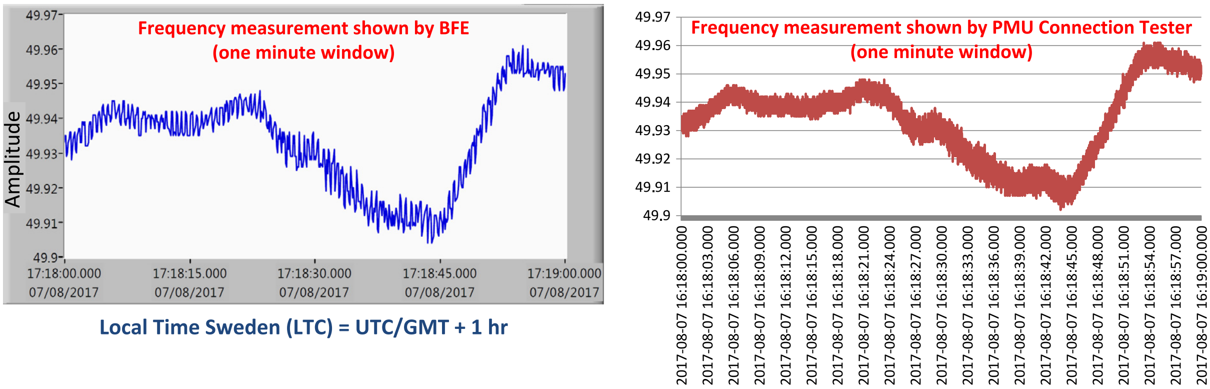


Fig. 5. Frequency measurement over a period of one minute as shown by BFE (left) and PMU connection tester (right).

(as shown in Fig. 1). The network packets exchanged between BFE and PMU/PDC as captured and decoded by Wireshark were analyzed to validate that the messages exchanged are in accordance with IEEE Std C37.118.2-2011.

In order to verify that the BFE is correctly parsing the IEEE Std C37.118.2-2011 frames, "PMU Connection Tester" [16] which is a de-facto standard testing tool in synchrophasor technology, is utilized. For this purpose, two identical output synchrophasor streams are configured, namely "PDC1" and "PDC2" and received in the same workstation. One stream "PDC1"is received using BFE, while the other stream "PDC2" is received using PMU Connection Tester. The important fields of the configuration frame "CFG-2" as parsed by BFE and PMU Connection Tester were compared, which further validated the IEEE Std C37.118.2-2011 compliancy of BFE.[2]

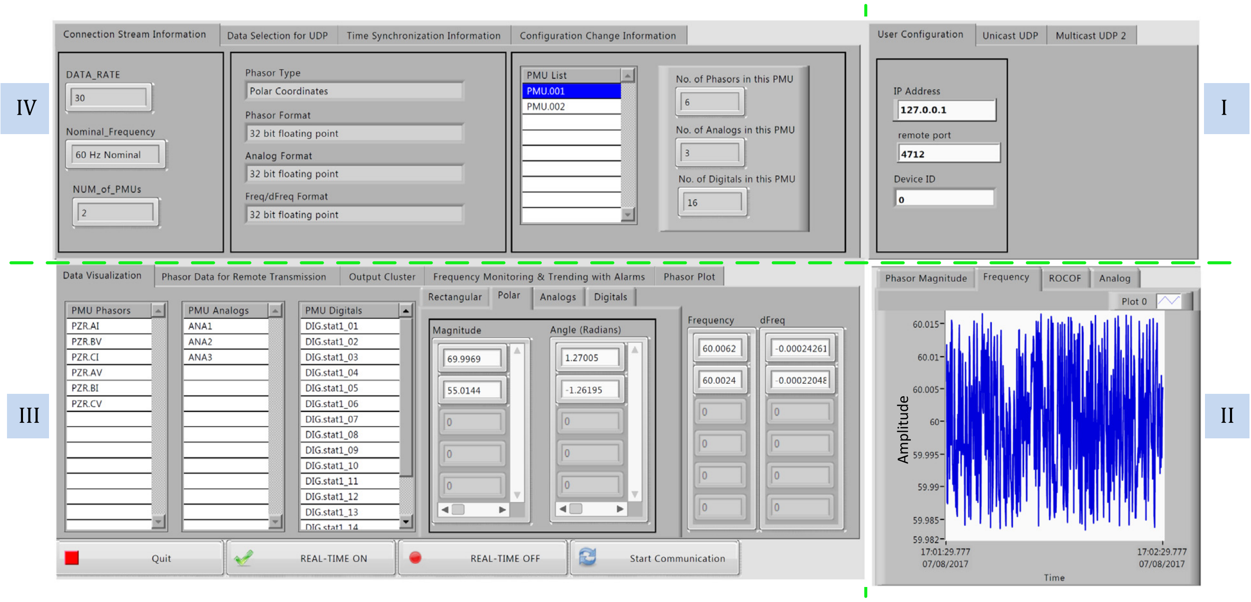
The frequency measurement of the same PMU available in PDC1 and PDC2, as displayed by BFE and PMU Connection Tester is shown in Fig. 5. It is worth noting that the one hour difference in time (x-axis) between BFE and PMU Connection Tester is because BFE utilizes Local Coordinate Time (LTC) while PMU Connection Tester uses UTC. These tests confirm that BFE functions according to the IEEE Std C37.118.2-2011 specifications.

3. Illustrative example

3.1. Graphical User Interface (GUI)

The main GUI of the BFE is primarily the PDC VI shown in Fig. 6. The GUI is divided into 4-quadrants (marked I–IV). The upper right quadrant (I) allows the user to configure the IP Address, Port number and Device ID of the PMU/PDC to establish a TCP/IP connection. In order to establish a TCP/IP connection [6], the user should press the "Start Communication" (on the bottom of the GUI in quadrant III) option to establish a TCP/IP socket with the required configured parameters. The upper left quadrant (IV) of the GUI provides the meta-data corresponding to the synchrophasor stream, which includes the data rate and nominal frequency of the incoming stream, number of PMUs in the stream, data format for the measurements, names of PMUs in the stream and number of different Phasors, Analogs and Digitals, in each PMU. Additionally, the time synchronization and configuration change information extracted from the PMU/PDC stream can be accessed

[2] The readers/users are requested to refer to the supplementary material (Appendix A) submitted with this manuscript for complete documentation on "BFE Software Testing and Validation".



DATA_RATE: 30

Nominal_Frequency: 60 Hz Nominal

NUM_of_PMUs: 2

Phasor Type: Polar Coordinates

Phasor Format: 32 bit floating point

Analog Format: 32 bit floating point

Freq/dFreq Format: 32 bit floating point

PMU List: PMU.001, PMU.002

No. of Phasors in this PMU: 6

No. of Analogs in this PMU: 3

No. of Digitals in this PMU: 16

IP Address: 127.0.0.1

remote port: 4712

Device ID: 0

PMU Phasors: PZR.AI, PZR.BV, PZR.CI, PZR.AV, PZR.BI, PZR.CV

PMU Analogs: ANA1, ANA2, ANA3

PMU Digitals: DIG.stat1_01, DIG.stat1_02, DIG.stat1_03, DIG.stat1_04, DIG.stat1_05, DIG.stat1_06, DIG.stat1_07, DIG.stat1_08, DIG.stat1_09, DIG.stat1_10, DIG.stat1_11, DIG.stat1_12, DIG.stat1_13, DIG.stat1_14

Magnitude	Angle (Radians)	Frequency	dFreq
69.9969	1.27005	60.0062	-0.00024261
55.0144	-1.26195	60.0024	-0.0002204E
0	0	0	0
0	0	0	0
0	0	0	0
		0	0

Fig. 6. GUI of BFE showing a list of different PMUs in the stream and their respective measurements and plots.

by navigating through respective tabs in this quadrant (IV). The user can select the "data-of-interest" and turn on its transmission to the remote application using "Data Selection for UDP" tab. The user can navigate through the PMU list to identify the number of the measurements available in each of the PMU. The lower left quadrant (III) allows the user to enable or disable synchrophasor data transmission; and displays phasors (both in rectangular and polar coordinates), analog, digitals, frequency and ROCOF measurements. Furthermore, the user can visualize the polar plot of the available phasors and access the frequency monitoring application by navigating through respective tabs in this quadrant (III). The lower right quadrant of the GUI (II) allows the user to visualize these measurements in the plots.

The GUI of BFE is shown in Fig. 6. A simple three step process is used to connect to and visualize this PDC stream.

1. Configure the IP Address, port number and Device ID of the PDC stream in the upper left quadrant.
2. Click on "Start Communication" to establish the TCP/IP socket and exchange command messages with the PDC.
3. Click on "Real-Time on" to start receiving and parsing the IEEE Std C37.118.2-2011 data frames for their visualization in the form of measurement values and plots.

3.2. Example application: Frequency monitoring with alarms

As an illustrative example, a frequency monitoring application is included in the BFE which can be accessed from the GUI of BFE (quadrant III in Fig. 6). This application monitors frequency of a system and activates warnings/alarms when the frequency goes beyond thresholds. The code snippet of this application is shown in Fig. 7. The GUI of this application is shown in Fig. 8, where the top figure (a) shows system with normal frequency i.e. within the threshold limits of 50.05 Hz and 49.95 Hz. Therefore no alarm is activated in Fig. 8(a). In Fig. 8(b), the frequency of the system is below 49.95 Hz. Therefore, an alarm corresponding to Frequency < Threshold is activated and the operator is prompted with a warning /corrective action on the GUI.[3]

3.3. Polar plots of phasors in BFE

The feature of making polar plots of the available phasors in the PMU/PDC stream is made available in the BFE as a sub-VI i.e. the user can simply drag & drop this sub-VI block to make phasor plots. The GUI of the BFE with the phasor plots of a PDC stream are shown in Fig. 9. In Fig. 9, (a) shows the names of all the available phasors in the PDC stream extracted from the "CFG-2" frame, while the phasor magnitude and phasor angle corresponding to the phasor names (a) are shown in (b) and (c), respectively. The phasor plot is shown in Fig. 9(d).

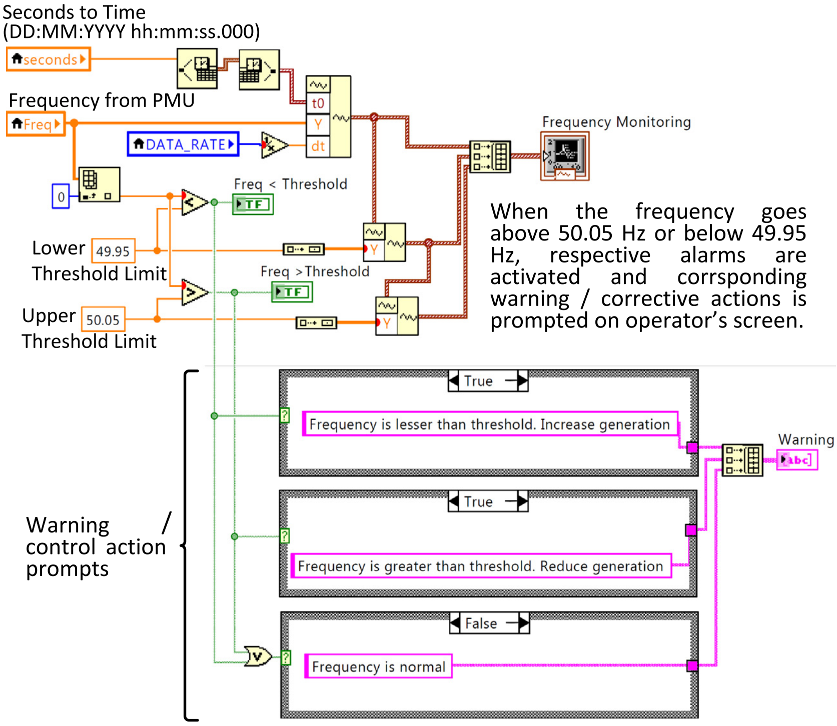
4. Impact

The BFE has significant utilization potential primarily in two user groups: (i) universities/research institutes and (ii) companies in the electrical power industry (e.g. vendors, service providers). In academia, the direct user would be the students and researchers with a background in power systems and/or signal processing, but lacking proficient software development and programming skills. The final beneficiaries are the companies working in the power system operation business such as generation, transmission and distribution system operators. BFE would allow these companies to develop or commission a prototype application.

The BFE software tool was utilized to develop a monitoring tool capable of detecting and providing alarms for subsynchronous wind farm oscillations in the transmission networks due to high penetration of the wind based electric power generation. The developed monitoring tool was validated by using real-measurements from Oklahoma Gas and Electric (OG&E) [17] through hardware-in-the-loop real-time simulation testing in KTH SmarTS-Lab [18] and with hardware-based emulation at IREC's microgrid laboratory [19]. In addition, the application's functionality was also appraised against OG&Es own in-house software in [20].

Numerous WAMPAC applications previously developed in KTH SmarTS-Lab utilize another in-house developed IEEE Std C37.118.2-2011 protocol parser named S3DK [21]. However, in principle all of these WAMPAC applications can be developed using BabelFish. From the wide-area monitoring perspective, these applications include (i) real-time monitoring display of the synchrophasor data [10], (ii) mode meter to determine

[3] These screenshots are taken with real PMU measurements received form the Nordic Power System. This particular PMU is installed in Lund which is in the South of Sweden.



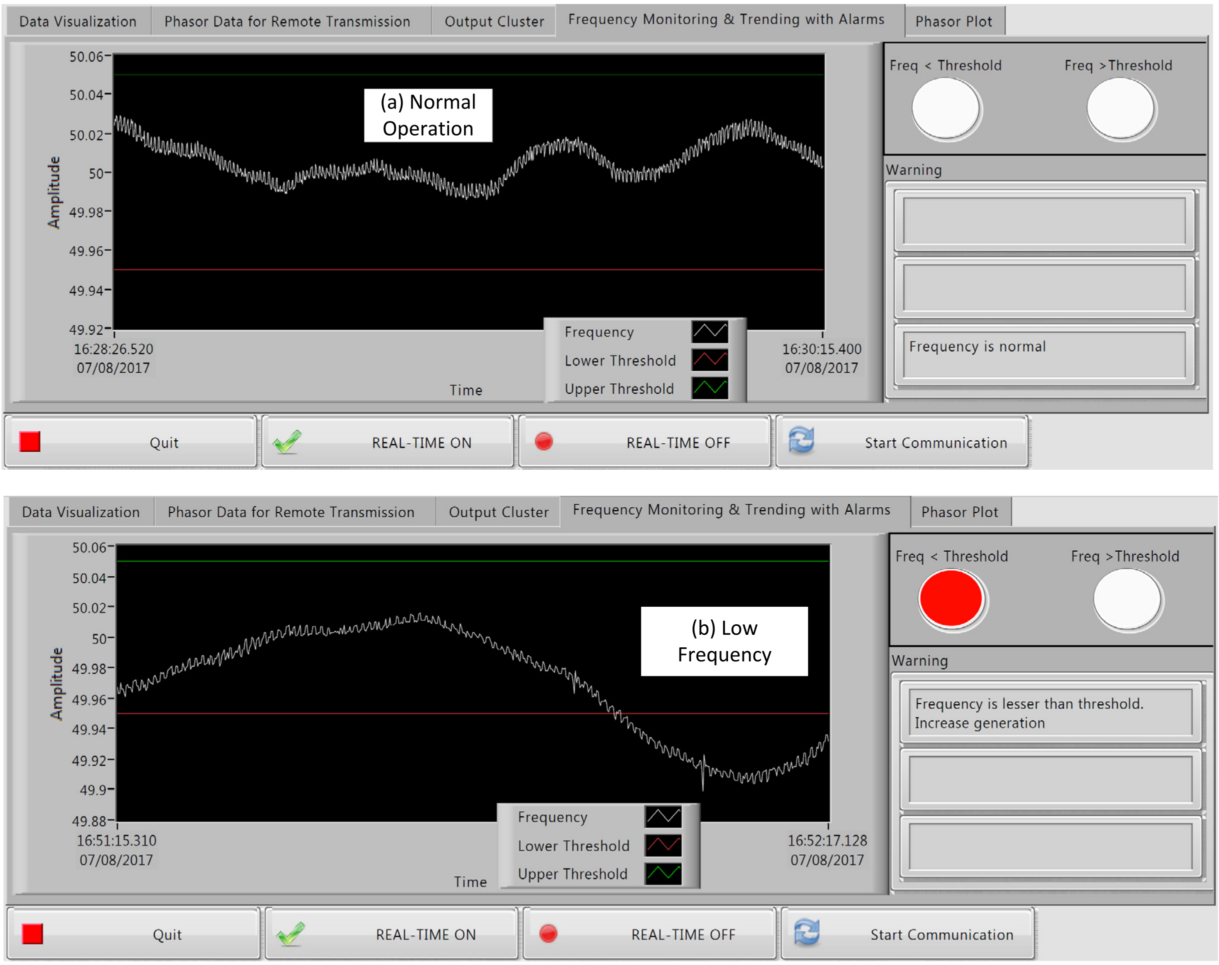
When the frequency goes above 50.05 Hz or below 49.95 Hz, respective alarms are activated and corrsponding warning / corrective actions is prompted on operator's screen.

Frequency is lesser than threshold. Increase generation

Frequency is greater than threshold. Reduce generation

Frequency is normal

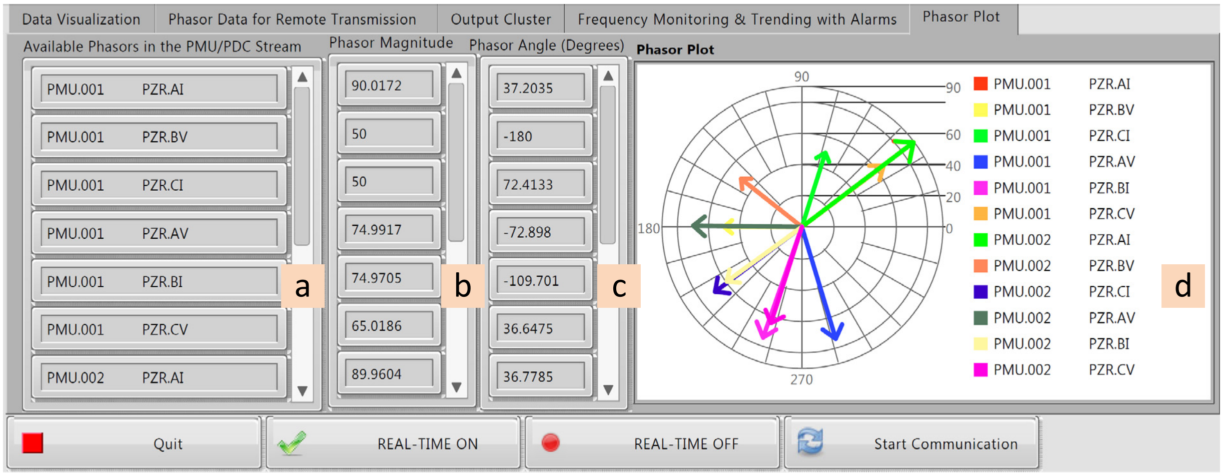
Fig. 7. Code snippet for Frequency monitoring and trending application with alarms and warning features.



Frequency is normal

Frequency is lesser than threshold.
Increase generation

Fig. 8. GUI of Frequency monitoring and trending application available in BFE.



Available Phasors in the PMU/PDC Stream		Phasor Magnitude	Phasor Angle (Degrees)
PMU.001	PZR.AI	90.0172	37.2035
PMU.001	PZR.BV	50	-180
PMU.001	PZR.CI	50	72.4133
PMU.001	PZR.AV	74.9917	-72.898
PMU.001	PZR.BI	74.9705	-109.701
PMU.001	PZR.CV	65.0186	36.6475
PMU.002	PZR.AI	89.9604	36.7785

🟥	PMU.001	PZR.AI
🟨	PMU.001	PZR.BV
🟩	PMU.001	PZR.CI
🟦	PMU.001	PZR.AV
🟪	PMU.001	PZR.BI
🟧	PMU.001	PZR.CV
🟩	PMU.002	PZR.AI
🟧	PMU.002	PZR.BV
🟦	PMU.002	PZR.CI
🟩	PMU.002	PZR.AV
🟨	PMU.002	PZR.BI
🟪	PMU.002	PZR.CV

Fig. 9. GUI of BFE showing a phasor plot of the PDC stream containing 2 PMUs with 6 phasors each (i.e. total of 12 phasors).

poorly damped modes through measurement-driven models [22], (iii) synchrophasor monitoring application for smart phones and tablets by exploiting different LabVIEW toolkits [23]. For wide-area protection applications, BFE provides an alternative for developing applications such as (i) anti-islanding protection [24], (ii) automatic synchronization, and (iii) auto-recloser. BFE can also be utilized for fast prototyping of wide-area controllers such as (i) phasor-based power oscillation damping [25], and (ii) load control for power system stability [26].

5. Conclusions

BabelFish Tools provide a repository with software data mediators (gateways) to translate synchrophasor data compliant to IEEE Std C37.118.2-2011 to raw numerical values and associated metadata. Two mediators are made available, BabelFish V1, that uses C++, Active X and LabVIEW; and more importantly BFE, developed entirely in LabVIEW. Both BabelFish V1 and BFE provide the following functionalities: (i) real-time IEEE Std C37.118.2-2011 frame parsing, (ii) selection of data-of-interest, and (iii) transmission of the chosen data over UDP to any remote or local destination. Hence the final user has the liberty to receive "data-of-interest" from any PMU/PDC and develop applications utilizing this data independent of platform, language, OS, and geographical location, to the users' best interests.

The BabelFish Tools are open source software and distributed under the GPLv3 License [27].

Acknowledgments

This work was supported by KIC InnoEnergy through their R&D project SmartPower and within the WP 2.6 "PMU-Based Power System Operation Tools" [28].

M. Shoaib Almas was supported by Nordic Energy Research through the STRONg2rid project. L. Vanfretti was supported by the STandUP for Energy collaboration initiative and Statnett SF, the Norwegian Transmission System Operator. M. Baudette was supported by Statnett SF, the Norwegian Transmission System Operator.

The authors would like to thank Dr. Iyad Al-Khatib for his contributions in the development of DLLs for BabelFish v1 and some modules of BFE.

Appendix. Supplementary data

Supplementary material related to this article can be found online at http://dx.doi.org/10.1016/j.softx.2017.08.002.

References

[1] Laverty DM, et al. The OpenPMU platform for open-source phasor measurements. IEEE Trans Instrum Meas 2013;62(4):701–9.
[2] DeLaRee J, Centeno V, Thorp JS, Phadke AG. Synchronized phasor measurement applications in power systems. IEEE Trans Smart Grid 2010;1(1):20–7.
[3] IEEE C37.118.1-2011. IEEE standard for synchrophasor measurements for power systems. IEEE Power and Energy Society; 2011.
[4] IEEE C37.118.2-2011. IEEE standard for synchrophasor data transfer for power systems. IEEE Power and Energy Society; 2011.
[5] Microsoft. The OSI model's seven layers defined and functions explained. Available online: https://support.microsoft.com/en-us/kb/103884.
[6] Internet Engineering Taskforce (IETF). TCP/IP tutorial — RFC1180. Available online: https://tools.ietf.org/html/rfc1180.
[7] Grid Protection Alliance (GPA). Phasor data concentrator — OpenPDC. Available online: http://openpdc.codeplex.com/.
[8] GPA's Grid Solution Framework (GSF). Time-Series Library (TSL) components. Documentation available online: https://gsf.codeplex.com/.
[9] National Instruments. LabVIEW-system design software. Documentation available online: http://www.ni.com/labview/.
[10] Almas MS, Baudette M, Vanfretti L, Løvlund S, Gjerde JO. Synchrophasor network, laboratory and software applications developed in the STRONg2rid project. In: IEEE PES general meeting, Washington DC, USA, July 2014.
[11] Vanfretti L, Al-Khatib I, Almas MS. Real-time data mediation for synchrophasor application development compliant with IEEE C37.118.2. In: Innovative smart grid conference (ISGT), North America, Washington DC, USA, Feb 2015.
[12] National Instruments. The CompactRIO platform. Documentation available online: http://www.ni.com/compactrio/.
[13] IEEE C37.244-2013. IEEE guide for phasor data concentrator requirements for power system protection, control, and monitoring. IEEE Power and Energy Society; 2013.
[14] National Instruments. Virtual Instrumentation (VI). Documentation available online: http://www.ni.com/white-paper/4752/en/.
[15] Wireshark. Network protocol analyzer. Available online: https://www.wireshark.org/.
[16] Grid Protection Alliance (GPA). PMU connection tester. Available online: https://pmuconnectiontester.codeplex.com/.
[17] Baudette M, et al. Validating a real-time PMU-based application for monitoring of sub-synchronous wind farm oscillations. In: Innovative smart grid technologies conference (ISGT), 2014 IEEE PES, Washington, DC, 2014, p. 1–5.
[18] Vanfretti L, Baudette M, Al-Khatib I, Almas MS, Gjerde JO. Testing and validation of a fast real-time oscillation detection PMU-based application for wind-farm monitoring. In: First international black sea conference on communications and networking (BlackSeaCom), Batumi, 2013, p. 216–21.
[19] Vanfretti L, et al. A PMU-based fast real-time oscillation detection application for monitoring wind farm-to-grid sub-synchronous dynamics. Electr Power Compon Syst 2016;44(2):123–34.
[20] Vanfretti L, Baudette M, White A. PMU and WAMS application to renewables integration. In: Jones Lawrence, editor. Renewable energy integration: Practical management of variability, uncertainty, and flexibility in power grids. Elsevier; 2014, Chapter 33.

[21] Vanfretti L, Aarstrand VH, Almas MS, Perić VS, Gjerde JO. A software development toolkit for real-time synchrophasor applications. In: IEEE powertech 2013, Grenoble, France, June 2013.

[22] Perić VS, Baudette M, Vanfretti L, Gjerde JO, Løvlund S. Implementation and testing of a real-time mode estimation algorithm using ambient PMU data. In: Power system conference 2014, Clemson, SC, USA, 11–14 March 2014.

[23] National Instruments. Case study – smart grid measurements from the control room to your hands. Available online: http://sine.ni.com/cs/app/doc/p/id/cs-16049#.

[24] Almas MS, Vanfretti L. RT-HIL implementation of hybrid synchrophasor and GOOSE-based passive islanding schemes. IEEE Trans Power Deliv 2015;31(3): 1299–309.

[25] Rebello E, Vanfretti L, Almas MS. PMU-based real-time damping control system software and hardware architecture synthesis and evaluation. In: IEEE PES GM 2015, July 26–30, Denver, Colorado, USA.

[26] Jonsdottir GM, Almas MS, Baudette M, Vanfretti L, Palsson MP. Hardware prototyping of synchrophasor and active load-based oscillation damping controllers using RT-HIL approach. In: IEEE PES GM 2016, July 17–21, Boston, Massachusetts, USA.

[27] GNU Operating System. General Public License — GPLv3. Documentation. Available online: http://www.gnu.org/licenses/gpl-3.0.en.html.

[28] KIC InnoEnergy. Smart grids from power producers to consumers-smart power. Documentation available online: http://www.kic-innoenergy.com/.