The OpenPMU Platform for Open-Source Phasor Measurements

David M. Laverty, *Member, IEEE*, Robert J. Best, Paul Brogan, Iyad Al Khatib, *Member, IEEE*,
Luigi Vanfretti, *Member, IEEE*, and D. John Morrow, *Member, IEEE*

***Abstract*—OpenPMU is an open platform for the development of phasor measurement unit (PMU) technology. A need has been identified for an open-source alternative to commercial PMU devices tailored to the needs of the university researcher and for enabling the development of new synchrophasor instruments from this foundation. OpenPMU achieves this through open-source hardware design specifications and software source code, allowing duplicates of the OpenPMU to be fabricated under open-source licenses. This paper presents the OpenPMU device based on the Labview development environment. The device is performance tested according to the IEEE C37.118.1 standard. Compatibility with the IEEE C37.118.2 messaging format is achieved through middleware which is readily adaptable to other PMU projects or applications. Improvements have been made to the original design to increase its flexibility. A new modularized architecture for the OpenPMU is presented using an open messaging format which the authors propose is adopted as a platform for PMU research.**

***Index Terms*—Open source, phasor measurement units, power systems, synchrophasors, time synchronization.**

I. INTRODUCTION

SYNCHROPHASOR measurement technology is gaining in popularity throughout the power system environment. A synchrophasor is a time-synchronized phasor, which is a representation of the amplitude and phase angle of an electrical quantity.

The marketplace offers a variety of phasor measurement unit (PMU) devices that assert compliance with the IEEE Standard for Synchrophasor Measurements for Power Systems C37.118.1-2011 [1]. The standard describes certification requirements which has yielded a certain degree of uniformity among devices from different vendors, yet known issues such as the transient, or dynamic, response of the devices remain [1]–[4].

A number of university research departments have identified limitations in the closed source, or "black box," philosophy under which commercial PMU devices are developed. This is seen as an obstacle to developing innovations in the field. An alternative, open model for the development of PMU technology is the subject of this paper.

OpenPMU is an open platform including a PMU device developed with the needs of university research departments specifically in mind. The complete design specifications, source code, and binaries associated with this project are made available under open licenses. OpenPMU has been developed largely using the Labview platform from National Instruments, which is commonly licensed by universities. OpenPMU has been re-structured into modules which decouple the various components of a PMU by means of defined information exchange structures. This modular approach allows advanced phase estimation algorithms, such as [5], [6], or measurement techniques such as [7] to be implemented on open hardware and subject to the same test procedures as commercial PMUs. Thus, the proposed algorithms can be compared in a practical setting.

This paper will set the historical context in which PMU technology has been developed. Similar projects to the OpenPMU are reviewed, and the rational for developing PMU technology in an open-source manner is set. The development of the present Labview-based OpenPMU is discussed. The performance of this unit is assessed against the IEEE standard and compared against a commercial device. Compatibility with the IEEE C37.118.2 [8] communications specification is achieved by middleware which was developed for this purpose. The new modularized architecture of the OpenPMU is presented and discussed as a platform for the ongoing open development of PMU technology.

II. HISTORICAL CONTEXT

Phasor representation of waveforms to solve ac circuit problems was first performed by the German–Austrian electrical engineer and mathematician Charles Proteus Steinmetz in a paper he presented to the International Electrical Congress in 1893 [9]. The method gradually became adopted for the analysis of early power networks and is widely used today in the design and operation of modern power systems.

A phasor is a mathematical representation that describes the amplitude and phase angle of an electrical waveform (voltage or current). PMU are devices that estimate phasor values from electrical waveforms. In order to make comparison of phasors

---

Manuscript received April 26, 2012; revised September 24, 2012; accepted October 1, 2012. Date of publication February 15, 2013; date of current version March 8, 2013. This work was supported by a Charles Parsons Energy Research Award, which is funded by the Science Foundation Ireland, award (06/CP/E002). Luigi Vanfretti is supported in part by the STandUP for Energy collaboration initiative, the KTH School of Electrical Engineering, and through the STRONg2rid project funded by Nordic Energy Research. The Associate Editor coordinating the review process for this paper was Dr. Carlo Muscas.

D. M. Laverty, R. J. Best, P. Brogan, and D. J. Morrow are with the Electrical Power and Energy Research Cluster, Queen's University Belfast, Belfast, BT9 5AH, Northern Ireland (e-mail: david@laverty.org.uk; r.best@qub.ac.uk; pbrogan02@qub.ac.uk; dj.morrow@ee.qub.ac.uk).

I. Al Khatib and L. Vanfretti are with the KTH Royal Institute of Technology, 100 44 Stockholm, Sweden (e-mail: Iyad.Al-Khatib@ics.kth.se; luigiv@kth.se).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIM.2013.2240920

from PMUs installed at different geographical locations useful, it is necessary to synchronize all measurements to a common time base. This is typically achieved using the GPS satellite navigation system. Synchronized phasor measurements are known as synchrophasors.

The measurement of voltage phase angles using synchronized clocks for power system applications dates back to the early 1980s when measurements of voltage phase angles were carried out between Montreal and SEPT-ILES [10], [11], and parallel efforts by Bonanomi in 1981 [12].

However, the synchrophasor technology available today emerged from the early efforts by Phadke *et al.* at Virginia Tech as described in [13], [14]. Phadke demonstrated the first synchronized PMU in 1988, and in 1991 Macrodyne Inc. launched the first commercial PMU product [15]. Due to the cost of early PMU devices, PMU technology has historically been limited to transmission system applications where the business case justified expensive phasor analysis equipment. One of the early applications that is important to mention is the implementation of the wide-area protection system "Syclopes" in France in the early 1990s, which was the first functional application of early forms of PMUs [16].

Recent developments across the electronics sector have seen the cost of the components from which PMUs are assembled (such as GPS receivers, microprocessors, and storage devices) drop dramatically in price. Consequently, PMUs have reached price points that have made them an attractive tool across the utility environment, including distribution systems and embedded generation [17]–[20].

Many PMUs are sold as dedicated devices which offer event recorder type functionality. Costs for such units vary between US$6000 and US $15 000 depending on the specification. Many equipment vendors have begun to offer PMU functionality as a supplementary feature on other products in their range, such as protection relays [21].

The standard for PMU devices is maintained by the IEEE C37.118 Working Group. IEEE Std. C37.118 [1] was released in 2005 and subsequently updated in 2011. The latest release comes in two parts; IEEE C37.118.1-2011 [1] describes how synchrophasors should be estimated and gives certification requirements while IEEE C37.118.2-2011 [8] describes data representation and data transfer. Concerns have been raised regarding the transient performance of PMUs under the 2005 standard [1], [20], [22]. These concerns are addressed in the 2011 release of the standard. IEEE C37.118.1-2011 states that it defines "synchrophasors, frequency, and rate-of-change-of-frequency measurement under all operating conditions [1]."

III. SIMILAR PMU PROJECTS

A significant barrier regarding the use of PMU technology in research is the closed philosophy under which commercial PMU devices are developed and sold. Commercial vendors tightly guard their hardware and software designs, meaning that the measurement processes and algorithms are not known to researchers. This has led to some research departments developing their own PMU systems. Many designs utilize low-cost hardware, such as described in [7]. Two university projects are described in this section. Duplication of such work leads to lost time and resources. The OpenPMU project provides a common set of resources for PMU development and research collaboration. The successful open-source Phasor Data Concentrator, openPDC [23], is discussed, and the rational for using an open-source model is developed.

A. GridTRAK PMU

The GridTrak PMU was produced at Baltimore University by Stadlin [24]; subsequently, the design has been published under open-source license. The aim of GridTrak is to produce an inexpensive PMU that can be widely distributed, among researchers and amateur enthusiasts, allowing widespread monitoring of the distribution network. The design works via a zero crossings technique, making the unit simple and robust; however, the loss of point-on-wave information reduces GridTrak's applications.

The GridTrak hardware converts the ac measurement signal into three square waves triggered at the crossing of reference voltages. Frequency estimation is determined by the interval between the crossings while voltage is estimated by imposing a perfect sine wave on the full set of crossing points and determining the magnitude. The GridTrak incorporates a GPS module from which it derives time and estimates phase angle. This design is limited to single phase measurements, and all point-on-wave and harmonic information is lost.

B. DTU PMU

The DTU PMU [25] was produced in several stages at the Technical University of Denmark. The DTU PMU utilizes two PCs to monitor the ac voltage signal, actively synchronizing the sample rate to 64 or 128 samples per cycle, to fit the waveform [26], [27]. The first PC runs MS-DOS in a near real-time state, stripping out background programs that might interrupt measurements. These measurements are packeted and exported to the second PC at intervals of 20 ms. The second PC runs Labview; in this environment waveform parameters are estimated and the information is archived locally as well as exported in IEEE C37.118 format to a central location.

The PMU was thoroughly tested in house before ten models were installed across the Danish electricity transmission and distribution grid including wind farms and consumer supply [28]. Through ambient monitoring, this wide-area monitoring system has successfully detected many transient system events as-well-as identifying a 0.8 Hz inter-area oscillation, believed to arise from rotor interaction between generators in Sweden and Eastern Denmark.

C. The openPDC

The openPDC was developed in the wake of the Northeast Blackout of 2003 [29]. Following the blackout, many grid improvements were recommended including increased real time observability. The Tennessee Valley Authority began developing the Super PDC in 2004 to monitor and archive its installed PMUs. In 2010, it released the code under an open-source license, and the Grid Protection Alliance took on responsibility of developing the program and entered into a contract with the North American Electric Reliability Council. The history and development of openPDC and associated projects are now recorded by the openPDC online community [23]. The openPDC is utilized by the North American Synchrophasor Initiative.

The openPDC runs as a Windows Service programmed in the Microsoft Visual Basic Studio (Linux versions are also available). It exists as a modular set of programs that can be combined in different forms to achieve different results. Modules, or Adapters as they are called, are activated through Structured Query Language (SQL) commands in the assigned database (DB) and can be reprogrammed through Visual Basic. The openPDC system primarily operates between real-world telecommunications infrastructure and an archive DB. The adapters within openPDC can be subdivided into three groups: Input, Action, and Output adapters. The input adapters receive the raw telecommunications information (in any of the major synchrophasor communication standards including C37.118), process it to extract the relevant data, and then send it for processing or archival in the SQL DB. Action Adapters can process in real time or post event as well as fulfilling the concentrate/compress functions in the DB. Furthermore, Action Adapters can introduce new measurements, for example by importing Comma Separated Value (CSV) files into an existing archive DB. Output adapters can be used to forward data in a chosen communication language such as to emulate a physical PMU. In this way the openPDC can operate a diverse variety of user-specific configurations.

The success of openPDC demonstrates that open-source software licensing can be successfully applied in relation to synchrophasor technology.

D. Rationale for Open Development Platform

The rationale for developing a PMU device in an open manner under open-source and open project licenses is so that the other interested parties can study the operation of the device, build duplicates, and contribute modifications to the project's community.

"Open source" is a general term for a number of philosophies regarding the development, distribution, and use of computer source code. Unlike "freeware," open-source computer programs are provided with their source code, usually attached to a license that allows the end user to modify the source code provided the license is extended to distribution of the new program. Some open-source licenses require modifications to be returned to the original authors, while others require only that the original authors are acknowledged. Although enforcement of licenses may be problematic, many open-source projects thrive due to the motivation of their members.

Parts of the OpenPMU are presently dependent on the Labview environment, which is not open source. The Labview project files are distributed under open-source license, but this might be regarded as an "open project." It is intended to gradually move to the Python language which will allow the code to be fully open sourced. Other software components of OpenPMU are presently fully open.

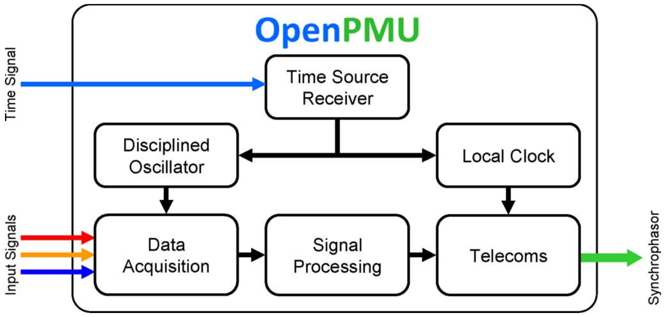


Fig. 1. Schematic diagram of the OpenPMU.

The OpenPMU includes hardware as well as software. Source code may be protected by copyright, whereas hardware is usually protected by patent. To overcome this issue, the design specifications and drawings for the hardware are copyrighted and distributed under an open-source license. The licenses used by the various sections of OpenPMU are published on the project website.

The authors believe that the community of researchers interested in PMU technology is now sufficiently established to be able to sustain an open-source project, and what's more, encourage its growth and development. OpenPMU is not intended as competition to commercial device manufacturers, rather it is desired to be a tool to compliment commercial devices in the research environment. The project's current contributors, along with all source code and design drawings and instructions are available online [30].

IV. LABVIEW OPENPMU

The OpenPMU system was originally designed and developed at Queen's University Belfast (QUB), United Kingdom, to meet the needs of specific projects investigating anti-islanding detection and phase control of islanded generators [19], [31], [32]. The original motivation was to produce a low-cost PMU optimized for real-time applications. The complete unit can be assembled for approximately US$1000.

The unit discussed in this section is the design that is available on the OpenPMU website. It is of the design used in QUB research projects with functions and intellectual property related to specific projects removed. The unit relies heavily on the Labview environment, an issue that is addressed later in discussion of the modularized OpenPMU.

A. Data Acquisition Hardware

At the schematic level, Fig. 1, the OpenPMU shares much in common with the PMU design described by Phadke in [13]. At the heart of the system is a National Instruments NI-DAQ data acquisition device. The sampling trigger of the NI-DAQ is disciplined to the GPS time signal via a PIC microcontroller operating firmware of the authors' design.

The National Instruments USB-6009-OEM is a budget 'barebones' device. All input/output is provided through a single connection. The analog-to-digital converter (ADC) has a resolution of 14 bits and a sampling rate of 48 kHz. The ADC can sample eight channels via a multiplexer, and acquisition can be triggered by software or a hardware interrupt. In the OpenPMU, the ADC is configured to sample 6 channels (three voltage, three current) at 6.4 kHz, that is 128 samples per cycle at 50-Hz nominal frequency. The device can also achieve 128 samples per cycle at 60 Hz. The ADC is configured to start acquisition on the rising edge of the external hardware sampling trigger.

The sampling trigger signal is a 50-Hz/60-Hz square wave which by means of a GPS receiver oscillates in phase with Coordinated Universal Time (UTC). The GPS receiver used is a Garmin GPS-18x which outputs a TTL level one-pulse-per-second (1PPS) signal, a 1-Hz square wave disciplined, and in phase with the transition of the UTC second. Firmware on a PIC microcontroller operated as a fuzzy logic phase-locked loop which multiplies the 1PPS by the synchrophasor reporting rates recommended in IEEE C37.118.1 [1]. The recommended reporting rates are 10/25/50 reports per second in a 50-Hz system, and 10/12/20/30/60 reports per second in a 60-Hz system. The PIC microcontroller also provides a time transfer mechanism to the Labview environment so that synchrophasors can be time coded with UTC-derived time.

B. Phase Estimation Software

Synchrophasors are computed on a standard Windows PC from the analog waveform data within the Labview environment. At present, signal processing is performed using National Instruments supplied signal analysis libraries. The spectrum of nominally four cycles at line frequency is found using an NI supplied fast Fourier transform (FFT). The three dominant frequency bands of the spectrum are used in an iterative curve fitting algorithm. The frequency, phase, and amplitude of the measured waveform are determined from the parameters of the synthesized waveform generated during curve fitting. The full algorithm can be visualized in schematic form within the Labview VIs on the OpenPMU website. In order to preserve open-source transparency within the project, the method will be adapted in subsequent releases to use open-source FFT code, such as that from the open-source FFTW project [33]. It is desired that algorithms found in literature, such as [5], [6], will be implemented on the platform. The NI supplied implementation of the curve fitting algorithm has been modified such that the phase angle is estimated with respect to the nominal system frequency (50 Hz/60 Hz), as opposed to the detected frequency, as specified in IEEE C37.118.1-2011 [1]. Thus amplitude, phase and frequency are estimated.

Much of the Labview software is concerned with configuration of the data acquisition card and correctly time coding the incoming waveform sample data. The incoming data is buffered in a queue and phase estimation performed asynchronously. In normal operation the queue will be empty, but in the event the host PC's CPU is busy with other duties, then data will be buffered rather than lost.

Estimated synchrophasor parameters are exported from Labview in human readable CSV format, which contains the time code, amplitude, frequency, and phase angle in ASCII strings representing decimal numbers. The format resembles that of National Marine Electronics Association sentences [34] which is a popular standard in marine communications also used by GPS receivers. The CSV string is transmitted by Labview by means of User Datagram Protocol over Internet Protocol (UDP/IP) to a destination IP address and port number. This method has its origins in the real-time control applications from which the OpenPMU originated [17], [19]. Middleware has been developed and tested using the 'PMU Connection Tester' from the openPDC project [23] which achieves compatibility with IEEE C37.118.2 [8]. This allows the OpenPMU to be used with a variety of existing data concentrator software. This middleware is discussed later.

Fig. 2. OpenPMU unit in 19″ rack form factor enclosure.

C. Physical Construction

The main components of the OpenPMU populate a single PCB that is easy to fabricate in basic workshop facilities. Two variants of the PCB exist, differentiated by the type of voltage transducer used to step distribution voltages down to the ±10 V required by the data acquisition card. The Hall Effect option has improved linearity but is expensive, whereas the wound transformer option is considerably more affordable. The user should select an option that best fits their needs. Since current transducers come in many varieties, these can be interfaced via a riser socket which supplies power to signal conditioning amplifiers on a daughter board.

The other PCB-mounted components include the PIC sampling trigger microprocessor, terminals to connect a GPS receiver, power supply conditioning, mounting post and connection to data acquisition card, and PC time transfer connection (RS232). A socket for front panel indicators outputs signals indicating that the device is operating correctly or that there is an error.

The components, including a small form factor computer, are intended to be housed in a 3U 19″ rack mount enclosure. AutoCAD diagrams for drilling appropriate mounting holes, front panel indicators, and logo are available online. A fully constructed unit is pictured in Fig. 2. The motherboard inside the enclosure is shown in Fig. 3.

D. Duplication From Open-Source Design

The OpenPMU has been successfully duplicated by both the KTH Royal Institute of Technology, Stockholm, Sweden, and Colorado State University, USA, from the design drawings and source code published on the OpenPMU website.

Fig. 3. Motherboard of OpenPMU system in 19″ rack unit. National Instruments data acquisition board is located under larger ribbon cable. Phase estimation algorithms are computed on PC (visible in lower right corner).

V. COMPLIANCE TESTING

The OpenPMU has been tested to determine its compliance with C37.118.1, including Total Vector Error (TVE), as described in [1], [35], [36]. Additionally, it has been compared against a commercial PMU which asserts compliance with IEEE C37.118-2005.

A. TVE

The TVE has been determined using an Omicron CMC156 test set with GPS synchronization module, as used in [36]. This allows tests to be programmed that activate in synchronism with the transition of the GPS unit's 1PPS. IEEE C37.118.1-2011 describes tests for compliance during steady-state conditions and, further to the 2005 version of the standard, also includes dynamic conditions. TVE is presented for steady-state conditions. Dynamic tests performed according to the 2011 standard are discussed in a later section.

TVE is a function of both the amplitude and phase angle of the estimated phasor with respect to the applied phasor. These two components are assessed independently, by fixing amplitude and varying phase, then fixing phase and varying amplitude. The OpenPMU's internal burden resistors are selected for measurements between 95 and 115 Vac, for its application in anti-islanding detection [32]. 115 Vac is also the upper limit of the Omicron test unit. The amplitude error, represented as a percentage of the applied value, is shown in Fig. 4.

TVE can be expressed as the magnitude of the vector that separates the theoretical and estimated phasors, normalized by the theoretical vector magnitude. Utilizing the Cosine Rule the equation for TVE from [1] can be rewritten with respect to phase error ($d\phi$), estimated magnitude ($\hat{X}$), and theoretical magnitude (X)

$$\text{TVE}^2 = 1 + (\hat{X}/X)^2 - 2(\hat{X}/X)\cos(d\phi). \tag{1}$$

The magnitude error was determined as $\pm 0.15\%$; therefore, $(\hat{X}/X)$ will equal, 0.9985 or 1.0015. From (1), to remain within 1% TVE the phase angle error must not exceed $\pm 0.567^\circ$. (Maximum permissible phase error is $\pm 0.573^\circ$ when $\hat{X}/X = 1$).

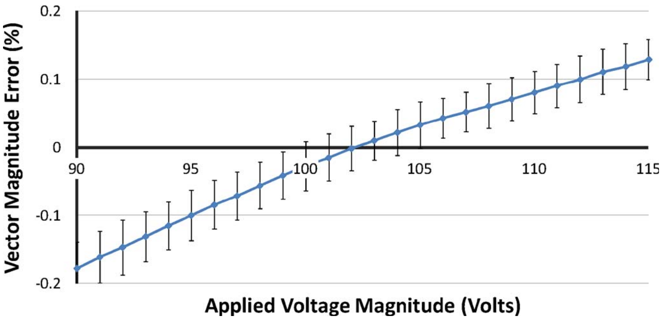


Fig. 4. Amplitude error as percentage of applied voltage.

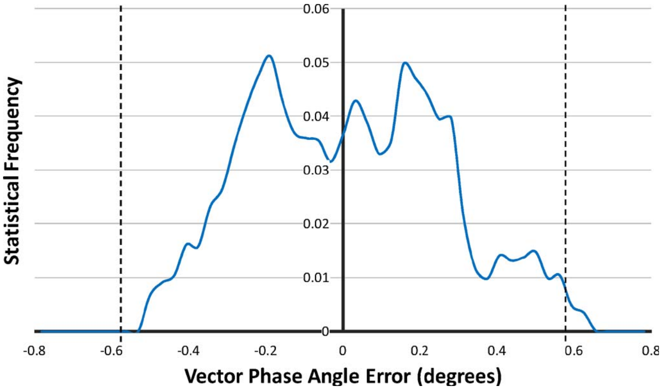


Fig. 5. PDF of phase angle error, permissible limits marked as dashed line.

The phase angle error has been measured across the range ±180°. Testing was automated to perform each measurement 20 times at 5° intervals. The observed error has been plotted on as a probability density function in Fig. 5, with the permissible limit marked in dashed line.

Fig. 5 indicates that 99% of measurements comply with the IEEE C37.118.1 standard for TVE under nominal conditions. Some measurements in these tests lie marginally outside of the permitted TVE. Future work will consider improved phase estimation algorithms with a view to achieving full compliance. However, this represents a good result for a low cost PMU with basic phase estimation algorithms.

B. Frequency Error

Frequency measurement accuracy was tested by varying the frequency between 45 Hz and 65 Hz, at 0.1 Hz intervals and amplitude 110 V. Results show a systematic error of −0.002 Hz within the system. Calculating frequency from the estimated phase angle at 60-s intervals, it was deduced that for a requested frequency of 50 Hz, the Omicron test set used is delivering 49.9998 Hz, an order of magnitude closer to desired frequency than the error seen in the estimation.

C. Dynamic Conditions

The OpenPMU was tested for dynamic compliance with the IEEE C37.118.1-2011 standard, subclauses 5.5.6 through

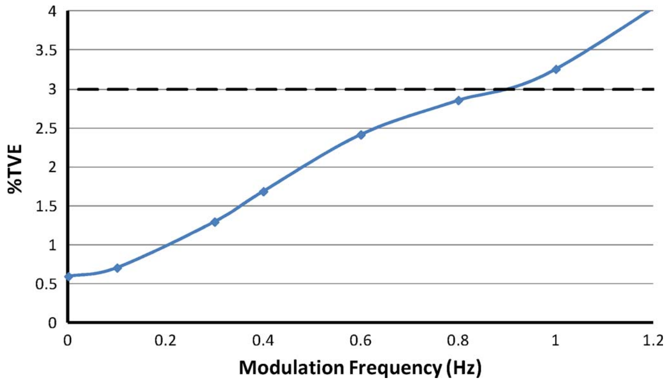


Fig. 6. TVE % versus modulation freq. under C37.118.1 subclause 5.5.6.

5.5.8. By applying the conditions specified in subclause 5.5.6, it was found that a %TVE of less than 3% was recorded for a modulation frequency of < 0.8 Hz. Fig. 6 shows the TVE during the bandwidth test for several modulation frequencies. Subclause 5.5.7 outlines frequency ramp tests, under M (measurement) and P (protection) class conditions. In these tests, a TVE of 1.5% was recorded for a ramp rate of 1 Hz/s between 45 and 55 Hz. Step changes in amplitude and phase are outlined in subclause 5.5.8; from these tests, a measurement time of 65 ms was determined. This is particularly useful in real-time control applications and has been applied in [19] to the phase control of diesel generators in synchronous islands. No overshoot or transitional states were recorded. The OpenPMU meets many of the requirements of P class [1] instruments, and with refinement of the measurement processes and estimation algorithms, it is intended to build toward a device fully compliant with P class and the more stringent M class test criteria.

The OpenPMU has been tested in parallel with a commercial PMU. A ramp of -0.1 Hz/s was applied. The estimated phasors from the two units remained within the equivalent of 0.6%TVE of each other. Similar results were found for frequency ramps in the range of ± 1.0 Hz/s.

VI. IEEE C37.118 COMMUNICATIONS

The OpenPMU was initially designed for real-time protection and control research, and so simple plain text communication using UDP was used for transmission of synchrophasors. To achieve compatibility with other PMU-related software, it is necessary to conform to the IEEE C37.118.2 standard format and defined communication model for synchrophasors. A communications module has been designed to achieve this. The module was developed to IEEE C37.118-2005 but is still compliant with IEEE C37.118.2-2011 with the exception of the new configuration frame introduced in the 2011 standard. This will be added in a subsequent release.

The design of the communication module has to take into consideration a set of critical parameters, such as: computation delay, reception and transmission speeds, functionality checks on data, and the IEEE C37.118 standard data and configuration message formats. Fig. 7 shows the dataflow to and from the communications module, which consists of three parts: an input interface to receive CSV from the Labview module, a communication engine to decide on the communication schemes, and an output interface to send data and forward commands to the communication engine.

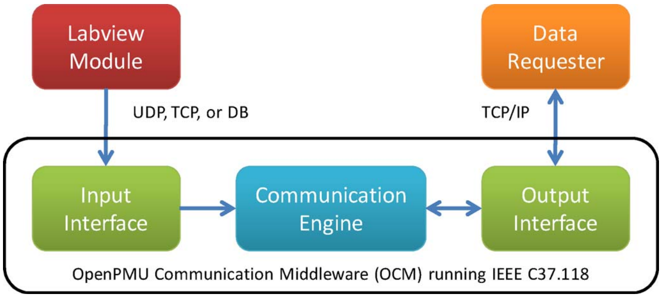


Fig. 7. Schematic for the design model of the OpenPMU Communication Middleware running IEEE C37.118.

The communication engine decides which data frame to send based on the request coming from the “Data Requester,” such as a PDC or a PMU Connection tester. The communication engine keeps checking if the input interface is receiving data from Labview in the required form, and if not (for example, empty fields, invalid time stamp), it halts its sending processes until the stream to the input interface is fixed. The design of the input interface is flexible and allows data exchange via UDP, TCP, or a DB.

This module has been programmed in Visual C++ based on code developed on Linux from [37].

VII. MODULARIZED OPENPMU

In its present form, the OpenPMU exits as a single program built within the Labview development environment, complimented by additional middleware coded in Visual C++. Information exchange between the data acquisition hardware and Labview is achieved using National Instruments proprietary mechanisms. Information between Labview and the communication middleware is exchanged in an open format, but subject to limitations due to the use of CSV formatting.

The openPDC [23] has gained momentum as a successful open-source tool for archival and retrieval of time series synchrophasor data. What immediately differentiates the architectures of openPDC and OpenPMU is that openPDC uses modules with defined ’adapter’ interfaces.

A. *Modules for OpenPMU*

The openPDC project has demonstrated that an architecture of modules and adapters can lead to a highly adaptable, configurable and extensible system. Unlike openPDC, which is entirely involved with moving data between a DB and communications, OpenPMU needs to be concerned with communication with physical hardware. In openPDC, modules and adaptors communicate via SQL DB commands. A DB is less suitable for OpenPMU in which data is time sensitive; there are continuous streams of data, and it is not necessary to store all of the data.

OpenPMU can be broken down into core modules and ancillary modules. Core modules are those of Measurement,

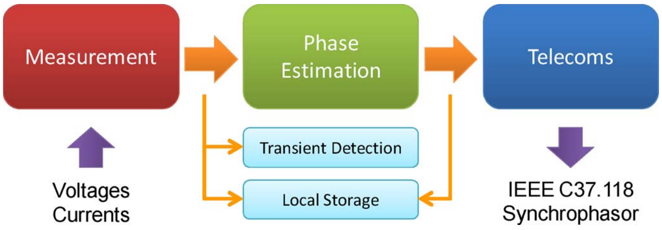
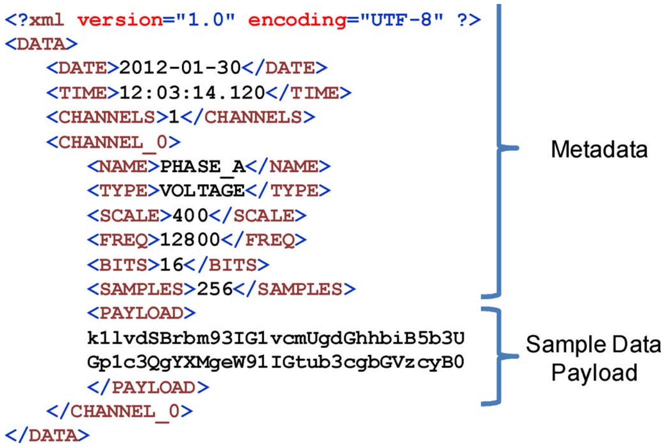


Fig. 8. Modules identified for the OpenPMU. Measurement, Phase Estimation, and Telecoms are core modules, while additional ancillary modules are optional.



```
<?xml version="1.0" encoding="UTF-8" ?>
<DATA>
    <DATE>2012-01-30</DATE>
    <TIME>12:03:14.120</TIME>
    <CHANNELS>1</CHANNELS>
    <CHANNEL_0>
        <NAME>PHASE_A</NAME>
        <TYPE>VOLTAGE</TYPE>
        <SCALE>400</SCALE>
        <FREQ>12800</FREQ>
        <BITS>16</BITS>
        <SAMPLES>256</SAMPLES>
        <PAYLOAD>
        k1lvdSBrbm93IG1vcmUgdGhhbiB5b3U
        Gp1c3QgYXMgeW91IGtub3cgbGVzcyB0
        </PAYLOAD>
    </CHANNEL_0>
</DATA>
```

Fig. 9. Example XML datagram containing a payload of samples from the data acquisition card/device. The payload is annotated with the date and time, and some information describing the signal.

Phase Estimation, and Telecommunications. Ancillary modules include transient detection and local storage among many other potential functions. The modules are identified in Fig. 8. The communications middleware of the current release is readily adaptable into a fully featured telecoms module.

B. eXtensible Markup Language Data Interface

The OpenPMU will be hardware and software agnostic. Data acquisition devices operate interchangeably by means of a defined open interface for sample data exchange. This is achieved by encoding the waveform samples in Base64 [38] and encapsulating this payload along with metadata indicating time code and source in an eXtensible Markup Language (XML) [39] datagram.

There are advantages of XML. It is widely supported by many data analysis packages and software languages, it may be extended to include additional fields as the need arises, and it is human readable to assist in ease of code development. These features make it a preferable choice for an open-source project in which many end users may not be highly proficient programmers.

An example datagram is pictured in Fig. 9. The XML datagram is sent from the data acquisition device, or measurement module, using UDP/IP. The data acquisition may reside on the local machine or as a device on the local area network. An example implementation of an XML signal generator and UDP receiver in the Python language is available for download from the OpenPMU website. The bandwidth required is approximately 400 kb/s per measured waveform.

This method allows continued use of National Instruments hardware by means of software to perform the XML messaging. The simplicity of XML, along with its human readability, allows rapid development of interfaces for many applications. The advantage is that phase estimation algorithms implemented in other software packages, such as Matlab or Python, can be easily integrated with the OpenPMU, and sample data can originate from various types of hardware or simulation package (e.g., PSSE/DIgSilent).

Additionally, to avoid duplication of effort, it is proposed that XML can be used as a simple, although bandwidth inefficient, method of transferring estimated synchrophasor parameters to the Telecoms module for processing into the IEEE C37.118.2 format. The present implementation uses CSV sentences, but it is difficult to add new parameters to these sentences without breaking compatibility with existing software. By design, XML is extensible. The use of XML for passing synchrophasor parameters would be limited to use within the PMU device. IEEE C37.118.2 would be used external to the device.

VIII. CONCLUSION

This paper has discussed the open-source PMU project, OpenPMU, developed by QUB. The origin of OpenPMU was as a low-cost alternative to commercial products for specific research applications requiring fast estimation of synchrophasors for real-time control. A need for a common, open platform for the development of PMU technology suited to academic research is the primary motivation for the project. It is desired to avoid duplication of effort that arises from universities working independently on PMU technology without exchanging resources.

Compliance testing has been performed under both nominal and dynamic conditions. The OpenPMU complies with IEEE Standard for TVE under 99% of measurements during nominal conditions. Dynamic performance is near the P class range. The device has demonstrated comparable frequency and phase estimation results to that of a commercial PMU. Further work will identify improved phase estimation so as to yield full compliance with the C37.118.1 standard.

This system has been successfully replicated at KTH, Stockholm, and Colorado State University from the open-sourced design drawings and source code provided on the project's website. KTH has added support for IEEE C37.118.2 messaging through a communications module.

The ongoing development of OpenPMU is now building upon the principles established by the openPDC project, specifically the use of modular components with defined interfaces for information exchange. The XML format for streaming PMU sample data is flexible across many hardware and software environments. Reference programs have been prepared and are available online to allow interoperability testing.

OpenPMU is made available under open licenses to support its continued development and adoption by a wide community.

REFERENCES

[1] *IEEE Standard for Synchrophasor Measurements for Power Systems*, IEEE Std. C37.118.1-2011. [Online]. Available: http://standards.ieee.org/findstds/standard/C37.118.1-2011.html

[2] A. G. Phadke and B. Kasztenny, "Synchronized phasor and frequency measurement under transient conditions," *IEEE Trans. Power Del.*, vol. 24, no. 1, pp. 89–95, Jan. 2009.

[3] R. Lira, D. Wilson, K. Hay, T. Porreli, and F. Steinhauser, "Performance testing for interoperability of phasor measurement units to meet application requirements," in *Proc. IPTS*, Brand, Austria, 2011, pp. 1–4.

[4] D. Trudnowski, "Recommended PMU dynamic requirements for small-signal applications," Bonneville Power Admin., Portland, OR, USA, Tech. Rep. 37508, Oct. 2009.

[5] D. Macii, D. Petri, and A. Zorat, "Accuracy analysis and enhancement of dft-based synchrophasor estimators in off-nominal conditions," *IEEE Trans. Instrum. Meas.*, vol. 61, no. 10, pp. 2653–2664, Oct. 2012.

[6] P. Castello, M. Lixia, C. Muscas, and P. A. Pegoraro, "Adaptive Taylor-Fourier synchrophasor estimation for fast response to changing conditions," in *Proc. IEEE I2MTC*, May 13–16, 2012, pp. 294–299.

[7] A. Carta, N. Locci, C. Muscas, and S. Sulis, "A flexible GPS-based system for synchronized phasor measurement in electric distribution networks," *IEEE Trans. Instrum. Meas.*, vol. 57, no. 11, pp. 2450–2456, Nov. 2008.

[8] *IEEE Standard for Synchrophasor Data Transfer for Power Systems*, IEEE Std. C37.118.2-2011. [Online]. Available: http://standards.ieee.org/findstds/standard/C37.118.2-2011.html

[9] P. C. Steinmetz, "Complex quantities and their use in electrical engineering," in *Proc. Int. Elect. Congr.*, Chicago, IL, USA, Aug. 21–25, 1893, pp. 33–75.

[10] G. Missout and P. Girard, "Measurement of bus voltage angle between montreal and SEPT-ILES," *IEEE Trans. Power App. Syst.*, vol. PAS-99, no. 2, pp. 536–539, Mar. 1980.

[11] G. Missout, J. Beland, G. Bedard, and Y. Lafleur, "Dynamic measurement of the absolute voltage angle on long transmission lines," *IEEE Trans. Power App. Syst.*, vol. PAS-100, no. 11, pp. 4428–4434, Nov. 1981.

[12] P. Bonanomi, "Phase angle measurements with synchronized clocks-principle and applications," *IEEE Trans. Power App. Syst.*, vol. PAS-100, no. 12, pp. 5036–5043, Dec. 1981.

[13] A. G. Phadke and J. S. Thorp, "History and applications of phasor measurements," in *Proc. IEEE PES PSCE*, 2006, pp. 331–335.

[14] A. G. Phadke, "Synchronized phasor measurements—A historical overview," in *Proc. IEEE/PES Transmiss. Distrib. Conf. Exhib. Asia Pacific*, Oct. 6-10, 2002, vol. 1, pp. 476–479.

[15] A. G. Phadke and J. S. Thorp, *Synchronized Phasor Measurements and Their Applications*. New York, NY, USA: Springer-Verlag, 2008.

[16] P. Denys, C. Counan, L. Hossenlopp, and C. Holweck, "Measurement of voltage phase for the French future defence plan against losses of synchronism," *IEEE Trans. Power Del.*, vol. 7, no. 1, pp. 62–69, Jan. 1992.

[17] D. M. Laverty, D. J. Morrow, T. Littler, and P. A. Crossley, "Loss-of-mains detection by Internet based ROCOF," in *Proc. IET 9th Int. Conf. DPSP*, Mar. 17–20, 2008, pp. 263–268.

[18] D. M. Laverty, D. J. Morrow, A. McKinley, and M. Cregan, "OpenPMU: Open source platform for Synchrophasor applications and research," in *Proc. IEEE PES Gen. Meeting*, Jul. 24–29, 2011, pp. 1–6.

[19] R. J. Best, D. J. Morrow, D. M. Laverty, and P. A. Crossley, "Synchrophasor broadcast over Internet protocol for distributed generator synchronization," *IEEE Trans. Power Del.*, vol. 25, no. 4, pp. 2835–2841, Oct. 2010.

[20] D. M. Laverty, D. J. Morrow, R. Best, and P. A. Crossley, "Performance of phasor measurement units for wide area real-time control," in *Proc. IEEE PES Gen. Meeting*, Jul. 26–30, 2009, pp. 1–5.

[21] B. Kasztenny and M. Adamiak, "Implementation and performance of synchrophasor function within microprocessor based relays," in *Proc. 61st Annu. Georgia Tech. Protect. Relaying Conf.*, Atlanta, GA, USA, May 2–4, 2007, pp. 1–43.

[22] A. J. Roscoe, I. F. Abdulhadi, and G. M. Burt, "P-class phasor measurement unit algorithms using adaptive filtering to enhance accuracy at off-nominal frequencies," in *Proc. IEEE Int. Conf. SMFG*, Nov. 14–16, 2011, pp. 51–58.

[23] "openPDC," *Open Source Phasor Data Concentrator—Grid Protection Alliance*, Codeplex Project Page, [Accessed: Feb. 15, 2012]. [Online]. Available: http://openpdc.codeplex.com/

[24] A. J. Stadlin, *GridTrak Open Source Synchrophasor PMU Project*, Codeplex Project Page, [Accessed: Feb. 15, 2012]. [Online]. Available: http://gridtrak.codeplex.com/

[25] Garcia-Valle *et al.*, "DTU PMU laboratory development—Testing and validation," in *Proc. IEEE ISGT Europe*, Oct. 11–13, 2010, pp. 1–6.

[26] A. M. Loftsson, "On-line presentation of data from PMU stations," M.S. thesis, Tech. Univ. Denmark, Kongens Lyngby, Denmark, Oct. 2006.

[27] A. H. Nielsen, K. O. Helgesen Pedersen, and O. Samuelsson, "An experimental GPS-based measurement unit," in *Proc. Nordic Baltic Workshop Power Syst.*, Tampere, Finland, Feb. 4–5, 2002, pp. 1–6.

[28] L. Vanfretti, R. Garcia-Valle, K. Uhlen, E. Johansson, D. Trudnowski, J. W. Pierre, J. H. Chow, O. Samuelsson, J. stergaard, and K. E. Martin, "Estimation of Eastern Denmark's electromechanical modes from ambient phasor measurement data," in *Proc. IEEE PES Gen. Meeting*, Jul. 25-29, 2010, pp. 1–8.

[29] "A Report to Congress Pursuant to Section 1839 of the Energy Policy Act of 2005," US Dept. Energy Fed. Energy Regulat. Commiss., Washington, DC, USA, Feb. 2006.

[30] D. M. Laverty *et al.*, *OpenPMU Open Source Phasor Measurement Unit*, Codeplex Project Page, [Accessed: Jan. 23, 2013]. [Online]. Available: http://www.openpmu.org/

[31] R. J. Best, D. J. Morrow, D. M. Laverty, and P. A. Crossley, "Techniques for multiple-set synchronous islanding control," *IEEE Trans. Smart Grid*, vol. 2, no. 1, pp. 60–67, Mar. 2011.

[32] D. M. Laverty, D. J. Morrow, R. Best, and M. Cregan, "Anti-islanding detection using synchrophasors and Internet protocol tele-communications," in *Proc. 2nd IEEE PES Int. Conf. Exhib. ISGT Europe*, Dec. 5-7, 2011, pp. 1–5.

[33] *FFTW Manual*, Aug 2012, ver. 3.3.2. [Online]. Available: http://www.fftw.org/fftw3.pdf

[34] *NMEA 0183 Protocol*, Nat. Marine Electron. Assoc., Severna Park, MD, USA, Sep. 2009. [Online]. Available: http://www.nmea.org/

[35] Z. Huang, T. Faris, K. Martin, J. Hauer, C. Bonebrake, and J. Shaw, "Laboratory performance evaluation report of SEL 421 phasor measurement unit," Pacific Northwest Nat. Lab., Richland, WA, USA, 2007.

[36] J. O. Fernandez, "The Virginia tech calibration system," M.S. thesis, Virginia Tech, Orlando, FL, USA, 2011.

[37] A. Al Hammouri, L. Nordström, M. Chenine, L. Vanfretti, N. Honeth, and R. Leelaruji, "Virtualization of synchronized phasor measurement units within real-time simulators for smart grid applications," in *Proc. IEEE PES*, San Diego, CA, USA, Jul. 22–26, 2012, pp. 1–7.

[38] Base64 Encoding, The Internet Society, 2006, [Accessed: Apr. 12, 2012]. [Online]. Available: http://tools.ietf.org/html/rfc4648

[39] About XML: Extensible Markup Language, W3, 2003, [Accessed: Apr. 12, 2012]. [Online]. Available: http://www.w3.org/XML/

David M. Laverty (M'10) was born in Belfast, Northern Ireland, in 1984. He received the M.Eng and Ph.D. degrees from Queen's University Belfast, Belfast, U.K., in 2006 and 2010, respectively.

Currently, he is a Lecturer with the Energy, Power, and Intelligent Control Cluster at Queen's University Belfast, with research interests in anti-islanding detection, telecommunications, and synchrophasor measurement.

Dr. Laverty is a member of and volunteers with the IEEE and IET.

Robert J. Best was born in Belfast, Northern Ireland, in 1980. He received the M.Eng. and Ph.D. degrees from Queen's University Belfast, Belfast, U.K., in 2004 and 2008, respectively.

Currently, he is a Research Fellow with the Energy, Power, and Intelligent Control Cluster at Queen's University Belfast.

Paul Brogan was born in Belfast, Northern Ireland, in 1983. He received the B.Sc. degree in physics from Swansea University, Swansea, U.K., and M.Sc. degree in plasma physics from Queen's University Belfast, Belfast, U.K., in 2005 and 2009, respectively.

Currently, he is completing a Ph.D. in distributed generation control in the Energy, Power, and Intelligent Control Cluster at Queen's University Belfast.

Iyad Al Khatib (M'12) was born in Saida (Sidon), Lebanon. He received the Bachelor and Masters degrees from the American University of Beirut, Lebanon. He received the LicEng and Ph.D. degree from KTH, Sweden.

Currently, he is with iITC (a Swedish ICT company headquartered in Sweden) and with the KTH Royal Institute of Technology, Stockholm, Sweden, with research interests including electric power smart grids.

Luigi Vanfretti (M'10) is an Assistant Professor at the Electric Power Systems Division, School of Electrical Engineering, KTH Royal Institute of Technology, Stockholm, Sweden. He received the M.Sc. degree in 2007 and the Ph.D. degree in 2007, from Rensselaer Polytechnic Institute, NY, USA.

His main research interest is on the development of phasor measurement unit data applications. Since 2009, he has been serving as a Secretary of the IEEE PES CAMS Task Force on Open Source.

D. John Morrow (M'99) received the B.Sc and Ph.D. degrees from Queen's University Belfast, Belfast, U.K., in 1982 and 1987, respectively.

Since 1987, he has been a Lecturer in electrical engineering at Queen's University Belfast, Belfast, U.K., with research interests in electric power systems, power system instrumentation, and embedded generation.

Dr. Morrow is a member of the IET and also a member of the IEEE PES Excitation Systems Subcommittee.