REAL-TIME PHASOR MEASUREMENT METHOD INCLUDING A GPS COMMON TIME-STAMP FOR DISTRIBUTED POWER SYSTEM MONITORING AND CONTROL

E. Chen
Department of Electrical and Computer Engineering
University of Toronto
10 King's College Rd
Toronto, ON M5S 3G4
email: er.chen@utoronto.ca

H. Shokrollah Timorabadi
Department of Electrical and Computer Engineering
University of Toronto
10 King's College Rd
Toronto, ON M5S 3G4
email: hamid@ele.utoronto.ca

F. P. Dawson
Department of Electrical and Computer Engineering
University of Toronto
10 King's College Rd
Toronto, ON M5S 3G4
email: dawson@ele.utoronto.ca

Abstract

The supply of high quality electrical energy in power systems requires access to a synchronized phasor that is measured at various locations via a phase measurement unit (PMU). These data are then tagged with timestamps, provided by a global positioning system (GPS), and then sent to a control unit. The existing methods place demanding constraints on the computing resources; hence, a new method of extracting synchronized phasors is proposed to address this issue. The new method combines a multirate phase locked loop (MPLL), a real-time positive sequence detector, and a GPS common time-stamp unit. The MPLL is based on a PLL architecture that incorporates frequency adaptation intrinsically. The MPLL constitutes a multirate sample rate system that relaxes the implementation of a front-end analog anti-aliasing filter. Furthermore, a digital bandpass filter is included to enhance the performance of the MPLL under severe line disturbance conditions. An automatic gain control (AGC) prevents any variations of amplitude in the input signals from affecting the PMU.

Keywords*: Adaptive Synchronous Sampling, Multirate Phase Locked Loop (MPLL), Field Programmable Gate Array (FPGA), Global Positioning System (GPS), Positive Sequence Detector*

1 Introduction

The supply of high quality power is an important industry requirement. A phasor measurement unit (PMU) extracts system parameters, such as frequency and synchronized phasors, to provide reliable data for power quality studies or real-time power system control. These data are obtained from various stations and/or substations and then sent to a control unit, where the data are analyzed and control signals are generated. A common time reference is supplied by a global positioning system (GPS) for all acquired data to satisfy the needs of real-time control [1], [2].

Reference [3] outlines the shortcomings of the existing methods, such as the great demands placed on computing resources; hence, the goal of this paper is to provide a single unit solution combining the GPS and PMU in a hardware environment. The hardware approach may also be used to develop a system-on-chip (SOC) solution that is presently not available. The overall system synchronously extracts system parameters which are tagged with a common time stamp. The synchronization should not be affected by system unbalance hence the synchronizing signal should be the fundamental positive sequence component. The overall system consists of a real-time positive sequence detector, a multi-rate phase locked loop (MPLL), and a GPS unit. Section 2 details the proposed synchronization method and section 3 presents conclusions for the paper.

2 Proposed Method

A detailed discussion for each building block followed by experimental results is presented in this section. Section 2.1 and 2.2 describe the MPLL and the positive sequence detector respectively. Section 2.3 explains the overall system.

2.1 Multirate Phase Locked Loop

The multirate phase locked loop (MPLL) shown in Fig. 1 is a synchronizing unit for AC utility networks [4]. This unit operates correctly in a severely distorted environment.

The MPLL unit synchronizes to the input frequency by varying the sampling rate which is generated by a numerical controlled oscillator (NCO). A bandpass filter extracts only the synchronizing signal; the resulting signal is then passed to a downsampler. An amplitude estimation block processes the incoming signal in parallel to obtain the gain factor for normalization. An amplitude gain control (AGC) block normalizes the error signal and thus allows decoupling of amplitude variations from the phase error. The controller varies the sampling rate of the analog-to-digital converter (ADC) via the feedback from the NCO until a zero error signal appears at the controller's input.

0-7803-8886-0/05/$20.00 ©2005 IEEE

CCECE/CCGEI, Saskatoon, May 2005

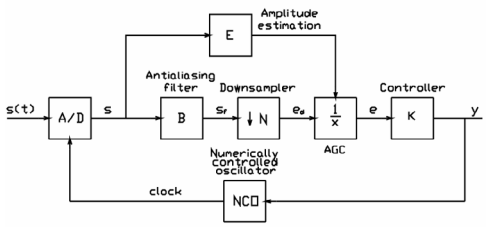

Fig. 1. Functional block diagram of the MPLL [4].

The principle of the MPLL is based on two parameters: f_{in}, the frequency of the incoming signal, and $f_{sampling}$, the frequency of the sampling signal. The MPLL ensures that the sampling frequency is always an integer multiple of the incoming frequency, according to the following relation: $N=f_{sampling}/f_{in}$.

Reference [4] described the implementation of the MPLL using a Texas Instrument microprocessor platform (TMS320C31). The conclusions of this paper were as follows:

- 60 dB immunity to impulse noise and harmonic contamination was achievable
- 20 dB dynamic range was achievable

Fig. 2 illustrates the frequency tracking performance in the presence of a notch type disturbance. In Fig. 2, the dashed line indicates the resulting synchronization signal whereas the solid line indicates the AC network voltage with notch type disturbances.

The existing platform does not allow the study of the proposed synchronization method at frequencies higher than 90 Hz. This is a constraint given that aerospace applications consider frequencies up to 800 Hz. The conventional approach to solving this problem would be to use a number of processors but then the aerospace reliability requirements can not be satisfied. Consequently, a fully integrated solution is required.

A field programmable gate array is an appropriate device for exploring hardware architectures that address the requirements of the aerospace industry. The MPLL has been implemented on a field programmable gate array (FPGA): the Altera Stratix DSP development board, professional edition [5]. Fig. 3 shows the implementation of the MPLL system block diagram. Due to a bit width limitation on the Altera DSP Builder, the software development tool, none of the data widths can exceed 51 bits [6].

As shown in Fig. 3, the main clock of the FPGA is supplied by a crystal oscillator operating at 80 MHz. This clock signal is distributed to all of the building blocks. Due to the limited number of ADCs on the development board and the bandwidth limitation of the transformer linking the onboard ADC to the input header (400 kHz – 450 MHz) [7], an on board sinusoidal waveform is generated by a NCO (Altera IP block) [8]. This IP block generates a discrete sinusoidal waveform and the output amplitude is set to {[13].[0]} (13 integer bits and 0 fractional bits). This wordlength is determined by the onboard digital-to-analog converter (DAC) (14-bits unsigned). The input signal is sent to a delay element that is enabled by the feedback NCO in order to realize the variable sampling rate. The sampled sinusoidal signal is then fed to a finite impulse response (FIR) filter (Altera IP block) [9] and its output is rescaled back to {[13].[0]} before entering the downsampler block. A peak detector extracts the peak value of the input signal every cycle. Based on this peak value, a normalization factor is determined from a look up table (LUT). The AGC multiplies the normalization factor {[13].[10]} with the resultant value from the downsampler {[13].[10]} and rescales the result back to {[13].[0]}. The error signal (the output of the AGC) drives the controller and controls the feedback signal to the NCO. A proportional integrator (PI) controller is used to achieve zero steady state error in response to phase steps or frequency steps. The value of gain and zero are 25 and 0.9375 respectively. These values are determined by selecting a crossover frequency of 1.7 Hz and a phase margin of 66°. The resultant bode plot for the loop gain that includes the PI compensator is illustrated in Fig. 4. The output of the feedback NCO is fed to an edge detector that adaptively changes the sampling rate for the delay element, the FIR, and the downsampler.

Fig. 5 shows the error signal (the controller's input) when a phase step of 120° is applied to the input NCO with an input signal having an amplitude of 13 bits full scale. The response time to the phase step is 1.22 seconds.

Fig. 6 illustrates the error signal when not only a phase step of 120° is applied but also the amplitude is reduced by half (the resolution remains at 13 bits). The response time to a step change in phase and amplitude remains unchanged thus showing the decoupling of a phase variation from an amplitude variation.

One implementation issue that should be emphasized is the amplitude estimation block. Reference [4] uses the fast fourier transform (FFT) approach to extract the fundamental component for each cycle. A peak detection strategy is used instead of a FFT in order to reduce the logic element (LE) usage in the FPGA implementation.

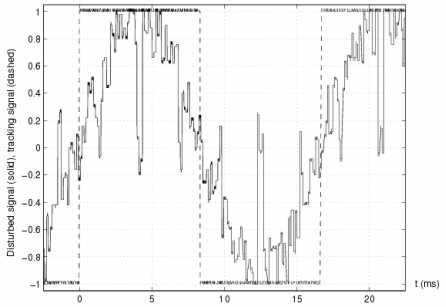

Fig. 2. Tracking performance for a notch type disturbance [4].

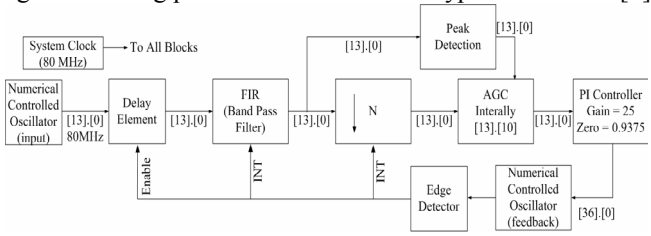

Fig. 3. MPLL system block diagram implemented on a FPGA

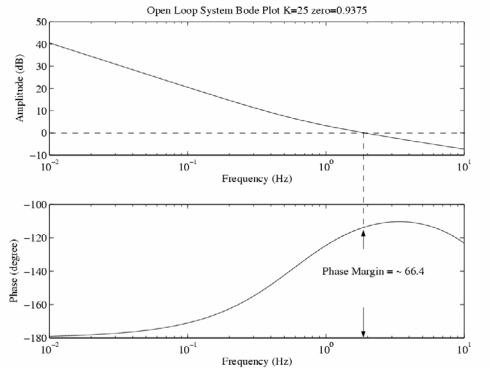

Fig. 4. Bode plot of the MPLL loop gain

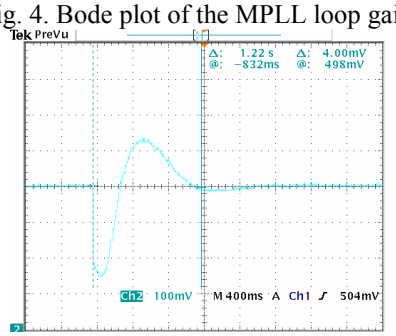

Fig. 5. Error signal (phase step of 120 degree)

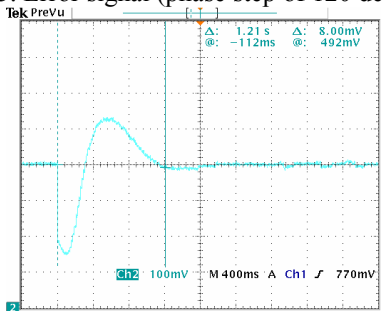

Fig. 6. Error signal (phase step of 120 degree and an amplitude step of half of the full scale amplitude)

2.2 Positive Sequence Detector

Reference [10] outlines the conventional way of extracting the positive sequence component. Reference [11] has modified the extraction method to satisfy the needs of a real-time discrete system, as illustrated in equation (2)

$$V_{pos}[k] = Va[k] + PbVb[k] - ZbVb[k-1] - PcVc[k] + ZcVc[k-1]/3 \quad (2)$$

where

$$Pb = 1/\left(\tan(\frac{\pi}{6}+\alpha)\cos(\frac{\pi}{6}) - \sin(\frac{\pi}{6})\right)$$

$$Pc = 1/\left(\sin(\frac{\pi}{6}) - \tan(\frac{\pi}{6}-\alpha)\cos(\frac{\pi}{6})\right)$$

$$Zb = 1/\left(\sin(\frac{\pi}{6}+\alpha) - \cos(\frac{\pi}{6}+\alpha)\tan(\frac{\pi}{6})\right)$$

$$Zc = 1/\left(\cos(\frac{\pi}{6}-\alpha)\tan(\frac{\pi}{6}) - \sin(\frac{\pi}{6}-\alpha)\right)$$

$$\alpha = 2\pi f_{sys}/f_{sampling}$$

V_{pos} represents the positive sequence component; Va, Vb, and Vc are the AC utility line-to-neutral voltages. The coefficients, Pb, Zb, Pc, and Zc, are in terms of the ratio (α); i.e. the ratio of f_{sys}, the frequency of the AC network, to $f_{sampling}$, the sampling frequency.

Fig. 7 illustrates the functionality of the positive sequence detector. Fig. 7a shows the input three phase AC utility line to neutral voltages and Fig. 7b shows the output of the positive sequence detector. Before t_1, the amplitude of the three phase network is half of its full scale value. At t_1, the amplitude changes to its nominal value. At t_2 and t_3, the three phase input network contains only the zero and negative sequence components respectively. At t_4, the three phase network contains only the positive sequence component whose amplitude remains at its nominal value. For the proposed synchronizing approach, the sampling frequency is varied to maintain an integer ratio between the sampling frequency and the fundamental component frequency. The frequency response of the positive sequence detector is shown in Fig. 8.

Fig. 8 shows that the positive sequence detector behaves similar to a highpass filter for a fixed sampling frequency near the fundamental component yet with respect to the entire spectrum for a fixed sampling rate, the positive sequence detector has a bandpass characteristic. The filtering effect can be also demonstrated using the FPGA implementation. Fig. 9, 10, and 11 show the positive sequence voltage and the phase A line to neutral voltage when the frequency of the three phase network is 55Hz, 60 Hz, and 65 Hz respectively. Channel 1 is the output of the positive sequence detector whereas channel 2 represents the phase A line to neutral voltage. The positive sequence has a lower amplitude than the phase A line to neutral voltage at 55 Hz, the same amplitude at 60 Hz, and a higher amplitude at 65 Hz. The FPGA implementation results confirm that the positive sequence detector has a highpass characteristic near the fundamental component for a fixed sampling rate.

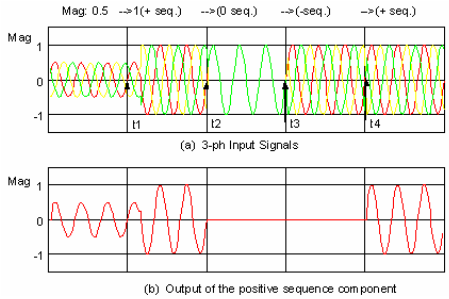

Fig. 7. Functionality of the positive sequence detector [11].

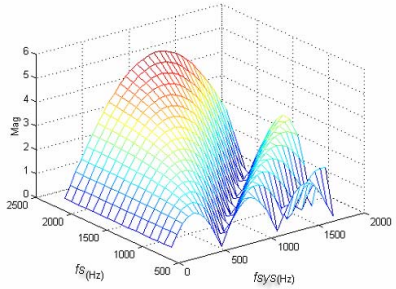

Fig. 8. Positive sequence detector freq. characteristics [11].

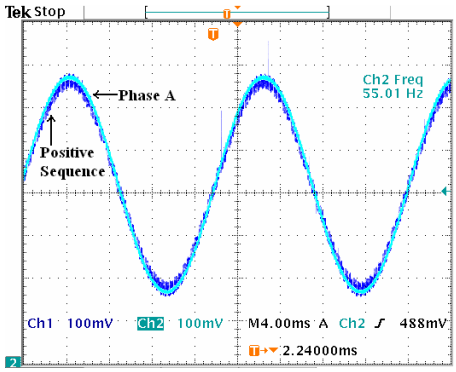

Fig. 9. Positive sequence and phase A at 55 Hz.

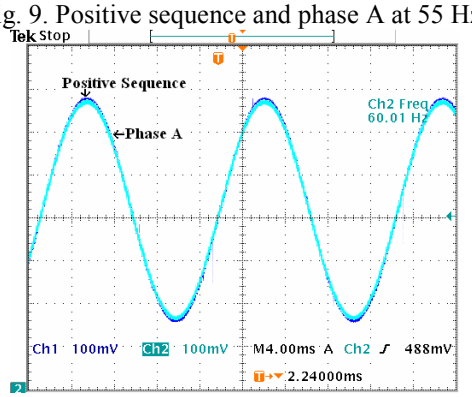

Fig. 10. Positive sequence and phase A at 60 Hz.

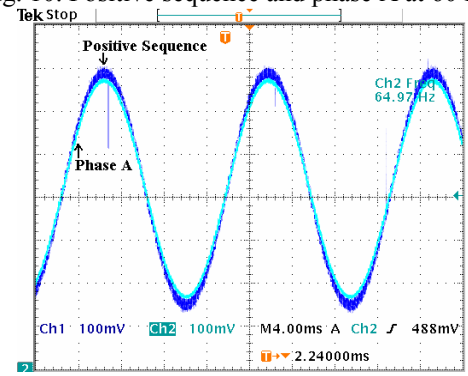

Fig. 11. Positive sequence and phase A at 65 Hz.

2.3 Overall System

Fig. 12 shows an overall system block diagram. The three phase utility signals are first sampled synchronously with respect to the network frequency (N times faster). The positive sequence detector extracts the positive sequence based on these synchronously sampled data. The purpose of the median filter is to remove the undesired impulses which occur when the AC network voltage undergoes a sudden transition [11]. The purpose of the BPF is to extract the synchronizing signal: i.e. the fundamental component. The GPS receiver provides common time-base timestamps at each zero crossing instance for measured system parameters (frequency, relative phase and amplitude) obtained synchronously with respect to the AC network frequency. The combination of amplitude, system frequency, and a common time-base timestamp are sent to a control unit that can be used to implement real-time power system control algorithms.

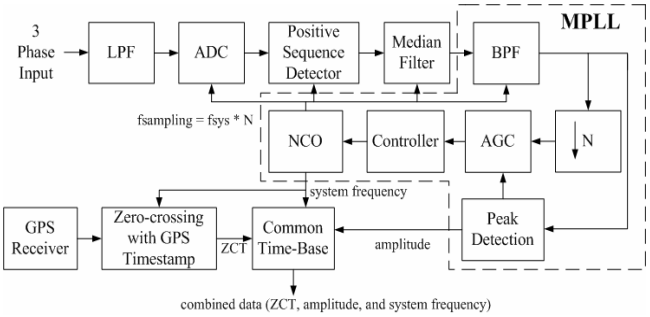

Fig. 12. Overall system block diagram.

3 Conclusion

The proposed synchronization method, which addresses the issue of robustness in the presence of severe line distortion or disturbances, provides essential information satisfying the need of real-time control and monitoring. To overcome the shortcomings of the existing methods, the proposed synchronization method has been implemented on a FPGA platform.

Acknowledgements

The Altera DSP development kit, workstation, and software development tools are sponsored by the Canadian Microelectronics Corporation (CMC).

References

[1] K.E. Martin, "Precise timing in electric power systems," *Frequency Control Symp., 47th. Proc. of IEEE Int'l*, pp. 15 – 22, June 02 – 04 1993.

[2] K.E. Martin et al, "IEEE Standard for Synchrophasors for Power Systems," *Power Delivery, IEEE Trans. on*, Vol.: 3, Issue: 1, pp. 73 – 77, Jan. 1998.

[3] H. Shokrollah Timorabadi, C. Li, and F. P. Dawson, "Application of a Fast Synchronization System in Real-Time Power System Monitoring and Control," *Proc. of 22nd Biennial Symp. on Communications*, Queen's University, Kingston, Canada, pp. 371 – 376, May 31 – Jun. 03 2004

[4] S. Pavljasevic and F. P. Dawson, "Phase Synchronization Using Zero Crossing Sampling Digital Phase-Locked Loop", *Proc. of the Power Conversion Conf.*, Osaka, Japan, pp. 665 – 670, April 2002.

[5] Altera Corporation, *Altera Stratix EP1S80 DSP Development Board*, July 2003, version 1.2, URL: http://www.altera.com.

[6] Altera Corporation, *Altera DSP Builder Reference Manual*, January 2005, version 2.1.3, URL: http://www.altera.com

[7] Altera Corporation, *Altera EP1S80 DSP Development Board Schematic*, version 1.2.0, URL: http://www.altera.com

[8] Altera Corporation, *Altera NCO Compiler MegaCore Function User Guide*, November 2002, version 2.0.2, URL: http://www.altera.com

[9] Altera Corporation, *Altera FIR Compiler MegaCore Function User Guide*, July 2003, version 2.7.0, URL: http://www.altera.com

[10] J. L. Blackburn, "Symmetrical Components for Power System Engineering", Marcel Dekker, Inc., 1993.

[11] C. Li and F. P. Dawson, "A New Algorithm for Fast Retrieval of Sequence Components in 3-Phase Networks," *Proc. of the Power Conversion Conf.,* Osaka, Japan, pp. 1357 – 1362, April 2002.