Measurement 90 (2016) 372–381

Contents lists available at ScienceDirect

Measurement

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

An implementation of a Power System Smart Waveform Recorder using FPGA and ARM cores

E.B. Kapisch [a], L.R.M. Silva [a,*], C.H.N. Martins [a], A.S. Barbosa [a], L.M.A. Filho [a], C.A. Duque [a], A.E. Tavil [b], L.A.R. de Souza [b]

[a] PSCOPE Signal Processing and Computational Intelligence for Power Systems, Electrical Engineering Program, Federal University of Juiz de Fora, Juiz de Fora, MG 36.036-900, Brazil
[b] KRON Medidores Eletrônicos, São Paulo, SP 04.760-020, Brazil

ARTICLE INFO

Article history:
Received 18 November 2014
Received in revised form 21 March 2016
Accepted 26 April 2016
Available online 4 May 2016

Keywords:
Detector
Compressor
Power Quality
Wavelet
FPGA
Embedded Processor

ABSTRACT

In this paper, the design and the prototype implementation of a Power Quality (PQ) disturbance detector and compressor are described. This instrument, named Power System Smart Waveform Recorder (PSSWR), is able to acquire the samples of the Electric Power System (EPS) signals and process them in order to detect, compress and store the disturbances waveforms into an SD card, from which it can be reconstructed and analyzed offline with a suitable computer application. The prototype uses, among other devices, Field Programmable Gate Array (FPGA) and ARM platforms to work with the EPS signals in a smart way. The PSSWR is able to detect and record either the waveform of the well-known PQ disturbance as well the new PQ disturbances not yet observed in EPS, thanks to the Novelty Detection concept. This characteristic is important in the new context of smart grids where hidden disturbances can be detected by the methodology.

1. Introduction

Nowadays there is a growing concern about Power Quality (PQ) problems. These problems are due to the massive use of non-linear loads and electronic-based equipment in residences, commercial centers, and industrial plants, and the proliferation of distributed generation in the Electric Power System (EPS). Therefore, the monitoring of EPS in real-time, along with off-line analysis using both centralized and decentralized schemes, has grown in importance [1].

In several applications the continuous data acquisition and storage are necessary. This is sustained by the fact that the post-processing of this data can uncover information not previously observed, allowing system enhancement, troubleshoot, algorithm optimization, among others. For example, high frequency disturbances like transient oscillations or switching processes are only visible in the full waveform, the signature hidden in raw data can be used to predict the breaking of cables, etc. Although aggregation is useful for data reduction and comparison, deep data analysis should be enabled considering full observation possibilities [2].

However, continuous raw data recordings of electrical signal is not a simple task due the large amount of data to be recorded and later transferred. Besides, few commercial types of equipment are currently available aiming at recording continuous raw data at high sampling rate [3]. Most of the conventional recorders are application oriented and are used only for acquiring either a short-term of failure signal or disturbance signal [4,5]. A thorough survey of the leading manufacturers of PQ analyzers was made. Nine of the top brands were examined, totaling 27 devices. From the investigation of their manuals and data sheets, it was observed that all of them are able to record the PQ parameters (data) for a long period of time. Depending on the aggregation time, some equipment are able to record over than a year of PQ parameters. Nevertheless, only two of them are able to save waveform recordings for a long period. The one described in [6] was used in this paper for a comparison purpose, because it is able to recording about 1 year of gapless waveform using a compression technique.

The system described in this paper, named Power System Smart Waveform Recorder (PSSWR), is able to reconstruct the entire waveform signal, acquired at a high sample rate. However, a continuous raw data recording is not employed. Instead, only samples around the detected electrical disturbance are packed, compressed and recorded for offline reconstruction of the whole signal. Apart those samples, an information about the frequency estimation of each cycle of the signal is also recorded. These informations are used to produce a smooth signal between two disturbances, recovering the entire electrical signal at any time, either when the frequency is time varying.

* Corresponding author.
E-mail address: leandro.manso@ufjf.edu.br (L.R.M. Silva).

http://dx.doi.org/10.1016/j.measurement.2016.04.062
0263-2241/© 2016 Elsevier Ltd. All rights reserved.

To further reduce the amount of data to be stored or transmitted, a compression algorithm is added to the system. The compression consists in eliminating the redundant information present in the signal. In general it is done trough a transform. The Discrete Wavelet Transform (DWT) [7–9] has been extensively used to this purpose due to its ability to sparse the signal.

The PSSWR has three levels of compression: (i) the novelty detection, (ii) wavelet decomposition and (iii) bit coding. These three levels allow the optimization of memory space. It ensures that long periods of measurement can be saved in a memory card.

Section 2 of this paper presents a description of the adopted methodology, showing the online compression scheme and the offline reconstruction system. In Section 3, the real time implementation of the PSSWR is shown, covering the role played by FPGA and ARM devices on the system. In Section 4, a prototype developed is shown. Some results and comments about the tests realized with the prototype of the PSSWR are presented in Section 5. Finally, some general conclusions are stated in Section 6.

2. Proposed methodology

The proposed methodology is based on the fact that only the novelties present in the signal must be saved [10]. In this way, the signal is divided in frames that contain four cycles of the fundamental component and the main objective is to identify which ones are novelty frames (frames that present some difference compared to a reference one) to be saved. Furthermore, two other stages of compression are performed: the novelty frame is compressed by a wavelet transform followed by a modified Lempel–Ziv coding algorithm that does a lossless compression.

2.1. Compression system – online

The compression system is composed by five main blocks as shown in Fig. 1: (a) the novelty Detector; (b) the Frequency Estimation; (c) the Wavelet Compression; (d) the Builder and (e) the Coding.

As mentioned above, knowing which frame is different from the reference one is the core of this system. The parameter chosen to make this decision is the energy of the signal at each frame. Before the energy calculation, the signal passes through a high-pass filter with cutoff frequency of 720 Hz in order to suppress detections due to fundamental frequency variations. Then, the energy of each frame is calculated in the output of this filter and subtracted from the energy of the reference frame. If this value is higher than a threshold, a novelty is detected.

The frequency estimation is necessary in the reconstruction algorithm since only the novelty frames are stored and the entire signal may be desired to be reconstructed. The non-novelty frames are then reconstructed based on the shape of the reference frame (the last frame of novelty) and on the averaged frequencies. The frequency estimation algorithm is the PLL (Phased Locked Loop) presented in [11,12]. The frequency is estimated instantaneously and averaged each cycle. Thus, four frequency values are stored for each frame. It is worth to mention that slow frequency variation does not trigger the novelty detector, however rapid or abrupt changes in the frequency estimation do.

When a novelty is detected, the corresponding frame of the signal must be stored. However, instead of storing the points themselves, a wavelet compression is performed in order to eliminates the redundant signal information. The filters of the wavelet were obtained from a Daubechies 3 (db3) mother wavelet and a decomposition in three levels was adopted, as shown in Fig. 2.

In this figure $v[n]$ is the signal to be decomposed, H_d and L_d are the highpass and lowpass filter respectively, CD_1, CD_2 and CD_3 are the detail coefficients and CA_3 is the approximation coefficient. The decomposition is performed in parallel with the other processes and a threshold is applied to the detail coefficients aiming at eliminating the low energy information. If they are lower than this threshold they are replaced with zero. The approximation coefficients are all remained since the processed signals have more information in the lower frequencies.

The information from the novelty detector, frequency estimation and wavelet compression is packed by a block called Builder that, according to the result of the novelty detector builds the file structure as follows:

- If the current frame is of novelty and the previous one is not: write the ‘1’ flag followed by the four frequency values and the wavelet coefficients.
- If the current frame is a novelty and the previous too: writes the ‘2’ flag followed by the four frequencies and the wavelet coefficients.
- If the current frame is not of novelty and the previous is of novelty: writes the ‘3’ flag followed by the four frequencies and one cycle of wavelet coefficients.

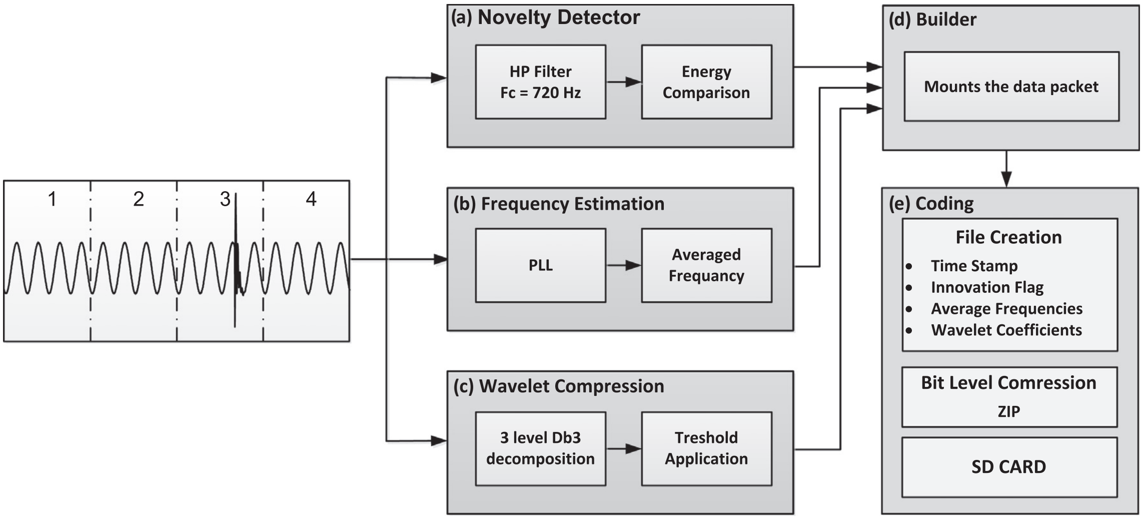


Fig. 1. Block diagram of the proposed methodology.

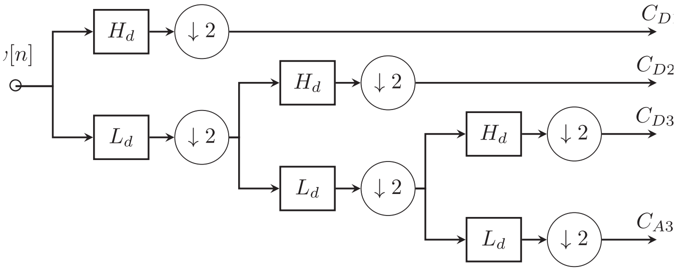


Fig. 2. Wavelet decomposition.

- If neither the current frame nor the previous one are novelty frames: writes the '4' flag followed by the four frequencies.

The extra cycle of the wavelet coefficients in the third item is justified by the fact that the wavelet reconstruction presents the boundary effect, and do not reconstruct the entire frame correctly, with the extra cycle it is possible to reconstruct the frame perfectly.

The Coding stage consists in the last compression level of this method and it is responsible for doing a lossless compression in the data, respecting the structure created by the builder. This compression is performed using the Lempel–Ziv–Welch (LZW) algorithm [13]. The stream of data from the Builder is compressed in real time and stored.

2.2. Decompression system – offline

The reconstruction system follows the flow shown in Fig. 3. The first action consists in applying the LZW decompression in the selected file. This process will generate a structured file as described before. The informations about each frame must be read consecutively. If it is an novelty frame the wavelet reconstruction is applied, and it is concatenated to the reconstructed signal. If it is not an novelty frame, the last novelty frame which was reconstructed is interpolated according with its frequency and the SWRDFT (Sliding Window Recursive Discrete Fourier Transform) [14] is applied using the frequency information. The reconstructed frame is also concatenated to the signal. The process repeats until the end of the file.

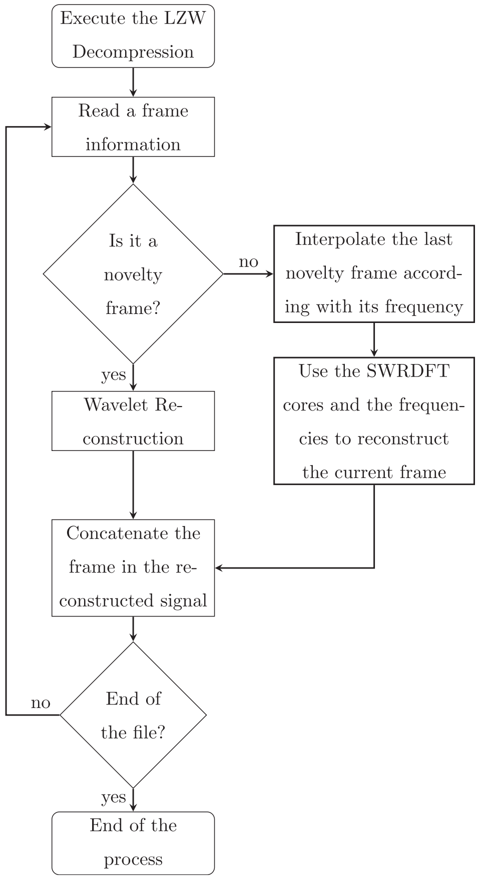
3. Real time implementation

The methodology described here was implemented in real time, using two platforms: a FPGA and an ARM Cortex M4. The former is responsible for all the signal processing as the Novelty Detector, the Frequency Estimation, the Wavelet and the Builder blocks. The latter uses a software framework to implement the Coding block, the SD Card control, management of the system and communication with the user.

3.1. FPGA

The FPGA, used for the signal processing algorithms implementations belongs to the Cyclone IV family (model EP4CE22F17C6) from the ALTERA company. This FPGA has 22,320 logic elements, 154 pins, 608,256 memory bits and 4 PLLs [15].

The external reference clock was 50 MHz. A PLL was allocated to generate a 32 MHz clock for the systems and a 2.5 MHz clock for SPI communication between the FPGA and the ARM processor. The data acquisition block, not shown in Fig. 1 is controlled by the FPGA. The used sample rate was 7.68 kHz, resulting in 128 points per cycle of 60 Hz. The A/D converter (AD7606 from Analog Device) has a 16 bits resolution and is capable of converting 8 channels simultaneously. It has three modes of interface: serial and parallel with both byte mode and word mode. The chosen interface was the word mode, where the 16 bits of the data are read in parallel.



Execute the LZW Decompression
Read a frame information
Is it a novelty frame?
no
Interpolate the last novelty frame according with its frequency
Use the SWRDFT cores and the frequencies to reconstruct the current frame
yes
Wavelet Reconstruction
Concatenate the frame in the reconstructed signal
no
End of the file?
yes
End of the process

Fig. 3. Block diagram of the reconstruction system.

The implementation of the three main blocks was performed by a customized processor synthesized in the FPGA. The Builder block consists of a Finite-State Machine (FSM). There are other auxiliary blocks implemented in the FPGA, such as: the ADC controller, a serial interface used to get custom calibration parameters, and a SPI interface to send data to the ARM processor.

The processor that was developed to this application is based on a Reduced Instruction Set Computer (RISC) Harvard Architecture and has floating point Arithmetic Logic Unit (ALU) that is capable to perform floating point operation (like addition, multiplication, comparison) in one clock cycle. In the way that this processor was developed it is possible to select via software how many bits the ALU will have for the mantissa and for the exponent. So, depending on the application it is possible to save logic in the FPGA using less bits in the ALU.

A block diagram that represents the processor is shown in Fig. 4. Besides the ALU, the processor has several other blocks among which can be highlighted the Program Counter (PC) that holds the address of the current instruction, the Instruction Decoder block that is responsible for translating the information from the instruction memory, and generating the control to all the others processor blocks. This processor also has a data stack, that is implemented as a part of the data memory, and an address stack (stack in the diagram) used to implement the call of subroutines.

To program this processor a custom C and Assembly compilers were developed using the GNU tools FLEX [16] and BISON [17]. In

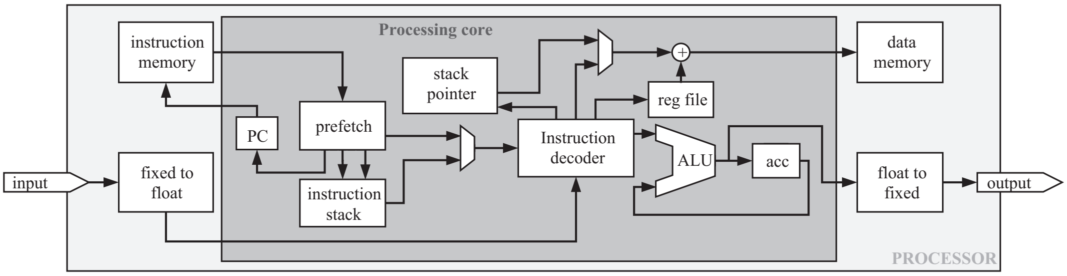


Fig. 4. Block diagram of the processor implemented in the FPGA.

this way, the source code for the novelty Detector, Frequency Estimation and Wavelet blocks could be written in this customized C language. The compilers were responsible to generate the binary files containing the machine code to be written in the instruction and data memories. The memories are implemented using internal FPGA memory blocks.

Table 1 shows the FPGA resources spent by each implemented block. It is important to note that each block is capable of processing 8 input channels. One advantage of the customized processors is that, apart the memory size, the other FPGA resources remain the same, regardless the algorithm complexity. In this way, the logic usage for processing one input channel is the same than for 8 channels.

The maximum number of allowed operations is determined by the relationship between the operation frequency of the processor and the sampling frequency. In our case it turned out that an operation frequency of 32 MHz is sufficient, since the sampling frequency is 7.680 kHz and the algorithms are relatively simple.

3.2. ARM

The ARM processor that was used is of the Stellaris/Tiva family, from Texas Instruments. It is a 32-bit ARM Cortex-M4F 80 MHz processor with single-precision Floating-Point Unit (FPU), and multi-channel serial/SPI interfaces [18].

For data compression, the used algorithm is a modification of the LZW, focusing in real time implementation on the ARM processor. The main modification of the LZW algorithm was the restriction of the dictionary size, allowing a fast search of data sequences.

The SD card management is employed by the interface with a dedicated IC (CH376 [19]) that handles the entire file system routines responsible for arranging the files and directories inside the SD card. The ARM has to control the operation of that chip as well as send the data that will be stored. A SPI interface is used to communicate the ARM with the CH376 and the data is composed by: (i) the data processed by the FPGA, (ii) a time stamp acquired from a RTC (Real Time Clock) and (iii) some control bytes needed to the decompression algorithm.

The ARM is also responsible for communicate with the user in order to collect some parameters such as the desired compression level and the number of channels that will be used. For this proposal it implements two communication interfaces, a Serial-USB and a Serial-Bluetooth. Both interfaces are capable to connect with a PC or with a Tablet, in order to parametrize the equipment or collect some information about the operation of the system.

Table 1
Logic usage in the FPGA.

	Logic cells	Memory bits	DSP blocks
Novelty detector	2096	31,324	7
Frequency estimation	1366	26,310	7
Wavelet compression	1770	25,771	2
Builder	3347	126,976	0

4. The prototype

The prototype is composed by two boards: a processing board and an analog front end board. It has the capacity of processing up to 8 input channels and it was built to process 4 voltage channels and 4 current channels.

A picture of the processing board is shown in Fig. 5. It is the core of the prototype and some components can be highlighted: besides the FPGA and ARM, there are the SD Card controller, the Bluetooth, the USB, a Li-Ion battery no-break and the ADC.

The analog front end is shown in Fig. 6. It is composed by two main circuits: The voltage circuit and the current circuit, each one is responsible for conditioning the acquired signal to the levels that are acceptable by the A/D converter. There are two options of conditioner circuits for the voltage and current signals, selectable by a jumper in the board. In the case of voltage, one can choose to use a voltage transformer or a resistor divider. The resistor circuit provides an accurate measure and the transformer circuit a galvanic isolation. The transformers were built using silicon-iron blades, so they provide a 20 kHz bandwidth. The current signals can be acquired through a current transformer (in the board) with the maximum value of 7.5 A or through an external current transform (plugged in the board). All signals also pass through anti-aliasing fourth order Butterworth filters with cutoff frequency of 3 kHz. This filters also provide magnitude gain and offset for each channel.

The system must be parametrized, so that an android application was built to transfer the parameter through bluetooth to the ARM processor that stores these parameters in its internal memory. Every time the system is powered on or during parameters updating it transfers these data to the FPGA (which has a volatile memory).

5. Practical results

Experimental tests were done using a Omicron CMC-256-6 plus source to generate the voltage and current signals for the tests. With this source it is possible to generate voltage signals up to 500 V phase-to-phase of magnitude and current signals up to 25 A, in a three-phase configuration. The test bench used to generate the results is shown in Fig. 7. This part of the test procedure aims to illustrate how the system is able to reconstruct the whole waveform signal, without discontinuities, even in the presence of some important disturbances, such as ramp voltage variation, voltage interruption and frequency variation. These tests are shown in the cases 1, 2 and 3 of Section 5.1.

In addition, comparative tests were carried out using a commercial equipment described in [6]. Finally, a real power system signal was acquired from a wind generator and the comparative results are presented in Section 5.2.

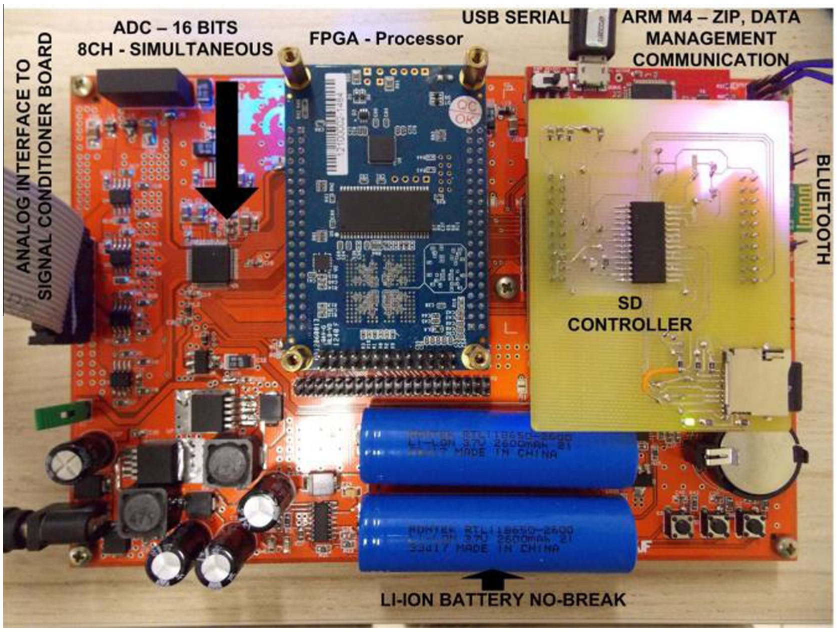


Fig. 5. Picture of the processor board.

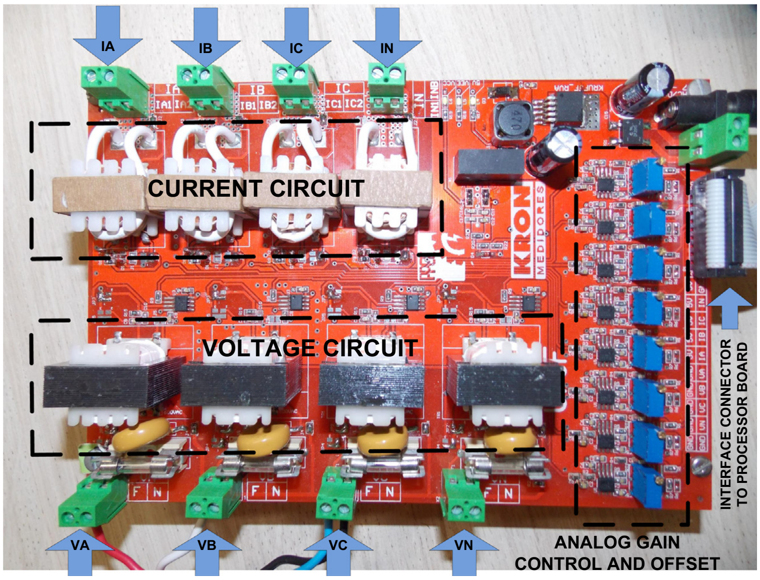

IA
IB
IC
IN
CURRENT CIRCUIT
VOLTAGE CIRCUIT
INTERFACE CONNECTOR TO PROCESSOR BOARD
VA
VB
VC
VN
ANALOG GAIN CONTROL AND OFFSET

Fig. 6. Picture of the analog front end.

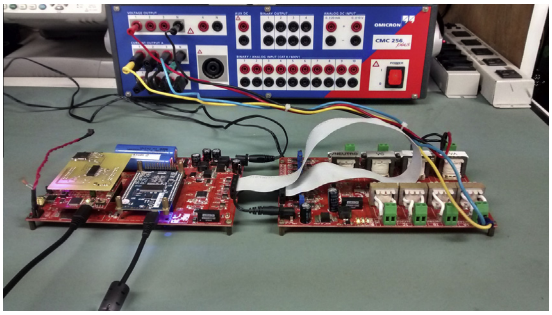


Fig. 7. The test bench for the prototype.

5.1. Illustrative tests

The signals used in these tests comprise some PQ phenomena. All the signal reconstruction figures were plotted using the same color scheme: At the superior plot, the continuous thin black line represents the original signal, the dashed blue line, over the previous one, represents the reconstructed non-novelty frames, and the continuous thick blue line depicts the reconstructed novelty frames. The inferior plot shows the Normalized Absolute Error (*NAE*) between the original and the reconstructed signal, calculated as shown in (1)

$$NAE = \frac{|x_{orig}[n] - x_{rec}[n]|}{V_{RMS}} \times 100(\%) \tag{1}$$

where $x_{orig}[n]$ are the original signal samples, $x_{rec}[n]$ are the reconstructed waveform samples, and V_{RMS} is the Root Mean Square Value of the fundamental component of the original signal.

5.1.1. Case 1

The first test is shown in Fig. 8. A ramp voltage variation was applied to a pure 60 Hz sinusoid. The magnitude of the signal varies from 180 V to 70 V in approximately 10 cycles, remains constant in 70 V during 20 cycles and then, in 10 cycles return to 180 V. This behavior is periodic each 2.5 s. It can be noted that novelties were detected only in the frames where the magnitude was changing. In this case, without any compression the signal would be recorded using 423 kB and with the proposed methodology it was recorded with only 42 kB, so the compression rate was approximately 10:1. The zoom figure highlights the correct match between the reconstructed signal and the novelty frame acquired. In Fig. 8 it is possible to see, into the bottom plot, the normalized error of the proposed algorithm. This error is smaller than 2.5% regarding the RMS value. Note the error in the non-novelties reconstruction frames is due to the small inaccuracy frequency estimation. The error in the novelty frames is due the thresholding in the wavelet coefficients.

5.1.2. Case 2

Fig. 9 shows a interruption signal with 5 cycles of duration. This interruption is repeated each 1.2 s. Novelties were detected only in a few frames. The size of the compressed signal is 85 kB while the original signal occupies 1000 kB. Hence the compression rate was approximately 12 : 1. The bottom plot presents the normalized absolute error in percentage of the RMS of the fundamental component. Again, it is possible to see that the higher error is because the inaccuracy of the frequency estimation, showing that improving the frequency estimation will decrease the reconstruction error.

5.1.3. Case 3

One of the most advantages of this methodology is that if the signal presents only frequency variations along time, novelties are not detected and the signal can be reconstructed using the reference frame and the estimated frequencies of each cycle. In Fig. 10 there is an example of this situation. The signal was generated only having the fundamental component. The frequency starts in 60 Hz and varies in ramp until reaches 62 Hz in 3 s, stays at 62 Hz for 3 s and then decreases in ramp until 60 Hz and stays there for 3 s, and then repeats this pattern. A thousand frames of this signal was recorded and only a few novelty frames were detected: six in the beginning and one in the end of the operation. In this way the size of the compressed file was 10 kB, and since it would be necessary 1000 kB to store this signal without compression, the compression rate was 100 : 1. Again, the zoom figure highlights the correct match between the reconstructed signal and the last frame acquired.

In bottom plot of Fig. 10, it can be seen the error which, in this case, approaches to 5%. It is well known the inability of the frequency estimator for tracking the actual frequency without a delay, so, a higher error is expected in this situation (see the NAE from 0.4 s up to 0.46 s).

5.2. Comparative tests

In order to compare the performance of the PSSWR with technologies currently commercially available, a deep research involving the main brands of PQ analyzers and recorders was done. As a result of this research it was found that the equipment described in [6] is the only device that has the ability to reconstruct the entire waveform of EPS signals and also incorporates a signal compression technique. Therefore, this one, named here as a Commercial Power System Waveform Recorder (CPSWR) was used with the comparison purpose. Both of equipment was submitted to the same test signals.

Several signals containing disturbances of different types and intensities were used. Some of them are shown here with the following same color scheme: At the superior plot, the continuous thin black line represents the original signal, the dashed blue line, represents the reconstructed waveform of the PSSWR, and the

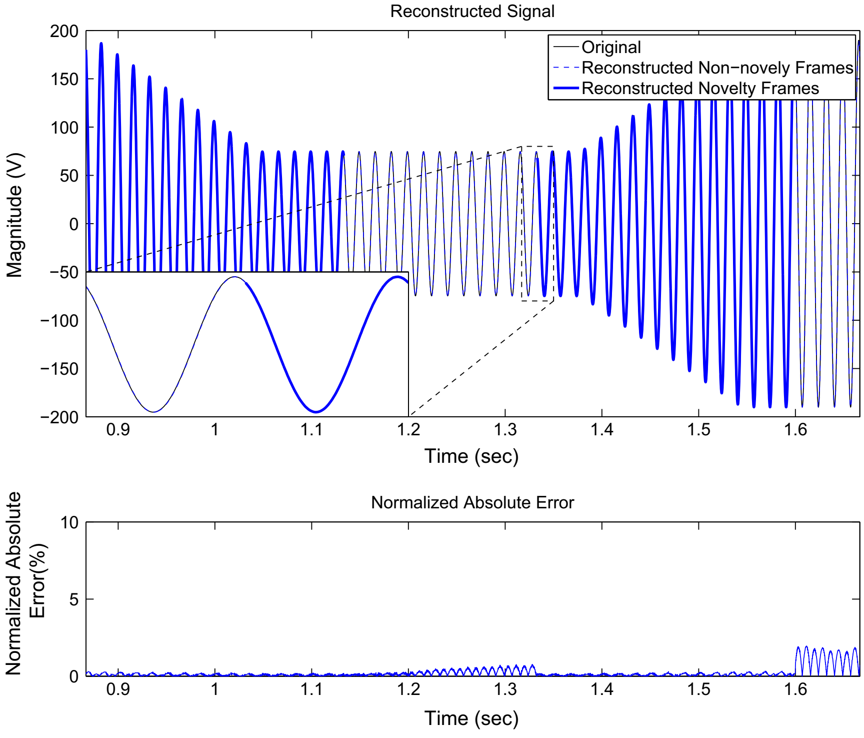


Fig. 8. Ramp voltage variation. Top: Waveforms; Bottom: NAE.

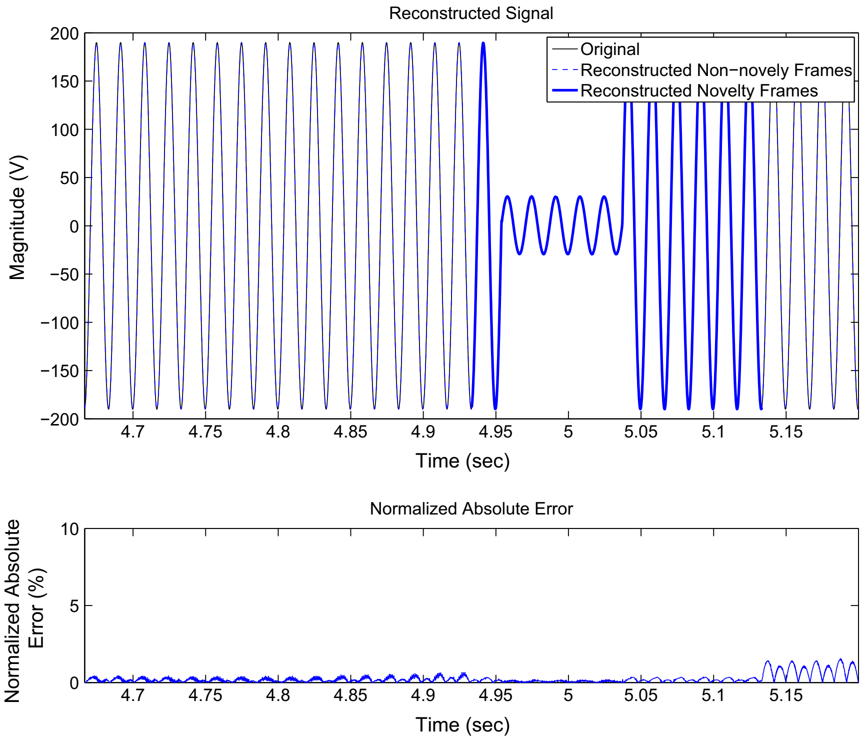


Fig. 9. Voltage interruption. Top: Waveforms; Bottom: NAE.

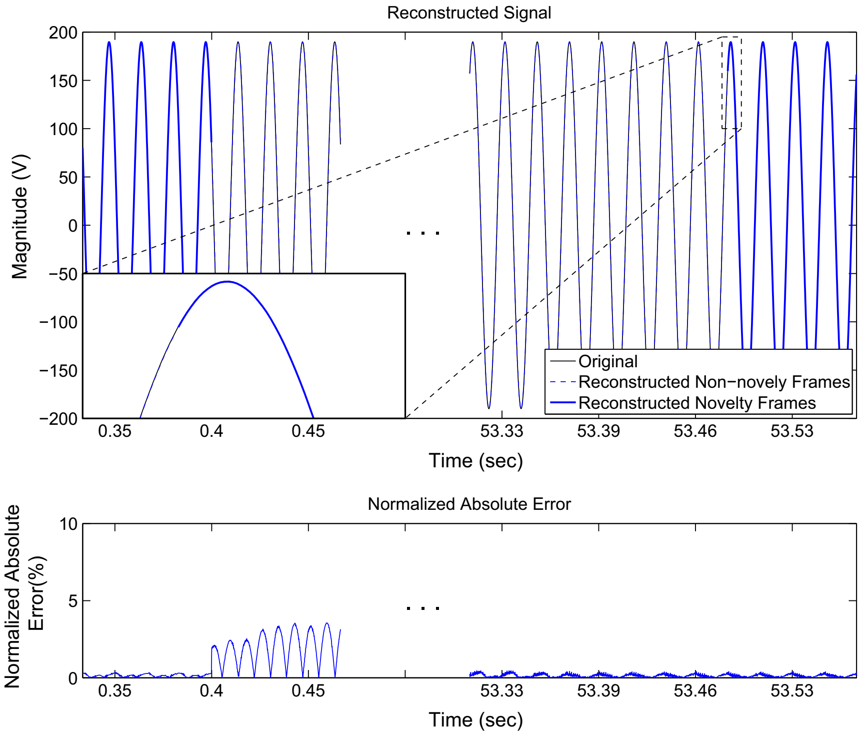


Fig. 10. Frequency variation. Top: Waveforms; Bottom: NAE.

dotted red line represents the CPSWR reconstructed waveform. The inferior plot shows the *NAE* of both equipment. The dotted red line represents the CPSWR error, and the dashed blue line represents the PSSWR error.

5.2.1. Case 4

In this case, the test signal is composed of a sequence of different disturbances interleaved with a pure sinusoidal signal with 180 V of amplitude and frequency of 60 Hz. The disturbances that were used are: Sags, Swells, Even and Odd Harmonics, Interharmonics, Notches, and ramp magnitude variations. A signal with approximately 13 min were generated.

To store the entire signal with a sampling rate of 7680 Hz (128 samples per cycle) with 16-bit resolution, 12.15 MB file would be necessary. The size of the file compressed by the PSSWR is 102.29 kB, constituting a compression ratio of approximately 119 : 1. The size of the compressed file by the CPSWR for the same test signal is 254.34 kB, more than two times larger than the PSSWR compressed file.

Fig. 11 shows a part of the mentioned signal. The top plot depicts a moment when an interharmonic component with an intensity of 10 V and frequency of 400 Hz is present. The reconstruction errors can be viewed in bottom plot.

For this case, it can be noted that exists a reconstruction error for both equipment. The error related to PSSWR is due to the fact that the current reconstruction method has 15 Hz frequency resolution, since the frame length is 4 cycles. High frequency resolution methods could be used in the offline reconstruction algorithm to overcome this problem, without modifying the online detection methodology. This is currently under investigation.

5.2.2. Case 5

In this case, a pure sinusoidal signal with fundamental frequency variation was used. The fundamental frequency varies between 59 Hz and 61 Hz. The duration of the stored signal is 86 s and the file size without compression would be 1.32 MB. The PSSWR compressed file has 28.34 kB, constituting a compression ratio of 47 : 1. The size of the compressed file by the CPSWR is 122.2 kB, which is much larger than the size of the file compressed by the PSSWR. The reconstruction errors can be viewed in Fig. 12.

This case, illustrates an important advantage of the PSSWR over the CPSWR. The signals with small frequency variations are highly compressed with good quality by the PSSWR as a result of the proposed methodology, which is based on the frequency estimation of each cycle and the SWRDFT for signal reconstruction. Additionally, another advantage of the PSSWR over the CPSWR is highlighted in the zoom box of Fig. 12, where it is possible to see discontinuities in the CPSWR reconstructed signal that are not present neither in the original one nor in the PSSWR reconstructed waveform.

5.2.3. Case 6 – Real signal

The signal used in Case 5 was acquired from a wind generator. There are 8 acquisition channels, 4 for the voltages and 4 for the currents. The signal duration is 34 h. The PSSWR records the 8 channel signals using 27 MB, with a compression ratio of 533:1, while the CPSWR used 22 MB, with a compression ratio of 651:1. A segment with duration of one hour from one of the current channels of the mentioned signal is shown in Fig. 13. In this figure it can be seen the reconstructed signals by both equipments.

The current signal was chosen to illustrate this case, since it presents more variations, what can be noted from Fig. 13. In this way, more novelties were detected. A piece of the reconstructed signals is shown in the bottom plot, where both signals are similar. Since it is a real signal, it is not possible to plot the reconstruction errors.

6. Conclusion

This paper presented an implementation of a Power System Smart Waveform Recorder (PSSWR). The methodology is based on saving the novelties present in the signal which are then compressed using lossy and lossless techniques. For detecting the signal novelties, an energy based detector is used together with a frequency estimator. The objective of the frequency estimator is twofold: (i) it works as a complementary and redundant trigger for detecting new disturbances and (ii) it estimates the frequency to be used in the signal reconstruction. Small frequency variations, as happens in "normal" signal does not trigger the novelty detector, but abrupt changing in frequency generates a new trigger. The system is implemented in FPGA and ARM processor. All digital signal processing algorithm were implemented in the FPGA, and

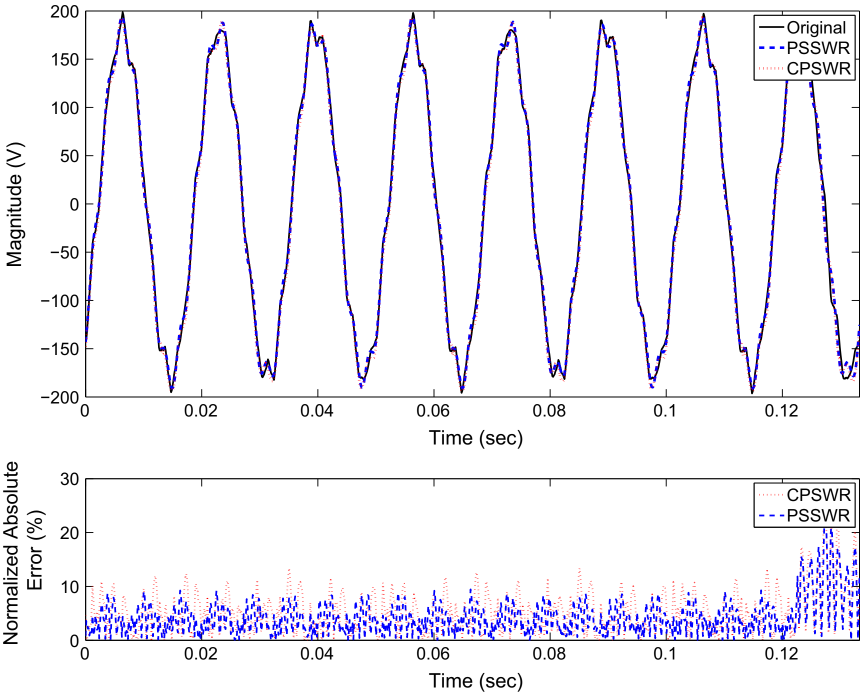


Fig. 11. Reconstructed waveform during the presence of an interharmonic component. Top: Waveforms; Bottom: NAE.

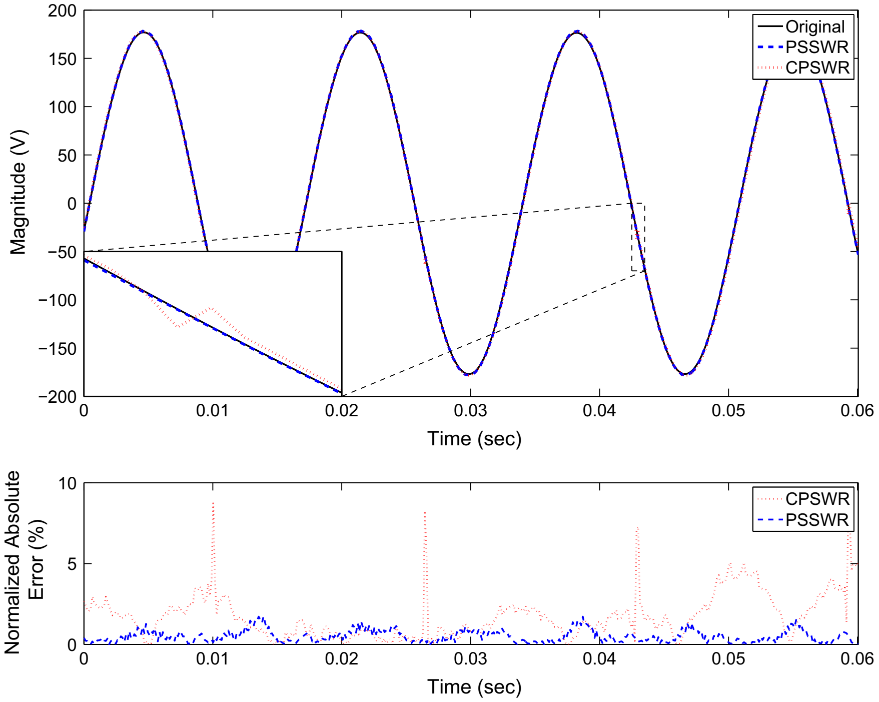


Fig. 12. Reconstructed waveform with frequency variation. Top: Waveforms; Bottom: NAE.

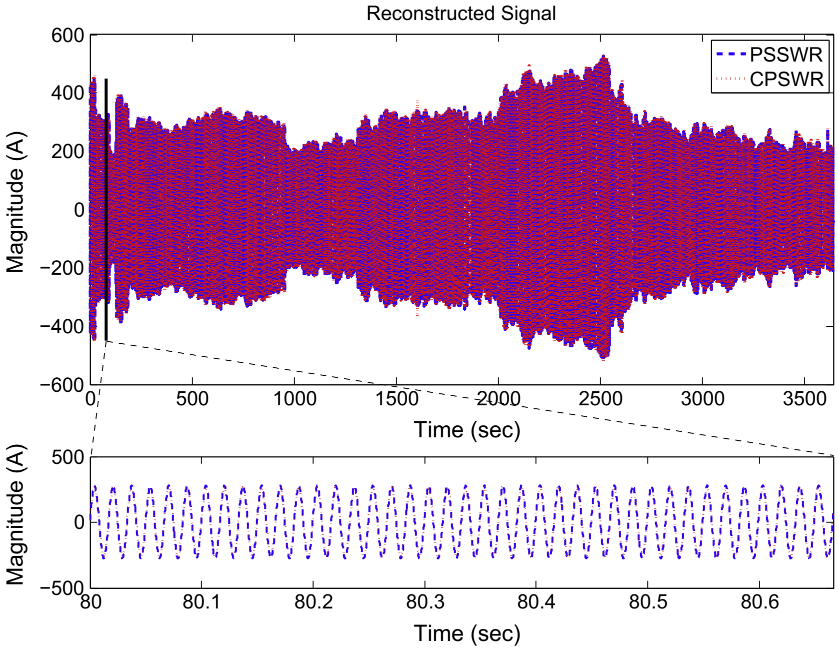


Fig. 13. Current signal during a period of one hour. Top: Reconstructed waveforms; Bottom: Zoom segment for detailed viewing.

the lossless compression algorithm, as well as the communication interfaces were implemented in the ARM processor.

Some illustrative tests were done in order to highlight the characteristics of the PSSWR, such as the Novelty Detector operation and the ability of the algorithm for matching reconstructed frames with novelty frames, even in the frequency variation scenarios. Also, comparison testes with a Commercial Power System Waveform Recorder (CPSWR) were done, using synthetic and real signals. The comparative tests show that the performance of the PSSWR is capable to overcome the CPSWR in some cases, and prove the efficiency of reconstruct the signal without discontinuities in frequency variation scenarios.

Acknowledgments

This work was supported in part by CNPq, CAPES and FAPEMIG (Brazilian research agencies).

References

[1] M.P. Tcheou, L. Lovisolo, M.V. Ribeiro, E.A. da Silva, M.A. Rodrigues, J.M.T. Romano, P.S. Diniz, The compression of electric signal waveforms for smart grids: state of the art and future trends, IEEE Trans. Smart Grid 5 (1) (2014) 291–302.

[2] H. Maaß, H.K. Cakmak, F. Bach, R. Mikut, A. Harrabi, W. Süß, W. Jakob, K.-U. Stucky, U.G. Kühnapfel, V. Hagenmeyer, Data processing of high-rate low-voltage distribution grid recordings for smart grid monitoring and analysis, EURASIP J. Adv. Signal Process. 2015 (1) (2015) 1–21.

[3] H. Maass, H.K. Çakmak, W. Suess, A. Quinte, W. Jakob, K.-U. Stucky, U.G. Kuehnapfel, First evaluation results using the new electrical data recorder for power grid analysis, IEEE Trans. Instrum. Measure. 62 (9) (2013) 2384–2390.

[4] M. Moreto, J.G. Rolim, Using phasor data records and sequence of events to automate the classification of disturbances of power generating units, Electric Power Syst. Res. 81 (7) (2011) 1266–1273.

[5] C. Qiong, W. Zhao-Hui, Research and design of portable fault recorder based on fpga, Energy Proc. 12 (2011) 686–692.

[6] P. Nisenblat, A.M. Broshi, O. Efrati, Method of compressing values of a monitored electrical power signal, 2008.

[7] H. Livani, C.Y. Evrenosoglu, A machine learning and wavelet-based fault location method for hybrid transmission lines, IEEE Trans. Smart Grid 5 (1) (2014) 51–59.

[8] M. Zhang, K. Li, Y. Hu, A high efficient compression method for power quality applications, IEEE Trans. Instrum. Measure. 60 (6) (2011) 1976–1985.

[9] F. Bezerra Costa, Fault-induced transient detection based on real-time analysis of the wavelet coefficient energy, IEEE Trans. Power Deliv. 29 (1) (2014) 140–153.

[10] C.A. Duque, M.V. Ribeiro, F.R. Ramos, J. Szczupak, Power quality event detection based on the divide and conquer principle and innovation concept, IEEE Trans. Power Deliv. 20 (4) (2005) 2361–2369.

[11] S.A.O. d. Silva, R. Novochadlo, R.A. Modesto, Single-phase pll structure using modified pq theory for utility connected systems, in: IEEE Power Electronics Specialists Conference, 2008 (PESC 2008), IEEE, 2008, pp. 4706–4711.

[12] M. Ciobotaru, R. Teodorescu, F. Blaabjerg, A new single-phase pll structure based on second order generalized integrator, in: 37th IEEE Power Electronics Specialists Conference, 2006 (PESC'06), IEEE, 2006, pp. 1–6.

[13] U. Nandi, J.K. Mandal, A compression technique based on optimality of lzw code (olzw), in: Third International Conference on Computer and Communication Technology (ICCCT), IEEE, 2012, pp. 166–170.

[14] D.F. Fabri, C.H. Martins, L.R. Silva, C.A. Duque, P.F. Ribeiro, A.S. Cerqueira, Time-varying harmonic analyzer prototype, in: 14th International Conference on Harmonics and Quality of Power (ICHQP), IEEE, 2010, pp. 1–7.

[15] Cyclone IV Device Handbook.

[16] Flex, version 2.5: A fast scanner generator. <https://www.gnu.org/software/flex/>.

[17] Bison: The yacc-compatible parser generator. <http://www.gnu.org/software/bison/>.

[18] Tiva TM4C123GXL Microcontroller Datasheet.

[19] File manage and control chip CH376.