Design of a Real-Time USB Interfaced Multi-Channel Power System Harmonics Detection System

Mehmet GÖK[1], Salih GÖRGÜNOĞLU[2], İbrahim SEFA[3]

[1]Department of Computer Engineering, Engineering Faculty, Karabük University, Turkey
mehmetgkk@gmail.com
[2]Department of Computer Engineering, Engineering Faculty, Karabük University, Turkey
sgorgunoglu@karabuk.edu.tr
[3]Department of Electric and Electronics Engineering, Faculty of Technology, Gazi University, Turkey
sefaibrahim@gmail.com

Abstract

In recent years, the increase of non-linear loads caused more harmonics on power systems. With the advent of the loads that are sensitive to power quality, harmonics are more important than in the past. Determination of those harmonics is required for the avoidance of problems. In this paper, a practical low cost harmonics detection system is designed and performance of the system is considered on single and multi-channel usage. 400th harmonic can be detected under 3.5 µs over 8- channels using the developed device and processing software. Results show that proposed system can be used for a broad range of harmonics detection systems in real-time.

1. Introduction

Harmonics can be defined as sinusoidal components of a distorted signal [1]. Frequencies of those components are integer multiple of distorted signal's fundamental frequency. In recent years increase of non-linear loads caused harmonics on power systems. Especially, those harmonics may be harmful for the loads driven by microcontrollers. Energy providers must determine and avoid the harmonics of the power line. Detection of harmonics is a critical task and many applications have been developed either to release or to use detected harmonics of monitored signals [2, 3, 4, 5, 6]. Detection of harmonics requires adequate signal monitoring and data acquisition application like implemented in [5, 7, 8, 9, 10]. Those implementations are high performance systems employing FPGAs, DSPs and hardware FIFO in different schemes. Those systems can be designed in ways more practical at lower costs.

In this paper a practical and low cost real-time harmonics detection system is developed and performance of the system is considered. The system consists of two parts: USB interfaced data acquisition unit and a computer. Data acquisition unit combines ADC (Analog-Digital Converter), ARM microcontroller and USB-FIFO (First In First Out). Signal processing application runs on the computer side. The application can either use CPU (Central Processing Unit) or GPU (Graphics Processing Unit) for signal processing. This paper is divided into these sections: Hardware Design of Data Acquisition Unit, Signal Processing Software, Performance Evaluation and conclusion.

2. Hardware Design of DAQ (Data Acquisition Unit)

DAQ unit is used to digitize analog signal and transfer data to PC. Its capture rate can be extended up to 40 KSPS over 8-channels. DAQ unit consists of three components: STM32F401 ARM Microcontroller board, AD7606 Analog-to-Digital converter and FT2232H USB-FIFO IC (Fig. 1).

FT2232 is used for buffering and transferring sampled data through USB port. FT2232 is a dual channel USB UART/FIFO solution chip. In this work FT2232 is used in asynchronous FIFO mode. One of the channels is used to send command to ARM processor: *start sampling, stop sampling and set sampling rate*. This is the control channel. Second channel is used for sending ADC data to PC.

AD7606 is a 16-bit charge redistribution successive approximation ADC with 8- channels simultaneous sampling. The AD7606 operate from a single 5 V supply and can accommodate ±10 V and ±5 V true bipolar input signals while sampling at throughput rates up to 200 kSPS for all channels. It has input clamp protection circuitry can tolerate voltages up to ±16.5 V. The AD7606 has 1MΩ analog input impedance regardless of sampling frequency. The single supply operation, on-chip filtering, and high input impedance eliminate the need for driver op-amps and external bipolar supplies. It has both serial and parallel interface making it flexible for wide range of application circuits. AD7606 is used for applications [7, 8, 9, 10, 11] requiring sensitive data acquisitions such as power quality metering and vibration detection.

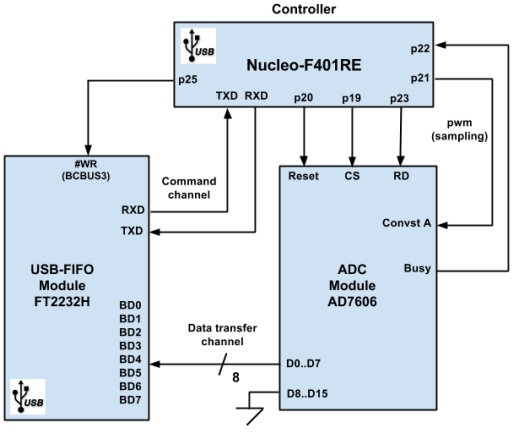

Fig. 1. Block diagram of proposed data acquisition device

The microcontroller board is Nucleo-F401RE with STM32F401RE ARM MCU. STM32F401RE is a low power Cortex-M4 ARM microcontroller can run up to 84 MHz It has 512 Kb flash program memory and 84 Kb SRAM. It has many peripheral modules like SPI, UART, I2C, and PWM making it suitable for broad range of applications. In this work, its UART module is used for data acquisition commands and PWM module is used for generating sampling signal to ADC. Nucleo-F401 is drag-and-drop programmable module making easy uploading programs to MCU. The firmware of DAQ is developed using mbed cloud compiler service. It is an ARM RVDS (RealView Development Suite) environment based service and freely available for developers [12]. The compiled binary file is saved to mbed drive and it's automatically written to MCU program flash. There is no need for an external programmer device.

ARM microcontroller generates sampling signal for the conversion, read signals for ADC and write signals for USB-FIFO unit. Sampling signal is generated using PWM (Pulse Width Modulation) module of MCU. Typically AD7606 sampling time is 4 µs. When ADC completes a sampling operation, MCU is stimulated over ADC Busy pin using falling-edge sensitive interrupt. Whenever BUSY interrupt fires, 8- read signals generated for ADC and 8- write signals are needed for FIFO unit (Fig. 2). But in this work, because FIFO channel bus width is 8-bits, ADC is read in 8-bits mode. It is required 16-read signals to transfer 16-bits wide samples data in 8- channels.

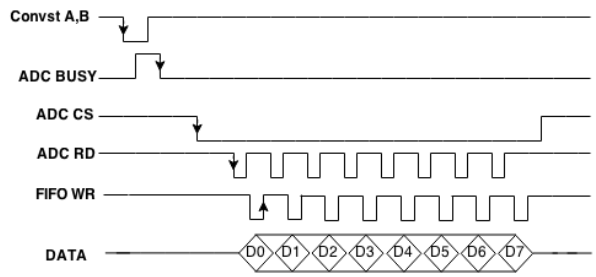

Fig. 2. Typical ADC sample, read and FIFO write signals

USB-FIFO has two communication channels through one USB connection. Its Channel A is used in UART mode to control sampling operation. Data processor application running on PC starts, stops operation or adjusts sampling rate using this channel. Channel B is used in asynchronous FIFO mode to transfer sampled data to PC (personal computer) application. Proposed device's data rate is maximum 40 kSPS x 2 bytes x 8 channels = 640 000 bytes. This data rate is handled by USB-FIFO without any issues because FT2232 maximum asynchronous data transfer rate is 8 Megabytes/second. The cost of components consisting DAQ unit in listed in Table 1.

Table 1. Cost of components of DAQ Unit

Component of DAQ	Cost (USD)
FT2232 USB-FIFO Board	20
AD7606 ADC Board	25
NUCLEO F401RE Board	13
Total	**58**

3. Data Acquisition and Processing Software

Proposed software parses and processes data coming from data acquisition device using USB FIFO channel. This software is designed using Qt 5.4 library and Microsoft Visual Studio 2013 compiler. D2XX API (Application Programming Interface) is used to communicate with data acquisition device. For discrete Fourier Transform operations, FFTW 3.3 library is used. All of the libraries linked to our application are also multi-platform compatible. Therefore it can be compiled both for Linux variants and Mac OS X operating systems.

In this paper base frequency of acquired signal is 50 Hz. When a single period of this signal completes, 0.02s x 40 000 = 800 samples are acquired and buffered. Each sample of acquired signal has a resolution of 16- bits. Because FIFO bus width is 8-bits, 1600 bytes are transferred for every period: 1600 bytes x 8 channels = 12800 bytes for 8- channels.

Data acquisition device's sampling rate is 40 kSPS. According to Nyquist approach at this sampling rate the signal components up to 20 KHz can be detected. So 20 KHz/50 Hz = 400th harmonic can be detected using this system.

Proposed software reads FIFO buffer at the end of the period of the base frequency. FT_GetQueueStatus API (Application Programming Interface) call is used to check if the buffer size reached to 12800 bytes. This value is the period limit of 50 Hz signal for 8- channels. FT_Read API call is used to read the buffered data placed into memory by the device driver. Whenever this read operation is performed, software parses data into appropriate channel. Because sample data is placed into the FIFO in the ascending channels order. While the software processes and displays parsed data, new period data is accumulated into FIFO by device driver (Figure 3).

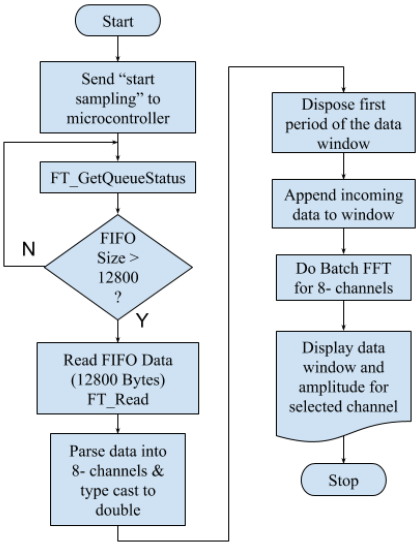

Fig. 3. Flowchart of Harmonic Detection

Processing software detects harmonics by employing Discrete Fourier Transform (DFT). DFT is a method to transform a periodic, discrete signal from time domain to frequency domain with finite length of data window. DFT of a discrete time periodic signal $x(n)$ is defined as in Equation (1) [1].

$$X_{DF}(k) = \sum_{n=0}^{N-1} x(n).e^{-j\left(\frac{2\pi}{N}\right)kn} \qquad \text{for k=01,2,..,(N-1)} \qquad (1)$$

where, n is nth sample of data, N length of data window and k is the frequency index. Resolution of frequency scale in frequency domain depends on the sample window size and sampling rate.

In this application, DFT is implemented using Fast Fourier Transform (FFT). FFT is one of the most widely used harmonics detection methods [1, 3, 6]. The software performs FFT operation using 10- periods of data. Every incoming period is placed at the end of the window using type casting to double type and first period is disposed. This is a simple operation of STFT (Short Time Fourier Transform). Short time Fourier transform (STFT) is a well-known method for time-frequency analysis of a non-stationary signal. The basic principle of STFT is to slice up the signal into suitable overlapping time segments (using the windowing method) and then to perform a Fourier analysis on each slice to ascertain the frequencies contained in it In the time domain, the STFT is defined as in Equation (2) [13].

$$STFT(t,\omega) = \int_{-\infty}^{\infty} x(\tau)\omega(t-\tau)e^{-j\omega\tau}d\tau \qquad (2)$$

where the variable $x(\tau)$ is the signal and $\omega(t-\tau)$ is the window function. The discrete form of STFT can be represented in equation (3) [1].

$$X_{SFTF}(m,f_k) = \sum_{n=0}^{N-1} x(n)h(n-m)e^{-j\left(\frac{2\pi}{N}\right)kn} \qquad \text{for k=0,1,2,..,(N-1)} \qquad (3)$$

where f_k is the kth harmonic frequency and m is an integer representing window position on time scale [1].

10- periods are used to achieve lowest possible spectral leakage [1]. Our FFT window length is 8000 samples. After each STFT operation, magnitude of the signal is calculated and the signal data is displayed using OpenGL ES 2.0 API. Open GL ES is preferred for application portability to embedded platforms and natively supported by Qt library. In Figure 4, application window view is given.

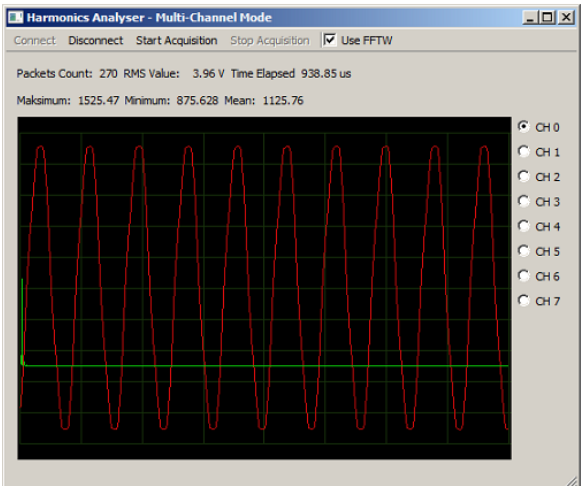

Fig. 4. Window view of proposed software

Elapsed times for data read, STFT operation and signal view is displayed on the application window. Processing time varies due to load of the operating system. To ensure the processing time varying under 20 ms limit, process priority of the processing application can be set to High using Task Manager. Average time shows average processing time. Packets label shows the count of data packets read from FIFO buffer for every period. Each packet contains 12800 bytes of data. In multi-channel mode, window of the selected channel signal and FFT result is displayed.

4. Performance Evaluation of the System

Performance evaluation of the system is realized using 220 volts 50 Hz AC power line signal. A transformer with 6 volts RMS seconder voltage is used to reduce the voltage of the line. AD7606 input range is ±10 V so seconder voltage is directly applied to differential inputs. Other inputs of AD7606 are short-circuited. Line voltage is monitored in this way.

For this application, period of the line signal is 20 ms. The proposed system must process and display the signal before upcoming period is placed into FIFO buffer. Data reading, parsing, FFT calculation, magnitude calculation and drawing operations take time here. Performance of the system is evaluated using FFTW (works on CPU) and CuFFT (works on GPU) libraries. Performance measurements are done on a notebook PC. Hardware configuration belong to PC is listed in Table 2. Measurements are done in High Performance power scheme while notebook is running on AC power. Performance measurements are done for single-channel and multi-channel modes.

Table 2. Hardware configuration of Notebook PC

Component	Configuration
CPU	Intel Core i5-2450m
RAM	12 Gb 1333 MHz
Storage	120 Gb Solid State Disk
GPU	Nvidia Geforce GeForce GT 525M (Fermi Architecture)
Operating System	Windows 7 Professional 64-bits

For CuFFT, pinned memory access method is used to avoid time consuming host-to-device or device-to-host memory operations. Elapsed processing times including data read, STFT and visualization for both single and multi-channel mode are listed in Table 3. For a single channel FFT window length is 8000 double-typed samples consisting of 10- periods of data acquired at 40 kSPS. Elapsed processing times are calculated for 6000- packets during 120- seconds.

Table 3. Elapsed times for STFT operations

Elapsed Processing Times (µs)	Single Channel Mode		Multi-channel Mode	
	FFTW	CuFFT	FFTW	CuFFT
Maximum Processing Time	578	3056	1796	2416
Minimum Processing Time	241	308	985	834
Average Processing Time	266	505	1098	1004

According to Table 3. for single-channel operation FFTW is more suitable. CuFFT library is slower than FFTW library while running for single-channel. In multi-channel operation, although maximum processing time for CuFFT is higher, mean time for processing is lower than FFTW. In that case, CuFFT library is a better choice.

Pinned memory access avoids device-to-host or host-to-device copy operations but cufftExecC2C calls must be synchronized by using calls like cudaDeviceSynchronize. Those results show that using synchronize calls lead to a decrease in performance for continuous data processing while using CuFFT library. But with use of batch FFT operations, this drawback can be avoided. On the other hand, by using CuFFT library, host CPU load can be reduced for other system and user tasks, too.

5. Conclusions

The test results show that proposed system can be used for purposes like harmonics detection or power quality metering applications. Maximum processing time shows that the system meets real time requirements for a 50 Hz or 60 Hz power system signal. FFTW processing is better for single-channel processing because CuFFT calls must be synchronized in continuous data processing. But CuFFT is more suitable for batch FFT operations even in continuous data processing.

In a future work DAQ unit will be powered using CPLD (Complex Programmable Logic Device) or FPGA (Field Programmable Gate Arrays) to use 8- channels of ADC at 200 kSPS acquisition rate. CUDA streams will be evaluated for continuous data processing with CuFFT library. The application will be ported to ARM CPU powered single board computers like Beaglebone or Raspberry Pi.

6. References

[1] Jain, S. K. and Singh, S. N., "Harmonics estimation in emerging power system: Key issues and challenges", *Electric Power Systems Research*, 81(9): 1754-1766 (2011).

[2] Penghui, L., Lijie, Z. and Yanhua, Z., "Power Quality Monitoring of Power System Based On Spectrum Analisys", *2010 International Conference on E-Product E-Service and E-Entertainment (ICEEE)*, Henan, 1-4 (2010).

[3] Lavopa, E., Zanchetta, P. and Cupertino, F., "Real-Time Estimation of Fundamental Frequency and Harmonics for Active Shunt Power Filters in Aircraft Electrical Systems", *Industrial Electronics*, 56 (8): 2875-2884 (2009).

[4] Ming, Z., Li, K. and Yisheng, H., "DSP-FPGA Based Real-Time Power Quality Disturbances Classifier", *Metrology and Measurement Systems*, 17 (2), 205-216 (2010).

[5] Wang, A., Pan, F., Li, Y. and Tao, R., "The Design of Power Quality Detecting System Based of OMAP-L138", *IEEE 13th Workshop on Control and Modeling for Power Electronics (COMPEL)*, Kyoto, 1-4 (2012).

[6] Roscoe, A.J., Sklaschus, T., Oldroyd, S.M., Blair, S.M. and Burt, G.M., "Measurement of 40 power system harmonics in Real-Time on an economical ARM Cortex-M3 platform", *Electronic Letters*, 49 (23), 1475-176 (2013).

[7] She, X. and Xiong, J., "Multi-channel Electrical Power data acquisition system Based on AD7606 and NIOSII", *International Conference on Electrical and Control Engineering (ICECE)*, Yichang, 1625-1627 (2011).

[8] He, Z. and Liao, Y., "The design of analog acquisition system in distribution automation", *2012 China International Conference on Electricity Distribution (CICED)*, Shanghai, 1-4 (2012).

[9] Guo, H., Yu, H., Sun, C., Zhang, Z. and Zheng, E., "Continuous and Real-time Vibration Data Acquisition and Analysis System Based on S3C6410 and Linux", *Fifth Conference on Measuring Technology and Mechanism Automation*, Hong Kong, 389-392 (2013)

[10] Shujing, S. and Jiansheng, L., "A Method of Multi-channel Data Acquisiton with Adjustable Sampling Rate", *Telkomnika*, 11 (9): 5299-537 (2013).

[11] "AD7606, 8-Channel DAS with 16-Bit, Bipolar Input, Simultaneous Sampling ADC" component datasheet, Analog Devices, April 2015. Online: http://www.analog.com/en/products/analog-to-digital-converters/ad-converters/ad7606.html

[12] Toulson, Rob. and Wilmshurst, T., "Fast and Effective Embedded Systems Design: Applying the ARM mbed", Elsevier Ltd., Waltham, USA (2012).

[13] Han, J., Kim, W. and Kim, C., "Fault Type Classification In Transmission Line Using STFT", *11th International Conference on Developments in Power Systems Protection*, Birmingham, 1-5 (2012).