Acquisition of Voltage and Current Signals using A Single Board Computer for Online Monitoring of Powers

Hari Arief Dharmawan\textsuperscript{1*}, Arinto Yudi P. Wardoyo\textsuperscript{1} and Chomsin S. Widodo\textsuperscript{1}

\textsuperscript{1}Department of Physics, Brawijaya University, Malang, Indonesia

\*Corresponding author: hari_arief@ub.ac.id

Abstract. During this time, Single Board Computers (SBCs) are widely used in monitoring and control applications. Many relatively low-cost SBCs are available. This paper explains the implementation of an SBC LattePanda for online monitoring of powers through Local Area Networks (LANs). The SBC was used as the main part of the monitoring system that mainly performed acquisition of voltage and current signals sent by some sensors. A program for the microcontroller on the SBC was written for the acquisition processes. Furthermore, a client/server application was developed using Delphi to perform online monitoring of the signals. Some experimentation was performed to mainly ensure that the microcontroller could capture the voltage and current signals with precise and constant sampling timing as well as send the data to the client/server application successfully. A PicoScope was utilized during the experimentation to gain the timing information performed by the microcontroller. For calibration of the voltage and current, a 4.5-digit digital multimeter was implemented. The final stage of the experimentation revealed that the intended signals could be sampled with a constant timing and in this case 128 samples per channel was chosen. The developed application could also show the expected parameters including the signal of powers using an additional remote computers connected to the LAN.

1. Introduction

Monitoring of electricity parameters are widely implemented. Some parameters that are commonly found to be monitored are voltages, currents and powers, such as discussed in [1-4]. Many current monitoring systems are supported by latest technologies of computer systems as they are getting more compact and affordable. Furthermore, with current implementation of massive storage for data and fast processing capabilities, the computer systems become a good choice for the base of various applications including electricity monitoring and control systems; see [5-7].

Currently there are many choices of computer systems, starting from microcontrollers to supercomputers and these may lead to confusion in choosing the appropriate type and family. For embedded applications, with less processing demand, microcontrollers are widely implemented. Personal computers to supercomputers, on the other hand, are more suitable for general purpose applications and less implemented in the embedded system applications. During this time, many types of single board computers (SBCs) are offered such as Raspberry Pi, Orange Pi, BeagleBoard, Asus Tinker Board and LattePanda. An SBC is a circuit board that contains a central processing unit or microprocessor, memory and input/output peripherals. SBCs have fewer capabilities compared to personal computers as they use less memory and less complex of microprocessors. However, with their
smaller sizes and more affordable prices, SBCs are more preferred as a choice in the embedded system applications, rather than the personal computers.

In this research, an SBC LattePanda is chosen to be implemented in a developed system as a main unit for online monitoring of electricity powers. To attain the power information, the system performs acquisition of voltage and current signals periodically using the microcontroller available on the LattePanda.

The following were considerations why the LattePanda was chosen:
1. Compared to microcontrollers: The LattePanda has a more powerful processing unit and more memory spaces. The LattePanda can also be programmed to provide more interactive user interfaces by implementing a display, a mouse and a keyboard.
2. Compared to other types of SBCs: The LattePanda has an on-board microcontroller, which will make it easier to perform acquisition of analog signals and control applications.
3. Compared to personal computers: The LattePanda is more compact and require less power.

Main parts that are available on the LattePanda are [8]:
1. Processor: Intel Cherry Trail Z8350 Quad Core
2. O/S: Windows 10 Home Edition
3. RAM: 2GB
4. Program/Data Storage: 32GB
5. Port & peripherals: USB, Wi-Fi, Ethernet & Bluetooth
6. Arduino hardware, containing an AVR microcontroller ATmega32u4.

2. Design of the System

2.1. Hardware

The hardware design of the developed system for monitoring the powers is shown in the Figure 1. Firstly, the signals of the voltage and current are acquired by using a voltage transformer (VT) and a current transducer (CT). Next, outputs of these VT and CT enter the analog input pins of the microcontroller on the board of the LattePanda. In the figure, these pins are marked by Arduino A0 and Arduino A1.

![Figure 1. The hardware of the developed system.](image)

Pins Arduino 4 and Arduino 9 are digital lines of the microcontroller which in this system are configured as output lines. For experimental purposes, the Arduino A9 was programmed to output a digital signal which indicate the timing of acquiring the voltage and current signals. This signal was captured using a
data acquisition PicoScope 2205A MSO. In addition, a 4.5-digit digital multimeter was utilized for calibration of the voltage and current.

The application for the microcontroller which mainly functions to acquire the signals was written using the development software Arduino IDE. This software was installed in the LattePanda along with a serial monitor application which was used during the software development for debugging purposes. The Arduino IDE was also used to write the program codes to the program memory of the microcontroller.

Beside the microcontroller that is available on the LattePanda, this SBC also contains a central processing (CPU) which can function as commonly CPUs in general computers. This will make it easier to build an application that will run on both the LattePanda and commonly personal computers. In the system, the CPU will run the client/server application related to the remote monitoring processes. This application was developed using Embarcadero® Delphi XE8. For this development, a separate computer was implemented and connected to the LattePanda through a Local Area Network (LAN). In the system, the LattePanda was accompanied with a 7 inch 1024x600 IPS display as well as a capacitive touch panel that functioned as a human user interface. Furthermore, a mouse and a keyboard were added and connected to the LattePanda wirelessly using Bluetooth.

2.2. Software

The first developed software was intended to run on the microcontroller to perform acquisition processes and data transfers between the microcontroller and the CPU of the LattePanda. Each acquisition process contained activities of sampling the voltage and current signals, analog to digital conversions, and temporarily storing as well as processing the digital information related to the signals. The Figure 2.a shows the flowchart of the program for the acquisition purpose.

As can be seen from this figure, after the initialization, the microcontroller will wait for an instruction from the CPU. Once a capture instruction is received, the acquisition process will be performed by the microcontroller. In case a print instruction is sent by the CPU, the microcontroller will send the temporary stored data to the CPU. The CPU may also send an instruction to modify parameters in the program of the microcontroller. The instructions/data are transferred from/to the CPU/microcontroller using a serial interface. In the microcontroller, the interface called Universal Synchronous and Asynchronous Receiver-Transmitter (USART) was used.

**Figure 2.** The flowchart for acquisition processes and data transfers.
The second developed software was a client/server application and which would run on the CPU. One of main functions of this software is related to a task of receiving the digital data from the microcontroller as the result of the signal acquisition. Figure 2.b. depicts a flowchart of a subroutine in the program to perform this task. Free Delphi components called ComPort and ComDataPacket were used in the program to handle serial data transfers to/from the microcontroller. The subroutine in the Figure 2.b will be executed whenever an accepted package arrives through a serial data transfer. Accepted packages are those that contain defined starting and end marks of strings representing data packages. In the subroutine, the received string will be firstly read and checked to get information of the data type. If the data contain information of a data block, then the program will read the number of packages inside the block that will arrive soon. Otherwise, the accepted package will be processed to get the digital data of the voltage and current signals. The instantaneous and average power are calculated from these digital data. Following this procedure, the results of the processed data are stored and displayed in the form of graphs and a table.

Further main function of the client/server application is to perform data transfers through a LAN. On the LattePanda, the application was run as a server. This server would serve requests of connections as well as data from one or more clients. Another copy of the client/server application was run on a separate computer and set as a client. The computer and the LattePanda were placed in the same LAN. Figure 3.a. shows the procedures that would be performed by the server and client to handle connections and data transfers through the network. In these procedures, both server and client were initiated by assigning the server’s Internet Protocol (IP) address and port (steps 1 and 2). The data request procedure done by the client (steps 6 to 8) would be repeated to perform online monitoring, once the client was successfully connected to the server (steps 3 to 5). The monitoring processes were terminated by the disconnection procedure (steps 9 and 10).

[Flowchart image: Procedures for the client/server (a) and the screen capture of the client/server application (b).]

3. Experimentation Results and Discussions

The most important part of experimentation was the one that was related to the signal acquisition timing performed by the microcontroller on the LattePanda. To get the timing information, the microcontroller was programmed to provide pulses through its digital pin (Arduino 9). These pulses were recorded by the PicoScope which was connected to a computer through a USB connection. The microcontroller would generate pulses by toggling the logic state of the pin when the signal sampling was started or ended.

The first stage of this experimentation was intended to attain the fastest sampling time that could be performed by the microcontroller when consecutively reading and converting two analog inputs, i.e. 128 samples for each voltage signal and current signal, without additional delay. The recorded pulses are
illustrated in Figures 4.a. and 4.b. These recorded pulses show that the fastest sampling time is 241.7 μS; see the start mark of 2\textsuperscript{nd} and 3\textsuperscript{rd} sampling, and measured delta time of these marks in the Figure 4.b. It can be also seen that the first sampling time differs from the rest sampling times, while the rest sampling times are the same. The difference in this case was caused by the longer time taken to perform the first analog to digital conversion.

As a solution, the result of the first conversion was ignored and 128 samples of the voltage and current signals were taken following this first conversion. Furthermore, additional delay was implemented in the program to achieve sampling time of 468.8 μS. This time was needed to achieve 128 samples per channel of 50 Hz signals within three periods. Figures 4.c. and 4.d illustrate the timing after modification of the program. The result showed that the 128 samples per channel could be achieved during 60 mS with a precise and constant sampling time. Hence, the acquisition processes could be performed by the microcontroller successfully.

![Image](image-url)

Figure 4. Pulses showing timing related to the signal acquisitions, recorded by the PicoScope.

In addition to the acquisition processes, the appropriate online monitoring mechanism was needed. In this research, the LattePanda was intended to handle multi-client services through a network. To achieve the goal, the client/server application was developed by implementing Indy components, i.e. idTcpServer and idTcpClient, that were available in Delphi XE8. Furthermore, the application was programmed so that a service for each client was handled by a separate thread.

The server and client were initialized with an IP address and port of the server by the use of these instructions:

```
idTcpServer.Bindings.Add.SetBinding (for the server)
idTcpClient.Host (for the client)
idTcpClient.Port (for the client).
```

To enable sending and reading strings, a client would need to firstly make a connection to the server by using an instruction `idTcpClient.Connect`. Following this step, the data string could be sent and read by using the following instructions:

```
AContext.Connection.Socket.WriteLine (the server send a string to the client)
```
AContext.Connection.Socket.ReadLn (the server read a string sent by the client)
idTcpClient.Socket.WriteLn (the client send a string to the server)
idTcpClient.Socket.ReadLn (the client read a string sent by the server).
The server would read string sent by the clients through ‘OnExecute’ event of the idTcpServer.

All of those instructions were written in the client/server application and implemented using the procedures as previously shown in the Figure 3.a. Result of the experiment showed that these procedures could perform connections and data requests successfully. In order to enable a client to request data from the server periodically, a timer was implemented in the client and the socket write string instruction representing a request was written inside the ‘OnTimer’ event handler.

Experimentation related to the developed client/server application was carried out by running the application as the server on the LattePanda and two copies of the applications run as clients on two additional computers which were separated and located on the same network. The two computers represented multi-client nodes that would request signal data from the server. The experimentation result revealed that the multi-client requests could be handled by the server through implementation of TThreadList class and records in the Delphi program. Information of connected clients and their related pointers could be temporary stored in the list. Next, the signal data received by the clients could be read and processed to get parameters including powers. Hence, it could be inferred that the online monitoring with multi-client services could be carried out by using the developed application.

4. Conclusion
The implementation of SBC LattePanda in the online monitoring of powers has been discussed in this paper. The SBC was used as a main part of the developed monitoring system that mainly performed acquisition of voltage and current signals sent by some sensors. Some experimentation that has been carried out revealed that the system could capture the voltage and current signals with precise and constant sampling timing as well as provide the data through the developed client/server application. Furthermore, it could be inferred that the online monitoring of powers with multi-client services could be carried out successfully.

References
[1] Irmak E, Köse A and Göçmen G 2016 Simulation and ZigBee based wireless monitoring of the amount of consumed energy at smart homes, 2016 IEEE International Conference on Renewable Energy Research and Applications (ICRERA), Birmingham, 2016, pp. 1019-1023.
[2] Dharmawan H A, Wardoyo A Y P and Widodo C S 2017 Timing management for acquisitions of AC voltage and current signals using an AVR microcontroller based system, 2017 International Seminar on Sensors, Instrumentation, Measurement and Metrology (ISSIMM), Surabaya, pp 1-4.
[3] Dharmawan H A and Ali S A M 2012 A Compact Remote Monitoring System for a Three-Phase 10-kVA Energy-Efficient, IEEE Transactions on Instrumentation and Measurement 61(3).
[4] Apse-Apsitis P, Avotins A and Ribickis L 2011 Concept of low-cost energy monitoring system for household application, Proceedings ELMAR-2011, Zadar, pp 149-152.
[5] Ikoma D, Abe Y, Kawazu N, Toda H, Sato M and Aomori H 2018 Noncontact heart rate measurement system on single board computer, 2018 International Conference on Electronics Packaging and iMAPS All Asia Conference (ICEP-IAAC), pp 321-323.
[6] Toshniwal K and Conrad J. M 2010 A web-based sensor monitoring system on a Linux-based single board computer platform, Proceedings of the IEEE SoutheastCon 2010 (SoutheastCon), Concord, NC, pp 371-374.
[7] Ahmad R. B, Mamat W M A, Juhari M R M, Daud S and Arshad N W 2008 Web-based wireless data acquisition system using 32bit single board computer, 2008 International Conference on Computer and Communication Engineering, Kuala Lumpur, pp 777-782.
[8] https://www.dfrobot.com