Design and Development of an Asynchronous Serial Communication Learning Media to Visualize the Bit Data

D Hariyanto\textsuperscript{1}, A C Nugraha\textsuperscript{2} and A Asmara\textsuperscript{3}

\textsuperscript{1,2,3} Yogyakarta State University, Indonesia Department of Electrical Engineering Education, Faculty of Engineering, Yogyakarta State University, Indonesia.

E-mail: didik_hr@uny.ac.id

Abstract. There were some limitations in the old version of serial communication learning media. The old module only showed some of data bits instead of full format data bits. There was no a signal graph display on that module as well. By utilizing the graphics LCD on our proposed module, the serial data information could be presented completely. This paper presents the design and development of an asynchronous serial communication module which can be used as a learning media in Data Communication course. To evaluate the serial module, four scenarios of the functional-based test has been performed, i.e. 1) performing basic functions test, 2) receiving data test, 3) sending data test, and 4) visualizing data test. The results showed that the serial communication module could performed all its intended functions correctly.

1. Introduction

The Department of Electrical Engineering Education in Yogyakarta State University is divided into three Study Program, namely 1) Electrical Engineering Education Study Program, 2) Electrical Engineering (Diploma) Study Program, and 3) Mechatronics Engineering Education Study Program. The students from those three study programs should take one compulsory course, Data Communication Practice, which has 2 credit [1]. In this course, the students learn about the data communication either in serial transmission or parallel transmission theoretically and practically. During this course, students commonly meet difficulties in serial communication practice due to the limitation of learning media.

The old learning media for doing serial communication practice has some limitations. It only shows the serial data through some LED installed on the project board. There is no information concerning serial data bits in detail. It cannot show the start bit, data bits, and stop bit. Therefore, students have difficulties to understand the serial communication comprehensively.

Based on that situation, our study aims to design and develop an asynchronous serial communication module as a learning media. This module has an ability to visualize the signal graph of the serial data bits completely, begins with the start bit, data bits, and stop bit. This module also displays the information about the data bits in binary, char, and hexadecimal form. The functional-based test is also conducted in order to get the performance level of the serial module.
2. Asynchronous Serial Communication

In the context of data communication, there are two kinds of the data sending process, namely 1) serial communication and 2) parallel communication. Serial communication is the process of sending data one bit at a time, sequentially, over a communication channel. This is in contrast to parallel communication, where several bits are sent as a whole, on a link with several parallel channels.

Serial data communication is divided into two ways, synchronous and asynchronous [2]. Synchronous transmissions are synchronized by an external clock, for instance, data transmission on a keyboard. Meanwhile, asynchronous transmissions are synchronized by special signals along the transmission medium. The example of asynchronous data transmission is the use of Universal Asynchronous Receiver Transmitter (UART) on the serial port (COM) of computer.

![Figure 1. Asynchronous serial bits data](image)

As can be seen in Fig. 1, that is an example of a serial communication signal. Commonly, the serial communication signal begins with one start bit, data bits, parity bit as a data check, and closed with 2 stop bits.

3. System Design

The serial communication module that we designed and developed is used to communicate with a computer. The computer that can be used in this project is either a desktop-based computer or notebook-based computer. Furthermore, the computer should be installed with the Windows operating system and it should work properly. Since most of modern computers is equipped with the USB port instead of the serial communication port (COM) [3], the connection between both computer and serial communication module will be conducted via USB port. Though the communication is physically connected via USB port, the data specification which is transferred along with that cable still complies with the serial data specification.

Looking at the diagram block of system, as seen in Fig. 2, it consists of two sub-circuits and one graphic LCD. The first circuit is the serial interface circuit. This circuit is used as the interface port to communicate with the computer. This circuit has an ability to convert the data sent by computer in the USB data format to serial data mode. The following circuit is the main controller circuit. There is a
microcontroller installed on it and functioned to process the serial data to display at the graphic LCD. The LCD will display the serial data for each bit related, such as start bit, data bits, and stop bit.

The main controller circuit as seen in Fig. 3 basically composes with one ATMega32 microcontroller, the external oscillator components, one eight-pin DIP switch, and one push button. ATMega32 is chosen because it is one often used and easy to apply. It is high-performance and low-power 8-bit AVR RISC-based microcontroller. It has a big enough 32KB flash memory size and operates in the wide voltage range from 2.7 to 5.5 Volt [4].

The main controller circuit can be set in two different functions, namely 1) to receive the serial data sent from the computer and visualize it to the graphic LCD, and 2) to send the serial data to the computer and in the same time visualizing it in the graphic LCD. In normal condition and without any setting in a DIP switch and push button, the controller will function as a data receiver. Moreover, if we want to make it as a data sender, we have to set the DIP switch as the representative of 8-bit serial data and then push the "SEND" button to send it to the computer. The 128x64 graphic LCD attached in the circuit is as an output device to visualize the serial data bits sent or received. The serial data bits are displayed in the form of one start bit, the data bits and stop bit. The graphics LCD also displays the data in the character format and hexadecimal number. There is also a status of "send", "receive", and "standby" to recognize the state of the controller at the actual moment.

Fig. 4 below shows the schematic of the serial interface circuit. The main component in this circuit is IC PL2303 which is a low cost and high-performance USB-to-Serial Bridge Controller. The PL2303 could simulate the traditional COM port on most operating systems and be made for USB ready [5].
This circuit is made based on the fact that the modern computer is almost most of it equipped with USB rather than COM port.

4. Development Phase

After the process of system design has been completed, the work continues at the development phase. It starts by producing the PCB, attaching and soldering all of electronic components, and downloading the free-of-bug software to the microcontroller.

In order to make the module more compact and handy, the use of single-layer PCB and SMD (Surface Mount Device) components is considered in this stage. Moreover, two sub-circuits as mentioned earlier are integrated into one piece of PCB.

The next step turns into the software work. There are two different programs that should be downloaded into two different devices. One is that should be installed on the computer and the other one is in the serial communication module. The software installed on the computer creates by using Delphi IDE (Integrated Development Environment). Furthermore, the serial communication module is programmed with Code Vision AVR software. The GUI of computer software can be seen in Fig. 5 and the output display of the serial communication module can be observed in Fig. 6.
5. Functional Based Test

The thorough test was conducted in the context of functional-based test in order to evaluate the performance of the serial communication module whether it could function as designed or not. Since the module comprises of hardware and software as one system, it needs to consider testing both hardware and software in one single test. When the software-based test performed, it also affects the hardware performance automatically. According to [6], software testing is an essential element to assure the system could react as its designed.

In the software coding stage, it can be conducted a debugging process. Debugging is the process to find the computer program problems and to resolve it until the program run correctly. Three coding errors may be avoided before continuing to the functional-based test: 1) language error or syntax error, 2) run-time error, and 3) logical error [7].

In the software testing, generally, there are two basic classes of test, namely 1) Black Box Testing, and 2) White Box Testing. Black Box Testing is commonly known as Functional Testing [8]. This test focuses only on the outputs generated in response to the selected inputs and ignores the internal mechanism of a system [9]. Meanwhile, White Box Testing is related to Structural Testing [8] which takes into account the internal mechanism of a system [9]. Although, the ideal option for software testing is by performing both Functional and Structural Testing, however, as it has been conducted by [10], the functional-based test is the reasonable choice to validate the system [11].

In this study, we have conducted four sessions of functional-based test. The first is related to the basic functions test and the other three are correlated to 1) performance test in receiving data, 2) performance test in sending data, and 3) performance test in visualizing the data to the graphics LCD.

The first test, as can be seen in Table 1, has been performed by doing the scenario mentioned in the functionality test item and then observing the systems' respond. If the system could behave in accordance with the scenario, write "Yes" in the column "Yes/No", otherwise write "No".

| Functionality Test Item                                                                 | Yes/No | Remark                                      |
|---------------------------------------------------------------------------------------|--------|---------------------------------------------|
| USB cable is connected, the indicator lamp can turn on                                 | Yes    | The system responds as designed             |
| Setting port serial in the computer, then the computer can connect to the             | Yes    | The system responds as                      |
|                                                                                       |        |                                             |
The serial data sent from the computer, the serial communication module can receive the data. The system responds as designed. The serial data sent from the serial communication module, the computer can receive the data. The system responds as designed.

Table 2 below describes the performance of the serial module when receiving data sent from the computer. Prior to the test, the serial communication program which is installed in the Windows-based computer should be ran to set up the communication properties, i.e. port number, baud rate, etc. The test begins by sending the character from the computer and observing the graphics LCD which represented as the receiver (the serial module) whether it received the same data or not. If the data received is similar with the data sent, it means that the system can perform as expected.

**Table 2. Data receive test**

| Data sent from computer | Data received by serial module | Hexa-decimal | Remark |
|-------------------------|--------------------------------|--------------|--------|
| A                       | A                              | 41h          | The system responds as designed |
| B                       | B                              | 42h          | The system responds as designed |
| a                       | a                              | 61h          | The system responds as designed |
| b                       | b                              | 62h          | The system responds as designed |
| 1                       | 1                              | 31h          | The system responds as designed |
| 2                       | 2                              | 32h          | The system responds as designed |
| *                       | *                              | 2Ah          | The system responds as designed |
| &                       | &                              | 26h          | The system responds as designed |
| %                       | %                              | 25h          | The system responds as designed |
| @                       | @                              | 40h          | The system responds as designed |
| ?                       | ?                              | 3Fh          | The system responds as designed |
| >                       | >                              | 3Eh          | The system responds as designed |
| <                       | <                              | 3Ch          | The system responds as designed |
| {                       | {                              | 7Bh          | The system responds as designed |
The capability of the serial module in sending data is examined by setting the serial module as the sender. Set the data through 8-pin DIP switch and push the SEND button to start the data sending process. Check the data received on the computer. Compare both data sent and received. If they represent the same data, this means that the system passes the data sending test. See Table 3 to get the result of the performance system on data sending.

Table 3. Data sending test

| Data sent from serial module | Data sent | Data received by computer | Remark |
|-----------------------------|-----------|---------------------------|--------|
| I                           | I         | I                         | The system responds as designed |
| J                           | J         | J                         | The system responds as designed |
| =                           | =         | =                         | The system responds as designed |
| :                           | :         | :                         | The system responds as designed |
| ,                           | ,         | ,                         | The system responds as designed |
| 3                           | 3         | 3                         | The system responds as designed |
| 5                           | 5         | 5                         | The system responds as designed |
| $                           | $         | $                         | The system responds as designed |

The last test is performed by comparing the data shown in the graphics LCD with the data sent either from the computer or serial module. It can be seen in Table IV that the graphics LCD can visualize the data in character form, binary data, and hexadecimal. It also draws the signal graph of the serial communication and displays the start bit, data bits, and stop bit.
Table 4. Data bit visualization test

| Char | Biner   | Data Bit Visualization | Remark                      |
|------|---------|-------------------------|-----------------------------|
| M    | 01001101|                         | The system responds as designed |
| 3    | 00110011|                         | The system responds as designed |
| =    | 00111101|                         | The system responds as designed |
| $    | 00100100|                         | The system responds as designed |

6. Discussion and Conclusion

The development of this serial module is not only focuses on hardware aspect but also related to the software programming that will be installed in computer and in the serial module. The functional-based test has been conducted in order to explore the performance level of the serial communication module. In this study, we made four sessions of the functional test in order to examine the capability of the serial module in more detail.

The first test has been performed by following the scenario described in the functionality test item. The respond of the system is observed by watching the system behavior. Based on the test result (see Table I), it can be concluded that the system could function as designed.

The second test has an objective to assess the performance of the serial module when receiving data sent from the computer. As seen in Table II, 15 test scenarios have been administered in this test. The results revealed that the serial module could receive the data as same as it sent from the sender (the computer).

The following test conducted in order to examine whether or not the serial module could send the serial data to the computer. The test ran 8 items test as it is shown in Table III. The results showed that the serial module could send the data correctly and the computer could receive it accordingly.

The last test has been managed to assure the user could easily understand and recognize the information of the serial data in depth. By observing the graphics LCD screen, it can be seen the signal graph, the start bit, data bits, stop bit, and the data in character as well as in hexadecimal form. The use
of the graphics LCD helps user to easily understand the data bit sent through serial communication mode. The result as described in Table IV showed the good performance of the serial module.

All in all, based on the four thorough sessions test, it can be summarized that the serial communication module we have designed and developed could perform as expected. By using this serial module as a learning media in the learning and teaching process, the students’ comprehension of serial data communication topic could be improved. Besides that, it is expected less time needed to understand that particular topic.

7. References

[1] TIM 2009 Kurikulum Fakultas Teknik UNY (Yogyakarta: FT UNY)
[2] W Stallings 2011 Data and Computer Communications (Pearson Education India)
[3] B Moon 2002 Survey on Serial Busses in Proceedings on the 15th CISL Winter Workshop
[4] Atmel 2011 Datasheet Atmel ATmega32 (Tech. Support)
[5] Prolific 2013 Datasheet Prolific PL2303 (Tech. Support)
[6] R S Pressman 2005 Software engineering: a practitioner’s approach (Palgrave Macmillan)
[7] H M Jogiyanto 2005 Analisis dan desain sistem informasi (Yogyakarta Andi Offset)
[8] L Luo 2001 Software testing techniques (Inst. Softw. Res. Int. Carnegie mellon Univ Pittsburgh PA) vol 15232 no 1–19 p19
[9] J Radatz, A Geraci & F Katki 1990 IEEE standard glossary of software engineering terminology IEEE Std vol 610121990 no 121990 p3
[10] D Hariyanto & T Kohler 2017 An Adaptive User Interface for an E-learning System by Accommodating Learning Style and Initial Knowledge (Proc Int Conf Technol Vocat Teach) ICTVT
[11] L Williams 2006 Testing overview and black-box testing techniques Retrieved Agustus vol 12 p2015