



INTISARI

Seiring perkembangan teknologi IoT, berbagai teknologi pendukung IoT juga ikut berkembang pesat. Salah satunya adalah sistem distribusi *audio*, dimana pengguna dapat memilih pengeras suara yang diaktifkan. Saat ini, terdapat berbagai cara untuk mendistribusikan suara secara digital seperti interfase I2S. Interfase I2S adalah interfase yang mengubah suara analog menjadi digital dengan menggunakan 3 *signal*, yaitu SCK, WS dan SD. Salah satu cara untuk mengimplementasikan interfase I2S adalah dengan menggunakan perangkat ESP32. Selain mampu mengimplementasikan interfase I2S, Esp32 juga dapat terhubung dengan WiFi, yang memungkinkan pembuatan sistem distribusi suara dengan mengimplementasikan interfase I2S untuk mentransmisikan data suara dalam format digital melalui jaringan WiFi. Namun, proses pengiriman data melalui WiFi akan menimbulkan latensi antara pengiriman dan penerimaan data. Oleh karena itu, diperlukan pengukuran *delay*, *jitter*, dan *packet loss* terhadap data suara digital yang dikirim melalui WiFi.

Sistem distribusi *audio* ini dikembangkan dengan menggunakan bahasa pemrograman C++, *framework* Arduino Core dan menggunakan protokol websocket serta udp sebagai media pengiriman data suara, serta menyediakan fitur untuk memilih pengeras suara yang diaktifkan. Selain mengembangkan sistem distribusi *audio*, tujuan utama dari penelitian ini adalah mengukur performa sistem dengan mengamati parameter *delay*, *jitter*, dan *packet loss*. Dari hasil pengujian yang sudah dilakukan, ketika mengirim data sebesar 16 bits, protokol websocket menghasilkan rata-rata *delay* sebesar 4-6 ms, rata-rata *jitter* sebesar 5-10 ms, dan rata-rata *packet loss* sebesar 0%. Sedangkan protokol udp menghasilkan rata-rata *delay* sebesar 2-4 ms, rata-rata *jitter* sebesar 5-17 ms, dan rata-rata *packet loss* sebesar 0,49-0,64% dengan ukuran data yang sama. Ketika mengirim data sebesar 16 kbits, protokol websocket menghasilkan rata-rata *delay* sebesar 5-10 ms, rata-rata *jitter* sebesar 6-17 ms, dan rata-rata *packet loss* sebesar 0%. Sedangkan protokol udp menghasilkan rata-rata *delay* sebesar 3-6 ms, rata-rata *jitter* sebesar 2-13 ms, dan rata-rata *packet loss* sebesar 0%. Kedua pengujian ini juga menunjukkan bahwa, sistem yang dikembangkan memiliki performa yang kurang stabil. Lalu pengujian I2S menghasilkan rata-rata *delay* sebesar 219,31 ms dan *jitter* sebesar 1,73 ms. Sehingga total *delay* dari sistem yang dikembangkan adalah sebesar 224,31-229,31 ms, *jitter* sebesar 7,73-18,73 ms, dan *packet loss* sebesar 0% pada protokol websocket. Sedangkan protokol udp menghasilkan total *delay* sebesar 22,31-225,31 ms, *jitter* sebesar 3,73-14,73 ms, dan *packet loss* sebesar 0%. Berdasarkan hasil pengujian, protokol udp cenderung lebih cocok digunakan pada sistem yang dikembangkan karena *delay* dan *jitter* yang lebih rendah daripada protokol websocket.

Kata kunci : sistem distribusi *audio*, I2S, ESP32, websocket, udp

ABSTRACT

Alongside the advancement of IoT technology, various supporting technologies for IoT have also rapidly evolved. One of these is the audio distribution system, where users can select activated speakers. Currently, there are several ways to digitally transmit sound, such as the I2S interface. The I2S interface converts analog sound into digital using 3 signals: SCK, WS, and SD. One way to implement the I2S interface is by using the ESP32 device. Besides implementing the I2S interface, ESP32 can also connect to WiFi, enabling the creation of an audio distribution system by implementing the I2S interface to transmit sound data in digital format over a WiFi network. However, transmitting data over WiFi can create a latency between sending and receiving data. Hence, measuring delay, jitter and packet loss for the digitally transmitted audio data over WiFi is necessary.

This audio distribution system is developed using the C++ programming language, Arduino Core framework, employing WebSocket and UDP protocols as media for transmitting audio data, while providing a feature to select the activated speakers. Aside from developing the audio distribution system, the primary aim of this research is to measure system performance by observing delay and jitter parameters. From the conducted tests, when sending 16-bit data, the WebSocket protocol yields an average delay of 4-6 ms, an average jitter of 5-10 ms, and an average packet loss of 0%. In contrast, the UDP protocol produces an average delay of 2-4 ms, an average jitter of 5-17 ms, and an average packet loss of 0.49-0.64% with the same data size. When sending 16 kbit data, the WebSocket protocol results in an average delay of 5-10 ms, an average jitter of 6-17 ms, and an average packet loss of 0%, while the UDP protocol yields an average delay of 3-6 ms, an average jitter of 2-13 ms, and average packet loss of 0%. Both tests indicate the developed system exhibits less stable performance. Furthermore, the I2S test resulted in an average delay of 219.31 ms and an average jitter of 1.73 ms. Consequently, the total delay from the developed system ranges between 224.31-229.31 ms, a jitter of 7.73-18.73 ms, and a packet loss of 0% on the WebSocket protocol. On the other hand, the UDP protocol generates a total delay of 22.31-225.31 ms, a jitter of 3.73-14.73 ms, and a packet loss of 0%. Based on the test results, the UDP protocol seems more suitable for use in the developed system due to lower delay and jitter compared to the WebSocket protocol.

Keywords : *audio distribution system, I2S, ESP32, websocket, udp*