

Tugas #4 Aplikasi TCP UDP Server Client

Daftar Isi

Contents

1	Server UDP dan Client UDP	2
2	Server TCP dan Client TCP	3
3	UDP Server di ESP32	4
4	UDP Client di ESP32	5
5	TCP Server di ESP32	6
6	TCP Client di ESP32	7

Daftar Gambar

Gambar 1 Hasil Eksekusi Program UDP Client	2
Gambar 2 Hasil Eksekusi Program UDP Server	2
Gambar 3 Hasil Eksekusi Program TCP Client	3
Gambar 4 Hasil Eksekusi TCP Server.....	3
Gambar 5 Data yang Dikirim ke Server UDP ESP32 dan Diterima dari Server UDP ESP32	4
Gambar 6 Data yang Dikirim oleh UDP Client ESP32	5
Gambar 7 Data yang Dikirim ke Server TCP ESP32 dan Diterima dari Server TCP ESP32	6
Gambar 8 Data yang Dikirim oleh TCP Client ESP32.....	7

1 Server UDP dan Client UDP

Tahap pelaksanaan tugas ini adalah seperti pada langkah-langkah berikut.

1. Membuat kode UDP *client* dan UDP *server*
2. Menjalankan UDP *server*
3. Menjalankan UDP *client*
4. Mengamati data yang dikirim dan diterima

Kode yang digunakan pada tugas ini ditulis dengan bahasa C. Kode merupakan modifikasi dari kode UDP *server* dan *client* pada tautan https://github.com/papageo-dev/UDP_Server-Client_for_Windows. Perubahan yang dibuat adalah seperti pada poin-poin berikut.

1. Perubahan format penamaan variabel di kode *server* dan *client*
2. Perubahan alamat IP dan port di kode *server* dan *client*
3. Penghapusan pesan sukses di kode *server* dan *client*
4. Perubahan data yang dikirim oleh *client*
5. Perubahan format pencetakan di kode *server* dan *client*
6. Penambahan masukkan alamat IP tujuan di kode *client*

Hasil yang didapatkan adalah seperti pada gambar berikut.

```
D:\Kuliah\s7\el4120-jarkom\tugas4>udp-client.exe 127.0.0.1
Data to send: Current local time and date: Fri Nov 25 23:05:16 2022

D:\Kuliah\s7\el4120-jarkom\tugas4>
```

Gambar 1 Hasil Eksekusi Program UDP Client

```
D:\Kuliah\s7\el4120-jarkom\tugas4>udp-server.exe
Receiving IP(s) used: 0.0.0.0
Receiving port used: 8888
I'm ready to receive data packages. Waiting...

Data Recieved: Current local time and date: Fri Nov 25 23:05:16 2022


```

Gambar 2 Hasil Eksekusi Program UDP Server

Data yang dikirimkan oleh *client* dan diterima oleh *server* adalah data waktu ketika *client* mengirimkan data ke *server*.

2 Server TCP dan Client TCP

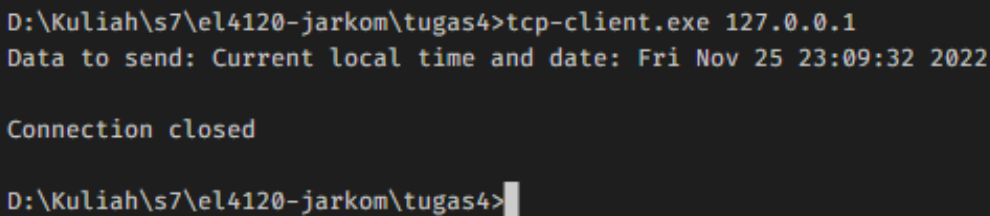
Tahap pelaksanaan tugas ini adalah seperti pada langkah-langkah berikut.

1. Membuat kode TCP *client* dan TCP *server*
2. Menjalankan TCP *server*
3. Menjalankan TCP *client*
4. Mengamati data yang dikirim dan diterima

Kode yang digunakan pada tugas ini ditulis dengan bahasa C. Kode merupakan modifikasi dari kode TCP *server* dan *client* pada tautan <https://learn.microsoft.com/en-us/windows/win32/winsock/finished-server-and-client-code>. Perubahan yang dibuat adalah seperti pada poin-poin berikut.

1. Perubahan format penamaan variabel di kode *server* dan *client*
2. Perubahan alamat IP dan port di kode *server* dan *client*
3. Penghapusan pesan sukses di kode *server* dan *client*
4. Perubahan data yang dikirim oleh *client*
5. Perubahan format pencetakan di kode *server* dan *client*
6. Penambahan masukkan alamat IP tujuan di kode *client*

Hasil yang didapatkan adalah seperti pada gambar berikut.

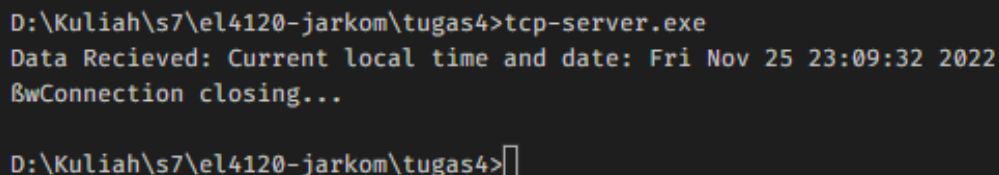


```
D:\Kuliah\s7\el4120-jarkom\tugas4>tcp-client.exe 127.0.0.1
Data to send: Current local time and date: Fri Nov 25 23:09:32 2022

Connection closed

D:\Kuliah\s7\el4120-jarkom\tugas4>
```

Gambar 3 Hasil Eksekusi Program TCP Client



```
D:\Kuliah\s7\el4120-jarkom\tugas4>tcp-server.exe
Data Recieved: Current local time and date: Fri Nov 25 23:09:32 2022
BwConnection closing...

D:\Kuliah\s7\el4120-jarkom\tugas4>
```

Gambar 4 Hasil Eksekusi TCP Server

Data yang dikirimkan oleh *client* dan diterima oleh *server* adalah data waktu ketika *client* mengirimkan data ke *server*.

3 UDP Server di ESP32

Tahap pelaksanaan tugas ini adalah seperti pada langkah-langkah berikut.

1. Membuat rangkaian di ESP32
2. Membuat kode UDP server untuk ESP32
3. Menjalankan kode di ESP32
4. Mengirimkan paket UDP ke ESP32 menggunakan Packet Sender
5. Mengulangi langkah 2 dan 3 untuk keadaan tombol dan potensiometer yang berbeda
6. Mengamati data yang dikirim dan diterima

Kode yang digunakan pada tugas ini ditulis dengan bahasa C++ di Arduino IDE. Kode merupakan modifikasi dari kode UDP server pada tautan <https://www.robotikindonesia.com/2022/08/komunikasi-esp32-soket-udp.html>. Perubahan yang dibuat adalah seperti pada poin-poin berikut.

1. Mengubah ESP32 menjadi WiFi Access Point
2. Mengubah penggunaan pin input (pembacaan potensiometer di GPIO35 dan tombol di GPIO25)
3. Perubahan alamat IP dan port
4. Penghapusan pesan-pesan di serial monitor
5. Perubahan data yang dikirim oleh server

Hasil yang didapatkan adalah seperti pada gambar berikut.

	Time	From IP	From Port	To Address	To Port	Method	Error	ASCII	
👉	23:19:37.107	192.168....	8888	You	58610	UDP		927 0\n\00?\e7\9cU	39 32 37 20 30 0A 00 3F E7 9C 55
👈	23:19:37.082	You	58610	192.168.4.1	8888	UDP		halo dunia	68 61 6c 6f 20 64 75 6e 69 61
👉	23:19:28.786	192.168....	8888	You	58610	UDP		2694 0\n\00?\e7\9cU	32 36 39 34 20 30 0A 00 E7 9C 55
👈	23:19:28.766	You	58610	192.168.4.1	8888	UDP		halo dunia	68 61 6c 6f 20 64 75 6e 69 61
👉	23:19:24.013	192.168....	8888	You	58610	UDP		2807 1\n\00?\e7\9cU	32 38 30 37 20 31 0A 00 E7 9C 55
👈	23:19:24.001	You	58610	192.168.4.1	8888	UDP		halo dunia	68 61 6c 6f 20 64 75 6e 69 61

Gambar 5 Data yang Dikirim ke Server UDP ESP32 dan Diterima dari Server UDP ESP32

Data yang dikirimkan server adalah data nilai analog dari potensiometer dan data nilai digital dari tombol.

4 UDP Client di ESP32

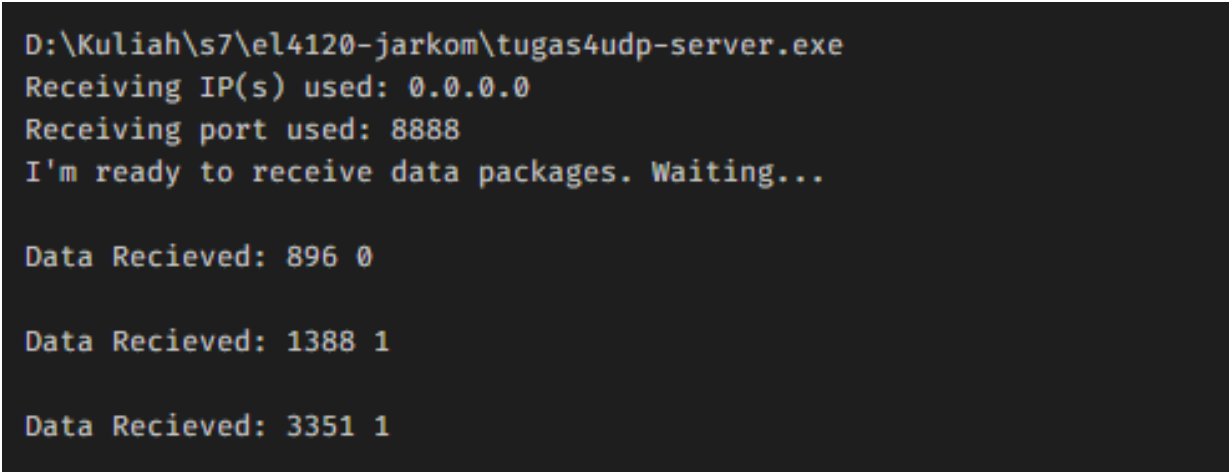
Tahap pelaksanaan tugas ini adalah seperti pada langkah-langkah berikut.

1. Membuat rangkaian di ESP32
2. Membuat kode UDP *client* untuk ESP32
3. Menjalankan kode UDP *server* yang digunakan untuk percobaan sebelumnya
4. Menjalankan kode di ESP32
5. Mengamati data yang dikirim *client* dan diterima oleh *server*

Kode yang digunakan pada tugas ini ditulis dengan bahasa C++ di Arduino IDE. Kode merupakan modifikasi dari kode UDP *server* pada tautan <https://www.robotikindonesia.com/2022/08/komunikasi-esp32-soket-udp.html>. Perubahan yang dibuat adalah seperti pada poin-poin berikut.

1. Mengubah ESP32 menjadi WiFi Access Point
2. Mengubah peran ESP32 menjadi UDP *client*
3. Mengubah penggunaan pin input (pembacaan potensiometer di GPIO35 dan tombol di GPIO25)
4. Perubahan alamat IP dan port
5. Penghapusan pesan-pesan di serial monitor
6. Perubahan data yang dikirim oleh *client*

Hasil yang didapatkan adalah seperti pada gambar berikut.



```
D:\Kuliah\s7\el4120-jarkom\tugas4udp-server.exe
Receiving IP(s) used: 0.0.0.0
Receiving port used: 8888
I'm ready to receive data packages. Waiting...

Data Recieved: 896 0

Data Recieved: 1388 1

Data Recieved: 3351 1
```

Gambar 6 Data yang Dikirim oleh UDP Client ESP32

Data yang dikirimkan *client* adalah data nilai analog dari potensiometer dan data nilai digital dari tombol.

5 TCP Server di ESP32

Tahap pelaksanaan tugas ini adalah seperti pada langkah-langkah berikut.

1. Membuat rangkaian di ESP32
2. Membuat kode TCP *server* untuk ESP32
3. Menjalankan kode di ESP32
4. Mengirimkan paket TCP ke ESP32 menggunakan Packet Sender
5. Mengamati data yang dikirim dan diterima

Kode yang digunakan pada tugas ini ditulis dengan bahasa C++ di Arduino IDE. Kode merupakan modifikasi dari kode TCP *client* pada tautan <https://github.com/espressif/arduino-esp32/blob/master/libraries/WiFi/examples/WiFiClientBasic/WiFiClientBasic.ino>. Perubahan yang dibuat adalah seperti pada poin-poin berikut.

1. Mengubah ESP32 menjadi WiFi Access Point
2. Mengubah peran ESP32 menjadi TCP *server*
3. Mengubah penggunaan pin input (pembacaan potensiometer di GPIO35 dan tombol di GPIO25)
4. Perubahan alamat IP dan port
5. Menghapus penggunaan protokol HTTP
6. Penghapusan pesan-pesan di serial monitor
7. Perubahan data yang dikirim oleh *server*

Hasil yang didapatkan adalah seperti pada gambar berikut.

	Time	From IP	From Port	To Address	To Port	Method	Error	ASCII	
👉	23:24:26.238	192.168....	27015	You	65002	TCP		899 0\n	38 39 39 20 30 0A
👈	23:24:26.234	You	65002	192.168.4.1	27015	TCP		halo dunia	68 61 6c 6f 20 64 75 6e 69 61
👉	23:24:21.974	192.168....	27015	You	53419	TCP		1285 1\n	31 32 38 35 20 31 0A
👈	23:24:21.970	You	53419	192.168.4.1	27015	TCP		halo dunia	68 61 6c 6f 20 64 75 6e 69 61
👉	23:24:13.474	192.168....	27015	You	53416	TCP		3313 1\n	33 33 31 33 20 31 0A
👈	23:24:13.471	You	53416	192.168.4.1	27015	TCP		halo dunia	68 61 6c 6f 20 64 75 6e 69 61

Gambar 7 Data yang Dikirim ke Server TCP ESP32 dan Diterima dari Server TCP ESP32

Data yang dikirimkan *server* adalah data nilai analog dari potensiometer dan data nilai digital dari tombol.

6 TCP Client di ESP32

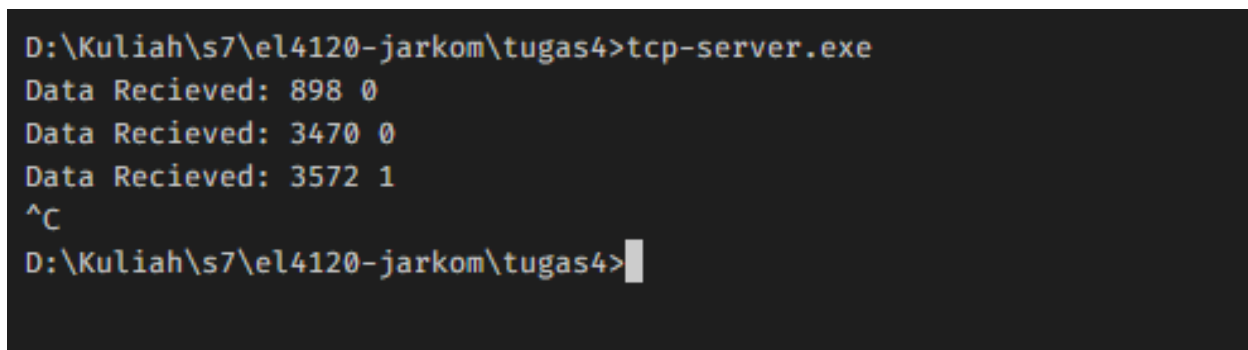
Tahap pelaksanaan tugas ini adalah seperti pada langkah-langkah berikut.

1. Membuat rangkaian di ESP32
2. Membuat kode TCP *client* untuk ESP32
3. Menjalankan TCP *server* yang digunakan pada percobaan sebelumnya
4. Menjalankan kode di ESP32
5. Mengamati data yang dikirim *client* dan diterima *server*

Kode yang digunakan pada tugas ini ditulis dengan bahasa C++ di Arduino IDE. Kode merupakan modifikasi dari kode TCP *client* pada tautan <https://github.com/espressif/arduino-esp32/blob/master/libraries/WiFi/examples/WiFiClientBasic/WiFiClientBasic.ino>. Perubahan yang dibuat adalah seperti pada poin-poin berikut.

1. Mengubah ESP32 menjadi WiFi Access Point
2. Mengubah penggunaan pin input (pembacaan potensiometer di GPIO35 dan tombol di GPIO25)
3. Perubahan alamat IP dan port
4. Menghapus penggunaan protokol HTTP
5. Penghapusan pesan-pesan di serial monitor
6. Perubahan data yang dikirim oleh *client*

Hasil yang didapatkan adalah seperti pada gambar berikut.



```
D:\Kuliah\s7\el4120-jarkom\tugas4>tcp-server.exe
Data Recieved: 898 0
Data Recieved: 3470 0
Data Recieved: 3572 1
^C
D:\Kuliah\s7\el4120-jarkom\tugas4>
```

Gambar 8 Data yang Dikirim oleh TCP Client ESP3

Data yang dikirimkan *client* adalah data nilai analog dari potensiometer dan data nilai digital dari tombol.

Lampiran

Source code yang digunakan dapat diakses di tautan berikut.

<https://github.com/ubbeg2000/tugas4-el4120>