

## **PESAWAT TELEPON PADA JARINGAN WIRED LOCAL AREA NETWORK (LAN) MENGGUNAKAN PROTOKOL TCP/IP**

**Christian Nugraha<sup>1</sup>, Darmawan Utomo<sup>2</sup>, Saptadi Nugroho<sup>3</sup>**

Program Studi Teknik Elektro, Fakultas Teknik Elektro dan Komputer,  
Universitas Kristen Satya Wacana, Salatiga

<sup>1</sup>bejo\_yuk@yahoo.com, <sup>2</sup>du88@yahoo.com, <sup>3</sup>saptadi\_nugroho@yahoo.com

### **INTISARI**

Dalam perancangan pesawat telepon berbasis mikrokontroler dengan arsitektur *Alf-Vegard* RISC (AVR) yang menggunakan jaringan LAN, suara dicuplik dari *microphone* setelah dikuatkan oleh untai penguat *inverting* menggunakan *Analog to Digital Converter* (ADC) dengan frekuensi 8 KHz. Kemudian data pencuplikan 8 bit yang disimpan pada *buffer* berukuran 100 byte diperbesar resolusinya menjadi 16 bit dan dikompresi menggunakan *Adaptive Differential Pulse Code Modulation* (ADPCM). Data untuk setiap 200 pencuplikan dikirimkan ke jaringan menggunakan protokol *User Datagram Protocol* (UDP). Setelah paket data sampai di tujuan, data harus didekompresi dan disuarakan kembali pada *speaker* setelah melewati *Low Pass Filter* menggunakan *Pulse Width Modulation* (PWM). Pesawat telepon yang direalisasikan adalah sebanyak tiga buah. Hasil pengujian menunjukkan untai penguat dan filter sudah bekerja sebagaimana semestinya. Hasil pembacaan paket data teks antar pesawat telepon dengan program uji pada komputer menggunakan Wireshark untuk paket ARP dan UDP sudah sesuai dengan seharusnya. Waktu tunda yang dihasilkan masih memenuhi standar untuk aplikasi suara. Sejumlah 30 responden memiliki tingkat keberhasilan mencocokkan kalimat sebesar 84,44%.

**Kata kunci** : pesawat telepon, LAN, TCP/IP

## 1 PENDAHULUAN

Walaupun hanya berjarak satu ruangan, manusia cenderung untuk tidak beranjak dari tempatnya tetapi tetap dapat saling berkomunikasi. Jaringan LAN yang sudah tersedia dapat dimanfaatkan sebagai jaringan telepon berbasis IP sehingga tidak diperlukan instalasi jaringan telepon yang baru. Sebuah perangkat sistem *embedded* dapat difungsikan untuk menggantikan peran komputer dalam melakukan pemrosesan data suara sehingga konsumsi daya dapat dikurangi.

## 2 DASAR TEORI

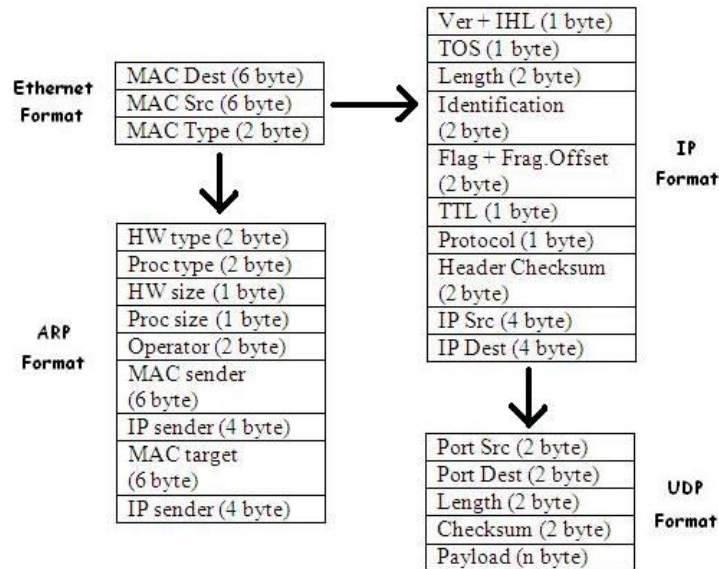
Mikrokontroler dapat dikatakan sebagai sebuah miniatur komputer dalam bentuk IC yang menjalankan sebuah fungsi dengan tujuan tertentu. Terdapat banyak fitur pendukung didalamnya seperti ADC, PWM pada *Timer*, *External Interrupt*, EEPROM dan berbagai antarmuka komunikasi perangkat keras seperti USART, I<sup>2</sup>C dan SPI. Mikrokontroler sering digunakan sebagai perangkat utama dalam sebuah sistem *embedded*.

Agar mikrokontroler dapat berkomunikasi menggunakan jaringan komputer, dibutuhkan sebuah *ethernet controller* yang dapat digunakan untuk mengirim dan menerima paket data dari jaringan LAN. Perangkat ini berfungsi sebagai *Network Interface Card* (NIC) bagi mikrokontroler.

Setelah terhubung secara fisik dengan jaringan LAN, dibutuhkan protokol komunikasi yang sesuai dengan jaringan komputer. Oleh karena itu, protokol UDP yang merupakan bagian dari TCP/IP dipilih sebagai protokol komunikasi karena cukup sederhana untuk diimplementasikan.

**PESAWAT TELEPON PADA JARINGAN WIRED LOCAL AREA NETWORK  
(LAN) MENGGUNAKAN PROTOKOL TCP/IP**

*Christian Nugraha, Darmawan Utomo, Saptadi Nugroho*



Gambar 1 : Format paket data.

Protokol ARP dibutuhkan untuk mendapatkan alamat MAC dari suatu perangkat pada jaringan komputer. Paket *ARP Request* dikirimkan secara *broadcast* dan paket *ARP Reply* dikirimkan secara *unicast*. Format paket yang digunakan adalah Ethernet Format + ARP Format ketika akan mengirimkan paket ARP, sedangkan Ethernet Format + IP Format + UDP Format ketika akan mengirimkan paket UDP seperti ditunjukkan pada Gambar 1.

Sebelum paket data hasil pencuplikan dikirimkan ke jaringan, terlebih dahulu data harus dikompresi menggunakan ADPCM agar jumlah data yang dikirimkan dapat dikurangi. Teknik kompresi ini menyandikan selisih cuplikan saat ini dengan nilai prediksi sebelumnya. Data hasil dekompresi bersifat *lossy* yang berarti tidak akan selalu sesuai dengan data aslinya.

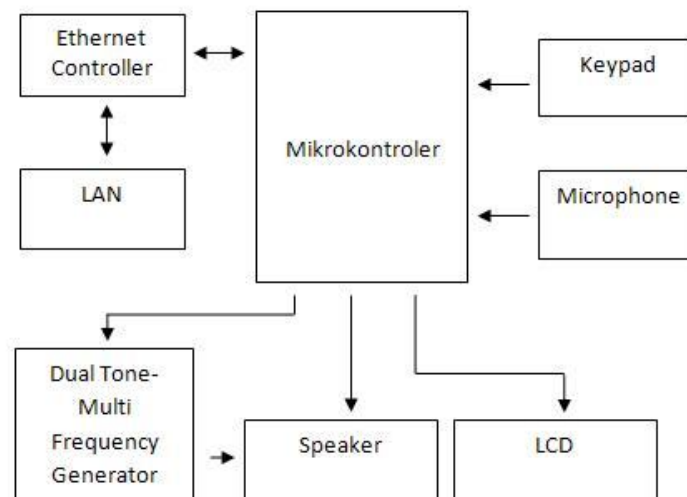
Fitur lainnya yang disediakan pada pesawat telepon adalah *Dual Tone Multi Frequency (DTMF) Generator* yang diberikan dengan tujuan agar pengguna lebih familiar dengan pesawat telepon tersebut. Setiap baris dan kolom diwakili oleh sebuah frekuensi tertentu seperti pada Gambar 2. Ketika sebuah tombol ditekan, maka kombinasi sinyal berdasarkan nilai frekuensi pada baris dan kolom akan dihasilkan. Pada pesawat telepon rumah, DTMF digunakan untuk pengiriman isyarat nomor yang dituju. Pada pesawat telepon ini, DTMF digunakan hanya untuk menyuarakan tombol mana yang ditekan.

1	2	3	A	697
4	5	6	B	770
7	8	9	C	852
*	0	#	D	941
1209	1336	1477	1633	Freq

Gambar 2 : Pemetaan frekuensi DTMF.

### 3 PERANCANGAN SISTEM

ATmega32A dipilih sebagai otak dari sistem karena jumlah pin dan kapasitas EEPROM yang sudah cukup memadai. Pembangkit osilasi dipilih menggunakan nilai maksimal 16 MHz dengan tujuan agar dapat dilakukan percakapan yang *real-time*. Gambar 3 berikut ini menunjukkan blok diagram sistem secara garis besar.



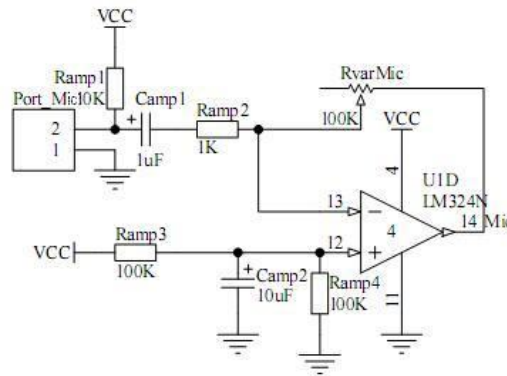
Gambar 3 : Blok diagram sistem.

*Digital switch* (HCF4066BE) pada berfungsi sebagai gerbang yang menentukan sinyal suara dari PWM atau sinyal suara dari DTMF yang akan dikirimkan ke *speaker*. Sedangkan *Logic Level Shifter* (7408) berfungsi mengubah

**PESAWAT TELEPON PADA JARINGAN WIRED LOCAL AREA NETWORK  
(LAN) MENGGUNAKAN PROTOKOL TCP/IP**

*Christian Nugraha, Darmawan Utomo, Saptadi Nugroho*

nilai tegangan high dari *ethernet controller* (EMS Ethernet Module) yang adalah sebesar 3 V menjadi 5 V agar dapat dibaca oleh mikrokontroler.



Gambar 4 : Untai penguat *microphone*.

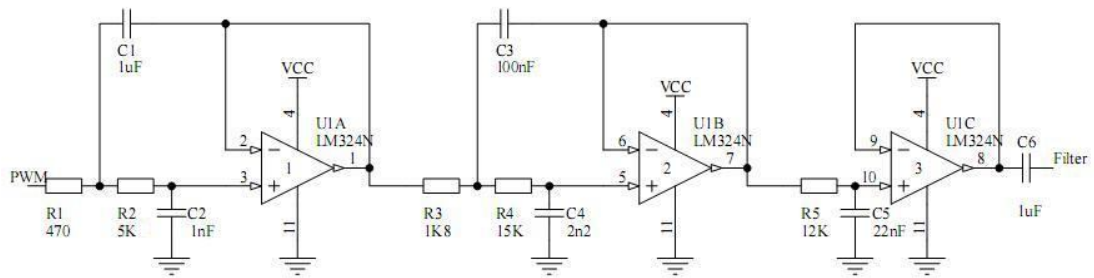
Untuk menguatkan sinyal dari *microphone*, digunakan untai dengan Op-Amp sebagai penguat sinyal seperti ditunjukkan pada Gambar 4. Besarnya penguatan tergantung dari nilai  $R_{var}$  saat itu dan dapat dihitung berdasarkan Persamaan (1) berikut :

$$A_v = - \frac{R_{var}}{R_{amp2}} \quad (1)$$

Diberikan juga nilai *offset* sebesar 2,5 V untuk menyesuaikan nilai tengah dari rentang ADC yang adalah 0 – 5 V. Besarnya nilai *offset* dapat dihitung menggunakan Persamaan (2) sebagai berikut :

$$V_{offs} = \frac{R_{amp4}}{R_{amp4} + R_{amp2}} \quad (2)$$

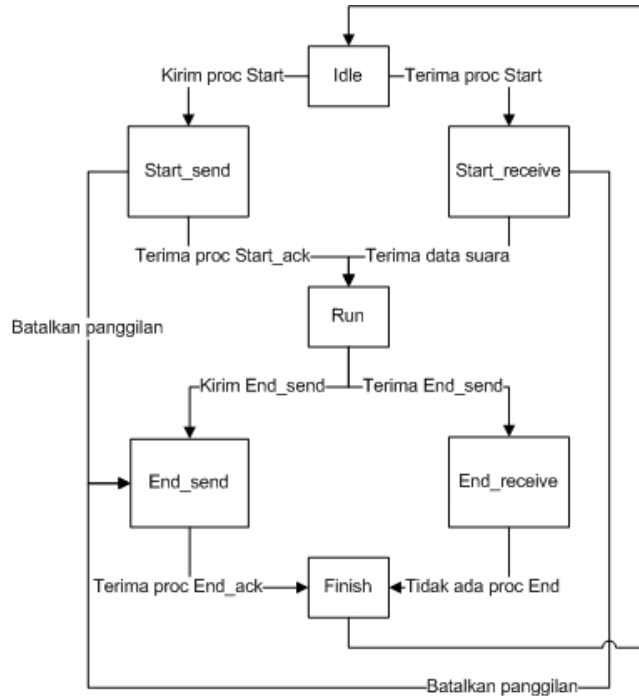
Pada sisi *speaker*, agar kualitas suara yang dihasilkan menjadi lebih baik sebuah *Low Pass Filter* orde 5 dengan frekuensi *cut-off* 3,4 KHz diberikan. Filter Chebyshev dengan *passband ripple* sebesar 3 dB pada Gambar 5 menggunakan topologi Sallen-Key pada kondisi penguatan sebesar 1.



Gambar 5 : Low Pass Filter orde 5 dengan frekuensi cut-off 3,4 KHz [1].

Selanjutnya adalah perancangan bagian perangkat lunak dalam bahasa C menggunakan *compiler* Codevision AVR 2.04.4a. Diawali dengan inisialisasi pin-pin mikrokontroler beserta fitur-fiturnya dan dilanjutkan dengan inisialisasi pada *ethernet controller* agar dapat bekerja sesuai dengan kebutuhan.

Agar proses melakukan panggilan dan menerima panggilan dapat berjalan dengan baik, diperlukan suatu prosedur atau aturan agar tidak terjadi kesalahan dalam pengiriman dan penerimaan data seperti tertera pada Gambar 6.



Gambar 6 : Prosedur panggilan pada pesawat telepon.

Setiap keadaan pada prosedur pesawat telepon selalu diawali dengan pengecekan data dari jaringan dan dilakukan secara terus menerus. Jika terdapat *ARP Request*

**PESAWAT TELEPON PADA JARINGAN WIRED LOCAL AREA NETWORK  
(LAN) MENGGUNAKAN PROTOKOL TCP/IP**

*Christian Nugraha, Darmawan Utomo, Saptadi Nugroho*

---

yang datang, maka secara otomatis *ARP Reply* akan diberikan. Jika terdapat paket UDP yang datang, akan diproses sesuai dengan keadaan saat itu. Setiap kali pengguna melakukan panggilan, *ARP Request* akan dikirimkan ke jaringan terlebih dahulu untuk mendapatkan alamat MAC lawan bicara. Setelah itu prosedur pertama untuk melakukan panggilan akan dijalankan.

Yang paling utama dari prosedur di atas adalah ketika terjadi percakapan. Proses pemutaran suara dan pencuplikan dilakukan secara bergantian dan berkesinambungan pada 2 buah *buffer* untuk pemutaran dan 2 buah *buffer* untuk pencuplikan. Penjadwalannya diatur oleh *Timer* yang akan terjadi ketika tercapai kondisi *overflow* dengan frekuensi sebesar 8 KHz.

Ketika sebuah paket suara yang berukuran 100 byte tiba, data tersebut harus didekompresi terlebih dahulu menjadi 200 byte. Data suara tersebut disimpan pada sebuah *buffer* pemutaran dan akan dimainkan selama 25 ms. Jika *buffer* pertama belum habis dimainkan dan ada paket suara yang datang, maka data tersebut akan disimpan di *buffer* pemutaran yang lainnya. Hal yang serupa juga dilakukan pada proses pencuplikan. Tiap hasil pencuplikan disimpan pada *buffer* pencuplikan. Setelah mencapai kapasitas maksimum 200 byte, hasil pencuplikan akan disimpan pada *buffer* pencuplikan yang lainnya. Sedangkan data pada *buffer* pencuplikan yang sudah penuh siap dikirimkan ke jaringan setelah dikompresi terlebih dahulu.

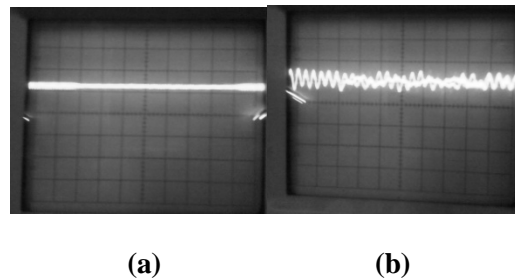
Proses kompresi dan dekompresi pada ADPCM menggunakan referensi dari *Interactive Multimedia Association* (IMA) karena tidak membutuhkan nilai pecahan dalam prosesnya sehingga dapat meringankan beban kerja mikrokontroler [2]. Pada dasarnya, masukan untuk proses kompresi harus data suara berukuran 16 bit dan keluaran akan berukuran 4 bit. Akan tetapi karena hasil pencuplikan berukuran 8 bit maka data suara harus diubah menjadi 16 bit. Penyesuaian dilakukan dengan cara menggeser data sebanyak 8 kali ke arah kiri. Oleh karena ukuran data suara asli hanya sebesar 8 bit dan keluaran berukuran 4 bit, maka jumlah data yang dapat direduksi hanya sebesar 50%.

Fitur lainnya yang disediakan oleh pesawat telepon ini adalah *speed dial* dan *log*. Dimana dengan menggunakan *speed dial*, pengguna cukup menekan urutan nomor lawan bicara yang datanya tersimpan di EEPROM. Jumlah maksimum yang dapat disimpan pada pesawat telepon ini adalah sebanyak 20 daftar. Sedangkan *log*

berfungsi untuk menyimpan data tentang 50 panggilan terakhir yang dilakukan dan juga termasuk panggilan tak terjawab.

#### 4 PENGUJIAN

Pengukuran terhadap keberhasilan perancangan dilihat dari berbagai aspek. Pengukuran nilai SNR, pengujian untai penguat *microphone*, pengujian terhadap rangkaian *Low Pass Filter*, pengujian komunikasi serial dengan DTMF, pengujian implementasi TCP/IP, pengujian implementasi ADPCM, pengujian protokol pesawat telepon, pengukuran waktu tunda, pengujian kualitas percakapan secara subjektif.



Gambar 7 : Pengukuran nilai SNR pada kondisi tanpa masukan(a) dan dengan masukan(b).

Pada kondisi tanpa masukan, terlihat di osiloskop (0,2V/Div-1ms/Div) sinyal derau sebesar 80 mV seperti pada Gambar 7a. Sedangkan pada kondisi pengguna mengeluarkan suara, dihasilkan sinyal keluaran sebesar 200 mV. Berdasarkan angka tersebut maka nilai dari SNR pada kondisi tersebut adalah 20,0823 dB.

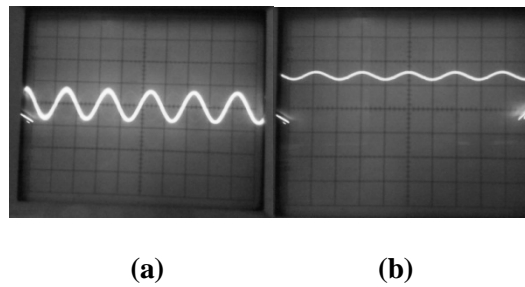
Pengujian untuk untai penguat *microphone* dilakukan dengan memberikan sinyal masukan sebesar 300 mVpp seperti pada Gambar 8a (0,2V/Div-0,5ms/Div). Didapatkan hasil keluaran sebesar 800 mVpp seperti pada Gambar 8b (2V/Div-0,5ms/Div). Sehingga pada kondisi tersebut dapat disimpulkan penguatan yang terjadi adalah sebesar 2,6 kali.



**PESAWAT TELEPON PADA JARINGAN WIRED LOCAL AREA NETWORK  
(LAN) MENGGUNAKAN PROTOKOL TCP/IP**

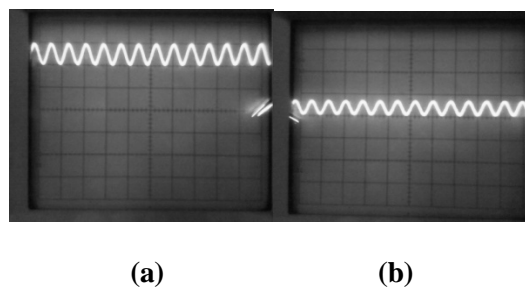
*Christian Nugraha, Darmawan Utomo, Saptadi Nugroho*

---



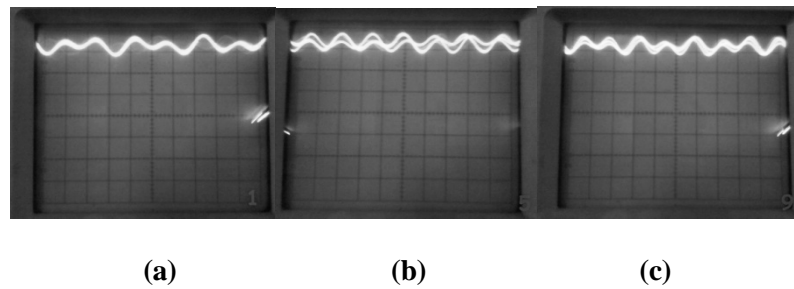
Gambar 8 : Sinyal masukan sebesar 300 mVpp(a) dan keluaran sebesar 800 mVpp(b).

Pengujian berikutnya adalah untuk rangkaian *Low Pass Filter*. Filter diberikan masukan sebesar 1 Vpp seperti pada Gambar 9a. Pada saat frekuensi dinaikan menjadi 3,1 KHz didapatkan amplitudo sebesar 0,7 Vpp yang menandakan frekuensi *cut-off* telah tercapai seperti pada Gambar 9b. Terdapat selisih nilai pada frekuensi *cut-off* sebesar 300 Hz terhadap bagian perancangan dikarenakan kondisi komponen yang tidak ideal.



Gambar 9 : Sinyal masukan sebesar 1 Vpp(a) dan nilai keluaran sebesar 700 mVpp pada 3,1 KHz(b).

Selanjutnya adalah pengujian terhadap keluaran dari *DTMF Generator*. Dilakukan dengan cara pengiriman nilai tombol secara serial. Sinyal keluaran kemudian ditampilkan pada osiloskop seperti pada Gambar 10 berikut ini.



Gambar 10 : Hasil keluaran DTMF untuk angka 1(a), angka 5(b) dan angka 9(c).

Untuk pengujian implementasi TCP/IP, Unit diatur menggunakan alamat IP 192.168.1.1 dengan alamat MAC 22:22:22:22:22:22. Sedangkan komputer menggunakan alamat IP 192.168.1.113 dengan alamat MAC 00:1B:38:F6:57:B3. Pengujian pertama dengan unit mengirimkan ARP Request ke komputer sebanyak 3 kali seperti pada Gambar 11.

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	22:22:22:22:22:22	Broadcast	ARP	64	who has 192.168.1.113? Tell 192.168.1.1
2	0.000015	CompalIn_f6:57:b3	22:22:22:22:22:22	ARP	42	192.168.1.113 is at 00:1b:38:f6:57:b3
3	0.000066	22:22:22:22:22:22	Broadcast	ARP	64	who has 192.168.1.113? Tell 192.168.1.1
4	0.000071	CompalIn_f6:57:b3	22:22:22:22:22:22	ARP	42	192.168.1.113 is at 00:1b:38:f6:57:b3
5	0.000134	22:22:22:22:22:22	Broadcast	ARP	64	who has 192.168.1.113? Tell 192.168.1.1
6	0.000139	CompalIn_f6:57:b3	22:22:22:22:22:22	ARP	42	192.168.1.113 is at 00:1b:38:f6:57:b3

Frame 1: 64 bytes on wire (512 bits), 64 bytes captured (512 bits)  
 Ethernet II, Src: 22:22:22:22:22:22 (22:22:22:22:22:22), Dst: Broadcast (ff:ff:ff:ff:ff:ff)  
 Address Resolution Protocol (request)

Gambar 11 : Hasil capture ARP pada Wireshark.

Pengujian TCP/IP berikutnya adalah pengiriman dan penerimaan data berupa teks yang dikirimkan dari dan ke komputer oleh Unit menggunakan program uji yang dibangun menggunakan Visual Studio 2005. Hasil pembacaan tersebut ditunjukkan pada Gambar 12 dan Gambar 13 berikut ini.

13 389.848927		192.168.1.113	192.168.1.1	UDP	54	Source port: gmrupdateserv	Destination port: cbt	
Frame 13: 54 bytes on wire (432 bits), 54 bytes captured (432 bits) Ethernet II, Src: CompalIn_f6:57:b3 (00:1b:38:f6:57:b3), Dst: 22:22:22:22:22:22 (22:22:22:22:22:22) Internet Protocol Version 4, Src: 192.168.1.113 (192.168.1.113), Dst: 192.168.1.1 (192.168.1.1) User Datagram Protocol, Src Port: gmrupdateserv (1070), Dst Port: cbt (7777) Data (12 bytes)								
0000	22	22	22	22	22	00	1b 38 f6 57 b3 08 00 45 00	..... 8.w...E.
0010	00	28	02	5a	00	00	80 11 b4 a8 c0 a8 01 71 c0 a8	.(Z.... ..q..
0020	01	01	04	2e	1e	61	00 14 94 60 f0 09 00 39 38 37	..... .987
0030	36	35	34	33	32	31		654321

Gambar 12 : Hasil capture Wireshark pada pengiriman teks ke Unit.

# PESAWAT TELEPON PADA JARINGAN WIRED LOCAL AREA NETWORK (LAN) MENGGUNAKAN PROTOKOL TCP/IP

Christian Nugraha, Darmawan Utomo, Saptadi Nugroho

The screenshot shows a network traffic capture in Wireshark. The top section displays a list of captured packets:

No.	Time	Source	Destination	Protocol	Length	Info
9	817.165647	22:22:22:22:22:22	Broadcast	ARP	64	who has 192.168.1.113? Tell 192.168.1.1
10	817.165662	CompalIn_f6:57:b3	22:22:22:22:22:22	ARP	42	192.168.1.113 is at 00:1b:38:f6:57:b3
11	850.276902	192.168.1.1	192.168.1.113	UDP	64	Source port: bmc-patrolldb Destination port: cbt

The bottom section shows the details of the selected packet (Frame 11):

- Frame 11: 64 bytes on wire (512 bits), 64 bytes captured (512 bits)
- Ethernet II, Src: 22:22:22:22:22:22 (22:22:22:22:22:22), Dst: CompalIn\_f6:57:b3 (00:1b:38:f6:57:b3)
- Internet Protocol Version 4, Src: 192.168.1.1 (192.168.1.1), Dst: 192.168.1.113 (192.168.1.113)
- User Datagram Protocol, Src Port: bmc-patrolldb (1313), Dst Port: cbt (7777)
- Data (12 bytes)

The raw data section shows the hexadecimal and ASCII representation of the data:

```
0000 00 1b 38 f6 57 b3 22 22 22 22 22 08 00 45 00  ..8.W."" """"..E.
0010 00 28 00 00 40 00 80 11 77 02 c0 a8 01 01 c0 a8  (.@... w.....
0020 01 71 05 21 1e 61 00 14 00 00 f0 09 00 31 32 33  .g!.a...123
0030 34 35 36 37 38 39 00 00 00 00 00 00 00 00 456789.. .....
```

Gambar 13 : Hasil *capture* Wireshark pada penerimaan teks oleh komputer.

Untuk pengujian pada implementasi ADPCM dilakukan dengan mengambil sejumlah data secara berurutan dari sebuah *file* .WAV berisi *speech* dengan resolusi 8 bit dan frekuensi pencuplikan 8 KHz. Data yang diambil sebanyak 100 buah secara berurutan dan memiliki perubahan sinyal yang dinamis. Dari 100 data tersebut, hasil dekompresi menunjukkan 3 buah data yang memiliki selisih nilai lebih besar dari 15 desimal (8 bit) bila dibandingkan dengan nilai aslinya. Sehingga dapat dikatakan persentase error kompresi adalah sebesar 3%.

Selanjutnya adalah pengujian untuk protokol pesawat telepon yang dibangun. Pengujian dilakukan dengan cara menggunakan 2 unit yang diatur pada kondisi berbeda. Unit 1 diatur pada kondisi normal atau dengan kata lain protokol otomatis. Sedangkan Unit 2 diatur pada kondisi manual atau pengiriman protokol berdasarkan masukan pada tombol oleh pengguna. Hasil yang didapatkan adalah Unit 1 dapat merespon sebagaimana seharusnya terhadap masukan yang diberikan oleh Unit 2.

Berikut adalah pengukuran untuk waktu tunda. Pengukuran dilakukan dengan cara menggunakan *Timer* sebagai acuan waktu pada Unit 1. Ketika Unit 1 memiliki paket suara mentah yang siap dikirimkan, *Timer* diaktifkan. Setelah itu paket diproses dan dikirimkan ke Unit 2. Setelah paket sampai di Unit 2, paket diproses dan disimpan di *buffer*. Kemudian Unit 2 memproses kembali data pada *buffer* tersebut untuk dikirimkan kembali ke Unit 1. Setelah Unit 1 menerima dan selesai memproses paket suara tersebut, *Timer* dinonaktifkan saat itu juga. Setiap kali *Timer* mencapai *overflow*, nilai variabel penghitung ditambah satu dengan tujuan untuk menghitung berapa kali *Timer* mencapai *overflow*. Kemudian nilai tersebut dikalikan dengan periode untuk 1 kali *overflow*. Hasil yang didapat adalah 4,672 ms untuk sekali transmisi paket suara dari kondisi suara siap dikirimkan hingga suara siap dimainkan. Berdasarkan dokumen referensi G.114 dari *International*

*Telecommunication Union* [3, h. 2] waktu tunda pada pesawat telepon ini masih lebih kecil dari standar maksimal waktu tunda yang diijinkan dalam sebuah aplikasi suara.

Pengujian yang terakhir adalah kualitas percakapan secara subjektif dari sejumlah responden. Sejumlah 30 responden melakukan pengujian dalam hal mencocokkan 3 kalimat yang dibacakan oleh 1 orang yang sama. Hasil pengujian menunjukkan presentase keberhasilan responden dalam mencocokkan kalimat adalah sebesar 84,44%. Responden juga memberikan opini tentang kualitas pesawat telepon menggunakan angka 1 sampai dengan 5. Hasil rata-rata yang didapatkan adalah sebesar 3,13.

## **5 KESIMPULAN**

Berdasarkan proses perancangan dan pengujian dari pesawat telepon ini dapat disimpulkan sebagai berikut :

1. Pesawat telepon pada Jaringan Wired LAN menggunakan protokol TCP/IP telah berhasil direalisasikan.
2. Dalam ruang lingkup LAN, kinerja protokol UDP sudah cukup memuaskan.
3. Teknik kompresi ADPCM referensi IMA dapat diimplementasikan pada mikrokontroler 8 bit karena tidak membutuhkan proses komputasi yang rumit.
4. Waktu tunda yang besar sangat berpengaruh terhadap kualitas suara sehingga algoritma harus dibuat sesederhana mungkin ketika melakukan percakapan.
5. Dibutuhkan algoritma penjadwalan yang baik agar sistem dapat bekerja secara berkesinambungan

## **DAFTAR PUSTAKA**

- [1] “AVR335 : *Digital Sound Recorder with AVR and Data Flash*”, Atmel Corporation, 2005.
- [2] Richey,Rodger, “*Adaptive Differential Pulse Code Modulation using PICmicro<sup>TM</sup> Microcontrollers (AN643)*”, Microchip Technology Inc, 1997.
- [3] “*Transmission Systems and Media, Digital Systems and Networks*”, ITU-T Recommendation G.114, 2003.