

# ANALISA AUDIO WATERMARKING BERBASIS TEKNIK REPLICA MODULATIONS MENGGUNAKAN METODE PHASE SHIFTING REPLICA

## AUDIO WATERMARKING ANALISYS BASED ON REPLICA MODULATIONS TECHNIQUE USING PHASE SHIFTING REPLICA METHOD

Saddan Hussein

Prodi S1 Teknik Telekomunikasi, Fakultas Teknik Elektro, Universitas Telkom

Email: [android106@gmail.com](mailto:android106@gmail.com)

### Abstrak

*Digital watermarking* adalah teknik penyembunyian data kedalam data lain sebagai host file dengan memanfaatkan kelemahan pada indera manusia, seperti pendengaran pada pengolahan sinyal *audio*. Pada jurnal ini, akan dikaji sedikit tentang *digital watermarking* pada audio menggunakan metode *Phase Coding*. Metode yang diusulkan tersebut akan diterapkan pada mode *ambient* yaitu kondisi ketika audio yang sudah disisipkan informasi diputar dan direkam kembali dengan menggunakan alat yang berbeda. Lalu, hasil rekaman audio tersebut lah yang dianalisa untuk didapatkan informasi yang telah disisipkan sebelumnya, berapa nilai BER yang dihasilkan, serta bagaimana sinkronisasi yang dihasilkan.

**Kata Kunci :** *Digital watermarking, Audio, Phase Coding, Mode Ambient.*

### Abstract

*Digital watermarking* is a technique for data hiding into another set of data as a host file by utilizing the weakness in human sense, such as hearing in audio signal processing. In this journal, will examine a brief of digital watermarking in audio using Phase Coding method. The proposed method will be applied in ambient mode which is conditions that the audio that has been embedded by information will be played and recorded with another device. Then, that recorded audio will be analyze to get the information that has been embedded before, BER value, and also the synchronization that has been produced.

**Keywords :** Digital watermarking, Audio, Phase Coding, Ambient Mode.

## 1. Pendahuluan

Perkembangan dan pertumbuhan yang pesat pada jaringan internet telah menciptakan tantangan baru untuk melindungi data digital dari pembajakan [1]. Masalah yang sering terjadi umumnya dialami oleh perindustrian dalam bidang multimedia maupun pada per-orangan yaitu kasus pembajakan hak cipta (*copyright piracy*) dan kepemilikan. Karena mudahnya pembajakan yang dilakukan pada data digital hanya dengan cukup meng-copy paste saja data yang diinginkan, maka masalah pembajakan semakin marak dilakukan. *Digital Watermarking* memberikan sebuah solusi ketahanan yang sangat handal untuk masalah tersebut.

*Digital watermarking* adalah sebuah teknik dimana sebuah informasi disisipkan kedalam sebuah set *host-file* (gambar, video, audio, dll) dengan cara tertentu sehingga informasi tersebut tidak mengganggu penggunaan *host-file* secara normal dan tidak dapat dilepaskan dengan cara yang biasa [2]. Sebagai contoh, jika sebuah informasi tersebut disisipkan kedalam audio atau musik, maka telinga tidak dapat mendengar informasi tersebut. Sehingga pada teknologi ini dikenal suatu persyaratan bahwa watermark harus *imperceptible* atau tidak terdeteksi oleh indera penglihatan (*human visual system /HVS*) atau indera pendengaran (*human auditory system /HAS*).

Seperti yang telah dijelaskan diatas, penyembunyian informasi pada audio memanfaatkan kelemahan HAS dikenal juga dengan istilah *audio masking*. Didalam suatu sinyal yang kuat (masker), suatu sinyal yang lemah mungkin tidak terdengar, tergantung pada karakteristik kedua jenis sinyal tersebut [2]. Ini juga menjadi salah satu syarat pada audio watermarking yang dikenal juga sebagai *perceptual transparency*. Syarat utama lainnya adalah *robustness* (ketahanan), *security* (keamanan), *data rate*, *verification and reliability* (verifikasi dan terpercaya) akan dijelaskan pada point berikutnya.

Pada tugas akhir ini, akan menganalisa tentang audio watermarking menggunakan metode *Phase Coding* dengan menggunakan metode *foldng* untuk sinkronisasinya ketika dalam keadaan *ambient mode* (direkam kembali menggunakan alat yang berbeda)..

Watermarking pada audio memiliki tantangan tersendiri dibandingkan dengan gambar atau video, ini karena HAS bekerja pada range frekuensi yang lebar [1]. Masalah ini menjadi salah satu dasar pada tugas akhir ini untuk mengimplementasikan audio watermarking menggunakan metode *Phase Coding*. Dan yang menjadi masalah

utama ketika audio yang sudah disisipkan informasi tersebut direkam ulang menggunakan alat yang berbeda, (*ambient mode*) apakah informasi yang disisipkan masih utuh atau tidak.

## 2. Phase Coding

Phase coding bekerja berdasarkan karakteristik sistem pendengaran manusia yang mengabaikan suara yang lebih lemah jika dua suara itu datang secara bersamaan. Dasar dibalik teknik phase coding adalah menyembunyikan data dengan cara menukarkan fasa asli segmen inisial dari sinyal suara dengan fasa absolut dari sinyal watermark dengan tetap menjaga fasa relative antara segmen sinyal menggunakan beda fasa segmen dari sinyal asli. Ketika beda fasa antara sinyal asli dan sinyal yang dimodifikasi besarnya kecil, maka perbedaan suara yang dihasilkan tidak terdeteksi oleh pendengaran manusia. Phase coding termasuk dalam kelompok teknik audio watermarking berbasis domain frekuensi yang bekerja dengan cara mengubah spectral content dalam domain phase coding. Phase coding merupakan metode paling efektif dari segi perbandingan noise signal-to-perceived. Dalam pembahasan berikutnya akan dijelaskan mengenai parameter apa saja yang mempengaruhi phase coding, bagaimana cara menyisipkan watermark ke dalam data audio (*embedding*), dan cara pengekstraksian watermark dari data audio yang telah disisipi (*extraction*) dengan menggunakan teknik phase coding ini.

### 2.1 Parameter [5]

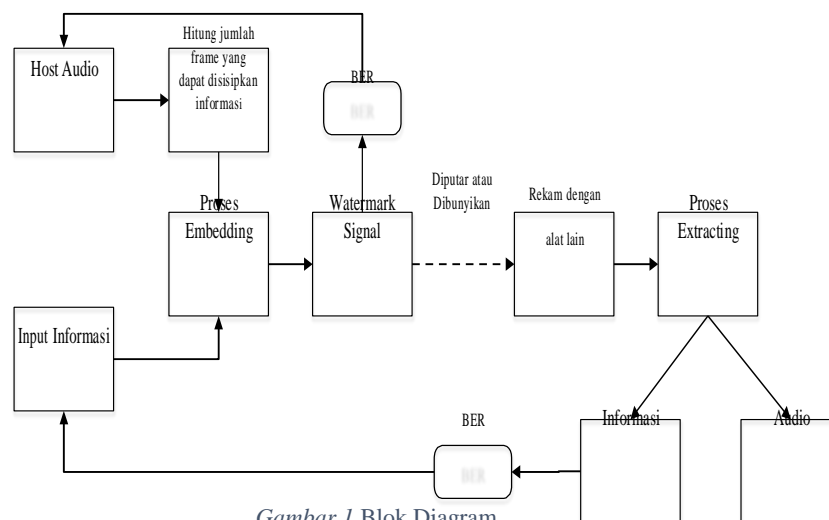
Terdapat beberapa parameter yang berpengaruh dalam teknik watermarking phase coding. Parameter tersebut antara lain fasa relatif dan amplitudo sinyal suara, fasa absolut data watermark, serta beda fasa relatif antara segmen sinyal suara.

Phase coding didasarkan pada kenyataan bahwa pendengaran manusia lebih peka terhadap beda fasa relatif dalam sinyal audio dibandingkan dengan fasa absolut yang ditambahkan. Jika hubungan fasa antar setiap komponen frekuensi diubah secara dramatis, akan terjadi dispersi fasa yang tampak dengan jelas. Akan tetapi, selama modifikasi fasa cukup kecil (tergantung pada pengamat) dan dengan menjaga beda fasa relatif antara segmen-segmen sinyal suara, maka modifikasi fasa yang dilakukan tidak akan terdengar.

Amplitudo sinyal suara digunakan bersama dengan nilai fasa sinyal suara untuk mengubah kembali sinyal suara dari domain frekuensi menjadi domain waktu sehingga dapat didengarkan.

### 2.2 Perancangan Sistem Secara Umum

*Audio watermarking* terdiri dari 2 jenis tahapan penting yaitu, *embedding* (proses penyisipan) dan *extracting* (proses pengambilan informasi) kedalam sebuah file host yang berupa audio. Namun dalam tugas akhir kali ini digunakan metode *phase coding* di *ambient mode* (ketika direkam kembali menggunakan alat yang berbeda). Sehingga akan terdapat berbagai macam jenis tahapan yang dilalui selain 2 jenis proses diatas. Tahapan – tahapan yang dilalui tersebut secara umum dapat dilihat pada diagram berikut :



Gambar 1 Blok Diagram

Pada gambar diatas, sebelum melakukan proses *embedding*, *host audio* yang diinputkan terlebih dahulu dilakukan proses *framing* untuk mengetahui jumlah *frame* yang dapat diinputkan informasi. Setelah didapatkan jumlah *frame* yang disisipkan informasi, maka dilakukan proses *embedding* yang di jelaskan dibawah ini. Setelah melalui proses *embedding* maka akan dihasilkan sebuah *watermark signal* atau yang disebut juga *watermark audio*. Kemudian *watermark audio* tersebut di cek terlebih dahulu nilai BERnya

apakan sudah 0% atau tinggi untuk menghasilkan kualitas audio yang baik sebelum di masukkan kedalam ambient mode. Jika sudah didapatkan nilai BER yang diinginkan, maka dapat dilanjutkan pada tahapan berikutnya. Sedangkan jika belum mencapai tingkat akurasi yang diinginkan, maka masih ada yang salah pada proses *embedding*.

Langkah selanjutnya, watermark audio tersebut dibunyikan pada pc, dan kemudian direkam kembali menggunakan alat berbeda, dalam pengujian ini digunakan sebuah smartphone untuk merekam audio yang dibunyikan tersebut. Hasil rekaman dari smartphone tersebut kemudian dimasukkan kembali kedalam pc untuk dilanjutkan ke dalam proses extracting untuk memperoleh informasi awal yang dikirimkan. Setelah didapatkan informasi yang dikirimkan, akan dicek kembali nilai BER nya untuk mengetahui apakah informasi yang diterima sama seperti yang dikirimkan.

**2.3 Proses Penyisipan Informasi [6]**

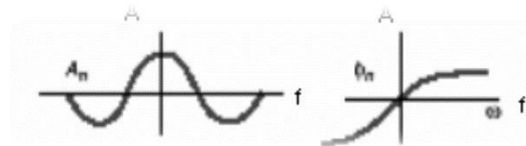
Langkah – langkah pada proses penyisipan watermark menggunakan teknik phase coding adalah :

- a) Bagi urutan suara menjadi N segmen (frame),  $s[i]$ ,  $0 \leq i \leq L-1$ , dimana setiap segmen memiliki panjang yang sama yaitu sebesar L (Gambar 2) frekuensi dari sinyal.



Gambar 2 Gambar Sinyal dibagi menjadi beberapa segmen

- b) Hitung DFT (dalam hal ini menggunakan FFT) pada masing – masing segmen. Hasil dari perhitungan ini adalah berupa  $X(k)$  untuk masing-masing segmen dimana  $0 \leq k \leq L-1$ .
- c) Hitung nilai fasa  $\phi$  dan amplitude A untuk setiap segmen dimana  $a_k$  adalah bagian real dari nilai FFT dan  $b_k$  adalah nilai imajiner. Hasil yang diperoleh akan seperti pada gambar 3 berikut ini.



Gambar 3 Magnitude dan Plot fasa FFT

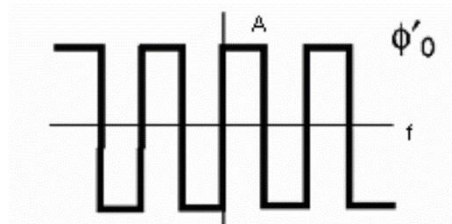
$$|X(k)| = \sqrt{a_k^2 + b_k^2} \tag{2.1}$$

$$\phi = \tan^{-1} \left( \frac{b_k}{a_k} \right) \tag{2.2}$$

- d) Setelah itu hitung beda fasa antar segmen yang berdekatan

$$\Delta\phi_{k-1}(k) = \phi_{k-1}(k) - \phi_k(k) \tag{2.3}$$

- e) Fasa absolut dari sinyal data watermark ditambahkan kedalam beda fasa yang dihasilkan. Sinyal watermark dengan panjang  $L_w$ ,  $w[j]$ ,  $0 \leq j \leq L_w-1$ , disajikan sebagai  $\Phi_{data} = \pi/2$  atau  $\Phi_{data} = -\pi/2$  yang merepresentasikan bit1 atau 0. Hasil yang diperoleh akan tampak seperti Gambar 4.



Gambar 4 Fasa Absolut

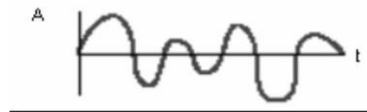
- f) Subtitusikan fasa segmen awal  $\Phi'_0$  dengan fasa sinyal watermark  $\Phi'_{data}$ .

$$\Phi'_0 = \Phi'_{data} \quad (2.4)$$

- g) Buat matriks fasa untuk  $N > 0$  dengan menggunakan beda fasa antar segmen. Hal ini dilakukan untuk menjaga kesinambungan sinyal setelah proses modifikasi fasa segmen awal A.

$$\begin{aligned} \phi'_1(\omega) &= \phi'_0(\omega) + \Delta\phi_1(\omega) \\ &\dots \\ \phi'_n(\omega) &= \phi'_{n-1}(\omega) + \Delta\phi_n(\omega) \\ &\dots \\ \phi'_{(n)}(\omega) &= \phi'_{(n-1)}(\omega) + \Delta\phi_{(n)}(\omega) \end{aligned} \quad (2.5)$$

- h) Gabungkan segmen-segmen yang telah dimodifikasi fasanya tersebut menjadi satu.  
i) Hitung kembali nilai fasa yang baru dan nilai amplitudo yang sudah dihitung sebelumnya untuk melakukan inverse DFT terhadap masing – masing segmen untuk mengembalikan sinyal ke domain waktu (Gambar 5).



Gambar 5 Sinyal watermark yang dihasilkan dengan teknik *phase coding*

#### 2.4 Proses Ekstaksi dan Ambient Mode [6]

Pada proses peng-ekstraksian informasi atau pengambilan informasi kembali, terlebih dahulu sinyal watermark harus melalui proses ambient mode. Ambient Mode adalah mode dimana sinyal watermark di rekam kembali menggunakan alat yang berbeda seperti smartphone. Setelah melalui tahapan ambient mode tersebut, maka sinyal watermark akan siap untuk di ekstraksi. Ekstraksi dilakukan dengan melakukan sinkronisasi terhadap proses *embedding*. Panjang segmen dan panjang *watermark* harus diketahui. Proses ekstraksi memerlukan sinyal suara asli untuk melakukan pendeteksian. Langkah-langkah pendeteksian watermark yang dilakukan adalah sebagai berikut :

- Ambil  $n$  bagian pertama dari sinyal suara dimana  $n$  adalah panjang segmen *embedding* yang diketahui,  $s[0] \dots s[n-1]$ .
- Lakukan FFT terhadap  $n-1$  sinyal tersebut, kemudian cari nilai fasanya  $\Phi$  sesuai persamaan 2.2.
- Konversi nilai fasa yang didapatkan,  $\pi/2$  menjadi bit 1 dan  $-\pi/2$  menjadi bit 0 sebanyak panjang bit *watermark*. Nilai fasa dibandingkan dengan nilai fasa sinyal suara asli.
- Didapatkan data bit-bit *watermark* sesuai dengan hasil konversi. Bit-bit *watermark* dibandingkan dengan bit-bit *watermark* asli untuk mengetahui kebenarannya.

#### 2.5 BCH dan RS Code

BCH code adalah salah satu algoritma pengkodean koreksi kesalahan biner yang ditemukan oleh Bose dan Chaudhuri pada tahun 1960 yang merujuk kepada metode yang sebelumnya ditemukan secara terpisah oleh Hocquenghem ditahun 1959. Kode BCH merupakan generalisasi dari kode Hamming. Kode BCH juga merupakan binary cyclic codes dimana beberapa simbol tersusun dari  $m$ -bit yang berurutan, dimana  $m$  adalah nilai integer positif yang lebih besar dari 2. Pada binary BCH code terdapat parameter sebagai berikut : [7]

$$\text{Panjang Block} : n = 2^m - 1 \quad (3.1)$$

$$\text{Parity Check Bits} : n - k \leq mt \quad (3.2)$$

$$\text{Jarak Minimum} : d \geq 2t + 1 \quad (3.3)$$

Reed – Solomon codes atau yang dikenal juga dengan sebutan RS code adalah algoritma pengkodean koreksi kesalahan yang ditemukan oleh Irvin Reed dan Gus Solomon pada tahun 1960. Kode RS ini merupakan kode non-binary cyclic dengan simbol yang tersusun dari  $m$ -bit yang berurutan, dimana  $m$  adalah nilai integer positif yang lebih besar dari 2. Kode RS ini juga dikategorikan sebagai subset dari kode BCH non-binary [sehingga memiliki parameter yang sama seperti BCH dengan kemampuan deteksi errornya adalah  $t < n - k$ . [8]

### 3. Analisa dan Pengujian Aplikasi

Watermarking pada audio memiliki tantangan tersendiri dibandingkan dengan gambar atau video, ini karena HAS bekerja pada range frekuensi yang lebar [1]. Dari hal ini saja sudah dapat menjadi dasar masalah untuk segala macam penelitian pada audio watermarking. Namun, pada penelitian tugas akhir ini digunakan metode Phase Coding pada proses penyisipan dan pengambilan datanya.

Dalam menyembunyikan sebuah informasi, ada beberapa kriteria yang harus dipenuhi antara lain:

- Imperceptibility* : pesan informasi yang disembunyikan tidak dapat dipersepsi atau diketahui oleh indra manusia. Dalam audio, keberadaan informasi tidak dapat terdengar oleh telinga.
- Fidelity* : kualitas media penampung tidak berubah banyak setelah disisipkan informasi.
- Recovery* : informasi yang disisipkan harus dapat diambil kembali sewaktu – waktu untuk digunakan sesuai kebutuhan.

Akan tetapi masalah utama yang dihadapi adalah ketika setelah selesai melakukan proses penyisipan dengan metode tersebut. Audio yang dihasilkan dari proses penyisipan atau disebut juga sebagai watermark audio atau watermark signal tersebut diberi berbagai macam serangan untuk menguji ketahanan data yang disisipkan dan menguji kualitas watermark signal tersebut. Pada tugas akhir ini, audio yang dihasilkan tersebut akan dimasukkan kedalam ambient mode atau mode ketika audio tersebut diputar atau dibunyikan, kemudian direkam kembali menggunakan alat yang berbeda, contohnya smartphone.

Didalam ambient mode biasanya watermark signal akan mengalami gangguan dari berbagai macam noise yang dihasilkan dari proses perekaman. Sehingga ketika audio hasil rekaman tersebut dimasukkan kedalam proses ekstraksi, data yang sudah disisipkan sebelumnya biasanya sudah tidak ada ataupun bisa berbeda dari data awal yang disisipkan. Untuk menangani hal tersebut diperlukan sebuah sinkronisasi yang hebat untuk penentuan letak data informasinya dan juga parameter – parameter yang pas saat proses penyisipan agar audio yang dihasilkan tahan terhadap proses ambient mode.

#### 3.1. Pengujian Aplikasi

Audio watermark yang dihasilkan akan diukur secara objektif pada Ambient Mode dan secara Subjectif pada File Mode, serta akan digunakan TestBed pada analisa File Mode. Parameter yang digunakan secara objektif pada Ambient Mode antara lain adalah BER, Jarak, dan Level Distorsi. Pada analisa Subjectif akan digunakan metode MOS sebagai parameter pengukur imperceptibility-nya.

Penilaian yang dilakukan pada penelitian ini menggunakan 5 jenis audio yang berbeda sebagai audio dasar yang akan disisipkan informasi yang berupa text. Audio tersebut dapat dilihat pada tabel 1 berikut:

Tabel 1 Berkas Audio dan Jenisnya

Audio	Jenis Audio
Car.wav	Sound effect
Htf.wav	Musik Pop
Ins.wav	Instrumental
Roll.wav	Musik Rock
Sule.wav	Voice / Suara

#### 3.2. Analisa Perbandingan Ukuran Frame Terhadap Nilai BER

Tabel 2 Perbandingan Ukuran Frame terhadap BER

Nilai N	Audio Asli	Audio BCH	BER BCH	Tingkat Akurasi BCH	Audio RS	BER RS	Tingkat Akurasi RS
512	car	car1e	0.5	50%	rcar1a	0.54	45.8%
	htf	htf1e	0.25	75%	rhtf1a	0.33	66.7%
	ins	ins1e	0.46	54.2%	rins1a	0.58	41.7%
	roll	roll1e	0.29	70.8%	rroll1a	0.58	41.7%
	sule	sule1e	0.38	62.5%	rsule1a	0.42	58.3%
1024	car	car1f	0.5	50.0%	rcar1b	0.46	54.2%
	htf	htf1f	0.46	54.2%	rhtf1b	0.38	62.5%
	ins	ins1f	0.46	54.2%	rins1b	0.33	66.7%
	roll	roll1f	0.67	33.3%	rroll1b	0.50	50.0%
	sule	sule1f	0.54	45.8%	rsule1b	0.63	37.5%
2048	car	car1a	0.63	37.5%	rcar1c	0.46	54.2%

	htf	htf1a	0.46	54.2%	rhtf1c	0.46	54.2%
	ins	ins1a	0.42	58.3%	rins1c	0.46	54.2%
	roll	rolla	0.58	41.7%	rrollc	0.33	66.7%
	sule	sule1a	0.63	37.5%	rsule1c	0.46	54.2%
4096	car	car1b	0.29	70.8%	rcar1d	0.42	58.3%
	htf	htf1b	0.46	54.2%	rhtf1d	0.58	41.7%
	ins	ins1b	0.33	66.7%	rins1d	0.42	58.3%
	roll	rollb	0.50	50.0%	rrol1d	0.38	62.5%
	sule	sule1b	0.29	70.8%	rsule1d	0.46	54.2%
8192	car	car1c	Error	#VALUE!	rcar1e	Error	#VALUE!
	htf	htf1c	0.42	58.3%	rhtf1e	0.50	50.0%
	ins	ins1c	0.63	37.5%	rins1e	0.50	50.0%
	roll	rollc	0.54	45.8%	rrol1e	0.63	37.5%
	sule	sule1c	0.46	54.2%	rsule1e	0.38	62.5%

Pada Tabel 2 diatas dapat dilihat bahwa terdapat 6 buah ukuran *frame* (nilai N) yang berbeda, yaitu 512, 1024, 2048, 4096, dan 8192. Ukuran *frame* tersebut masing – masing diinputkan terhadap tiap jenis audio dengan informasi yang disisipkan, dan jarak serta level distorsi perekaman sama pada setiap audio. Berdasarkan hasil tersebut, maka dapat disimpulkan bahwa tingkat akurasi tertinggi dimiliki oleh BCH code dengan ukuran *frame* sebesar 512 pada masing masing *frame*-nya. Akan tetapi BCH *code* bergantung pada jenis audio sebagai host dan ukuran *frame* yang tepat untuk menghasilkan tingkat akurasi yang tinggi. Sedangkan RS *code* memiliki kestabilan akurasi yang lebih tidak tergantung pada jenis audio, hanya membutuhkan ukuran *frame* yang tepat untuk menghasilkan tingkat akurasi yang tinggi pada *ambient mode*.

### 3.3. Analisa Pengaruh Jarak Terhadap Nilai BER

Tabel 3 Pengaruh Jarak Terhadap BER

Nilai N	Audio Asli	Audio BCH	BER BCH	Tingkat Akurasi BCH	Audio RS	BER RS	Tingkat Akurasi RS
512	car	car1e	0.50	50.00%	rcar1a	0.54	45.83%
		car2e	0.54	45.83%	rcar2a	0.54	45.83%
	htf	htf1e	0.25	75.00%	rhtf1a	0.33	66.67%
		htf2e	0.42	58.33%	rhtf2a	0.33	66.67%
	ins	ins1e	0.46	54.17%	rins1a	0.58	41.67%
		ins2e	0.50	50.00%	rins2a	0.58	41.67%
	roll	rol1e	0.29	70.83%	rrol1a	0.58	41.67%
		rol2e	0.63	37.50%	rrol2a	0.58	41.67%
	sule	sule1e	0.38	62.50%	rsule1a	0.42	58.33%
		sule2e	0.38	62.50%	rsule2a	0.75	25.00%

Pada tabel 3 diatas merupakan hasil pengujian dengan jarak rekam yang berbeda yaitu 30cm (kode: 1) dan 1m (kode: 2), namun parameter ukuran *frame* dan level distorsi yang sama. Perbandingan perbedaan pada tabel diatas menunjukkan untuk tiap masing – masing jenis audio. Dari hasil diatas dapat disimpulkan bahwa, tingkat akurasi dengan menggunakan BCH code lebih tinggi dibandingkan dengan menggunakan RS code, akan tetapi RS code memiliki tingkat kestabilan akurasi yang lebih dibanding dengan BCH code yang bergantung pada jarak.

### 3.4. Analisa Pengaruh Level Distorsi Terhadap Nilai BER

Tabel 4 Pengaruh Level Distorsi terhadap BER

Nilai N	Level Distorsi	Audio Asli	Audio BCH	BER BCH	Tingkat Akurasi BCH	Audio RS	BER RS	Tingkat Akurasi RS
512	60 dB	car	car1e	0.6	42%	rcar1a	0.54	45.8%
		htf	htf1e	0.33	67%	rhtf1a	0.38	62.5%
		ins	ins1e	0.58	41.7%	rins1a	0.63	37.5%
		roll	rol1e	0.38	62.5%	rrol1a	0.58	41.7%
		sule	sule1e	0.58	41.7%	rsule1a	0.46	54.2%
	70 dB	car	car1e	0.5	50%	rcar1a	0.54	45.8%
		htf	htf1e	0.25	75%	rhtf1a	0.33	66.7%
		ins	ins1e	0.54	45.8%	rins1a	0.58	41.7%
		roll	rol1e	0.29	70.8%	rrol1a	0.58	41.7%
		sule	sule1e	0.58	41.7%	rsule1a	0.42	58.3%

Pada tabel 4 diatas, dapat dilihat bahwa dengan menggunakan BCH code, tingkat akurasi pada level distorsi 70 dB (volume lebih besar) lebih tinggi untuk semua jenis audio disbanding dengan level distorsi 60 dB. Sedangkan dengan menggunakan RS code, hanya pada audio berjenis sound effect dan instrument yang tingkat akurasinya pada level 70 dB sama dengan 60 dB, untuk audio jenis lainnya tingkat akurasi pada 70 dB lebih besar dari 60 dB.

### 3.5. Analisa File Mode dengan Menggunakan TestBed

Tesbed merupakan sekumpulan tes yang dilakukan secara mandiri untuk mengetes kualitas ketahanan audio yang dihasilkan dari sistem. Dari 13 jenis serangan yang dilakukan terhadap file mode dengan BCH code dihasilkan data nilai BER yang dapat dilihat pada tabel 5 berikut:

Tabel 5 Hasil TestBed BCH code

Item #	carbch	htfbch	insbch	rolbch	sulebch
test 1	0.46	0	0	0	0.38
test 2	0.42	0.04	0	0	0.42
test 4	0.50	0	0	0	0.38
test 5	0.38	0	0	0	0.38
test 6	0.50	0	0	0	0.38
test 9	0.25	0.54	0.54	0.33	0.38
test 10	0.42	0	0	0	0.50
test 11	0.46	0	0	0	0.42
test 12	0.63	0.63	0.46	0.58	0.50
test 13	0.50	0	0	0	0.38
test 14	0.50	0	0	0	0.38
test 15	0.50	0	0	0	0.38

Sedangkan untuk RS code, dari 13 jenis serangan yang dilakukan terhadap 5 jenis audio yang berbeda, didapatkan hasil pada tabel 6 sebagai berikut :

Tabel 6 Hasil TestBed RS Code

Item #	carrs	htfrs	insrs	rolrs	sulers
test 1	0.46	0	0	0	0.42
test 2	0.46	0	0	0	0.50
test 4	0.46	0	0	0	0.42
test 5	0.58	0	0	0	0.46
test 6	0.46	0	0	0	0.42

test 9	0.42	0.63	0.58	0.58	0.38
test 10	0.42	0	0	0	0.54
test 11	0.50	0.04	0	0	0.46
test 12	0.38	0.42	0.42	0.46	0.38
test 13	0.46	0	0	0	0.42
test 14	0.46	0	0	0	0.42
test 15	0.46	0	0	0	0.42

Dari kedua tabel diatas dapat dilihat bahwa serangan yang paling merusak untuk semua jenis audio pada BCH code dan RS code adalah item no. 9 yaitu *pitch shifting* dan item no. 12 yaitu *cropping* audio.

#### 4. Kesimpulan

Berdasarkan pembahasan dan evaluasi hasil penelitian, maka dapat ditarik kesimpulan sebagai berikut :

1. Pengaruh nilai ukuran frame terhadap tingkat akurasi juga ikut di pengaruhi oleh jenis audio yang digunakan sebagai host. Berdasarkan hasil yang didapatkan, nilai BER dengan menggunakan BCH code didapatkan tingkat akurasi yang cukup besar untuk nilai ukuran frame tertentu terhadap jenis audio tertentu. Sedangkan nilai BER yang dihasilkan dengan menggunakan RS code lebih stabil tidak bergantung kepada jenis audio, hanya membutuhkan ukuran frame yang tepat agar didapatkan tingkat akurasi yang tinggi.
2. Pengaruh Jarak rekaman terhadap tingkat akurasi pada BCH code didapatkan hasil bahwa pada jarak yang lebih dekat tingkat akurasi yang dihasilkan akan lebih tinggi. Sedangkan jika menggunakan RS code, tingkat akurasi yang dihasilkan tidak terpengaruh terhadap jarak 30cm dan 1m. Tingkat akurasi yang dihasilkan dari RS code cenderung lebih stabil dibanding dengan BCH code.
3. Level distorsi juga berpengaruh besar dalam ambiet mode, semakin tinggi level distorsi atau volume yang digunakan dalam proses perekaman, maka akan semakin bagus tingkat akurasi yang dihasilkan baik menggunakan BCH atau RS code. Pada kondisi perekaman juga, kondisi noise ruangan haruslah sepi agar menambah tingkat akurasi pada saat perekaman.
4. Dari ke 16 jenis serangan pada *file mode* dengan menggunakan *TestBed*, serangan yang paling merusak informasi adalah serangan no. 9 yaitu *pitch shifting* dan serangan no. 12 yaitu *cropping*.

#### Referensi

- [1] Marimoto, N., Bender, W., Gruhl, D., & Lu, A. 1996 . *Techniques for Data Hiding*. IBM Systems Journal vol.35 (3-4), 313-336.
- [2] Petrovic, R. 2001. "*Audio signal watermarking based on replica modulation*", 5<sup>th</sup> International Conference TELSIKS'01, 227-234.
- [3] Hargtung, F., Kutter, M. 1998. "*Multimedia watermarking techniques*", Proc. IEEE, Vol. 86, 1079-1107.
- [4] Chauhan, S., Rizvi, S. 2013. "*A Survey: Digital Audio Watermarking Techniques and Applications*", 4<sup>th</sup> International Conference on Computer and Communication Technologies, 185-192.
- [5] Sibagariang, Julius A. 2012. "*Analisis Watermark Pada File Audio Berbasis Metode Phase Coding*". Bandung:IT TELKOM.
- [6] Rumondang, Martharany. 2006. "*Perlindungan Hak Cipta Pada Data Audio Menggunakan Teknik Watermarking Phase Coding*". Bandung:ITB.
- [7] Wallace, Hank. 2001. "*Error Detection and Correction Using BCH code*". URL [www.aqdi.com](http://www.aqdi.com)
- [8] Sklar, Bernard. 2001. "*Reed – Solomon Codes*" Prentice-Hall.