

PEREKAM/PEMUTAR SUARA DIGITAL BERKAPASITAS BESAR UNTUK APLIKASI *EMBEDDED SYSTEM*

Yoppy*, Waru Djuriatno**, dan Suprpto**

*Pusat Penelitian Sistem Mutu dan Teknologi Pengujian (P2SMTP) LIPI
Komplek PUSPIPEK Gedung 410 Serpong, Tangerang, Banten
e-mail: elektro_ub@yahoo.com

**Jurusan Teknik Elektro Universitas Brawijaya
Jln. Mayjend Haryono 167, Malang

ABSTRACT

In embedded-system designs that involve just a small capacity voice-recorder, one may use EEPROM or voice-recorder IC. However, sometimes one needs a high capacity voice recorder also. In this case, EEPROM can't fulfill the needs anymore because it's hard to get EEPROM with capacity even at 512 kbyte. Meanwhile, the ISD[®] voice-recorder IC could store voice up to 16 minutes only. For this background, the writer proposes a high-capacity digital voice recorder which is able to record voice for several hours. It uses SD Card as the storage media.. With sampling rate at 8 kHz/8 bit and 512 MB SD Card, the recorder can store voice until 18 hours. The recorded voices are stored in WAV file format so that can be transferred from and to PC when needed.

Keywords: WAV file, FAT16 file system, SD card, Voice recorder.

PENDAHULUAN

Secure Digital Card (SD Card) adalah media penyimpanan digital yang menggunakan teknologi *flash*. Jenis memori ini memiliki ukuran kecil, 24 x 32 x 2,1 mm. Selain itu, SD Card memiliki kapasitas yang besar hingga 8 GB. Tidak hanya itu, SD Card tergolong relatif murah pula. Memori SD Card diformat dengan *file system* FAT16 seperti pada *hard-disk*.¹ SD Card menyediakan *interface* SPI (*Serial Peripheral Interface*) untuk komunikasi dengan devais lain. Memori satu ini banyak digunakan pada kamera digital, ponsel, PDA, dan sebagainya.

Dalam sebuah desain *embedded system* yang membutuhkan perekam suara berkapasitas kecil, seseorang cukup menggunakan IC *Electrically Erasable Programmable Read Only Memory* (EEPROM) atau IC perekam suara ISD[®].² Namun, ada kalanya juga dibutuhkan sebuah perekam suara yang memiliki kapasitas besar seperti aplikasi kamus bersuara, *voice announcer*, dan sebagainya. Dalam hal ini, EEPROM tidak

bisa lagi memenuhi kebutuhan tersebut karena sulit sekali mendapatkan EEPROM berkapasitas 512 kbyte sekalipun. Sementara itu, IC perekam suara ISD[®] yang umum digunakan saat ini maksimal sampai 16 menit.² Lalu bagaimana jika ingin merekam lebih dari itu?

Untuk menjawab pertanyaan tersebut, maka dirancanglah sebuah alat perekam suara berkapasitas besar menggunakan SD Card sebagai media penyimpanan. Dengan sampling 8 kHz/8 bit dan SD Card 512 Mbyte, alat ini mampu merekam hingga 65.000 sekon atau setara dengan 18 jam.

Secara garis besar, pembuatan alat ini terdiri atas dua bagian, yaitu *hardware* dan *software*. Implementasi *hardware* meliputi pengondisi sinyal (*penguat, anti-aliasing filter, reconstruction filter*), *keypad, LCD display*, dan antarmuka dengan SD Card. Sementara itu, bagian *software* dirancang sedemikian rupa agar dapat merekam, menghapus, dan memutar suara serta memilih *file*. Tidak hanya itu, *software* juga dirancang kompatibel dengan *file system* *File Allocation*

Table (FAT16). Artinya, alat ini dapat membaca *file* WAV yang direkam dari PC yang kemudian di-copy-kan ke SD Card. Sebaliknya, *file* WAV yang direkam dengan alat ini dapat dipindahkan ke PC sewaktu-waktu jika diperlukan.

Proses Perekaman dan Pemutaran Suara

Secara sederhana, proses perekaman suara diawali dengan mengubah gelombang akustik menjadi sinyal analog menggunakan mikrofon sebagai transduser. Setelah itu, sinyal analog tersebut diproses lagi menjadi sinyal digital menggunakan *Analog to Digital Converter* (ADC). Kini, sinyal suara sudah siap untuk disimpan dalam media penyimpan digital, misalnya SD Card.

Dalam proses perekaman yang sesungguhnya, sinyal analog keluaran dari mikrofon tidak bisa langsung dimasukkan ke ADC karena level tegangannya masih terlalu lemah. Sinyal analog ini juga tidak boleh mengandung komponen frekuensi yang melebihi $F_s/2$, dengan F_s adalah frekuensi sampling ADC. Dalam hal ini, sinyal suara yang memiliki komponen frekuensi dari 300 hingga 3.000 Hz biasanya di-sampling dengan frekuensi F_s 8 kHz.³ Oleh karena itu, sinyal analog dengan komponen frekuensi lebih dari 4.000 harus diredam terlebih dahulu sebelum masuk ke ADC. Jika tidak, akan terjadi suatu fenomena yang dikenal dengan istilah *aliasing* seperti yang ditunjukkan dalam Gambar 1.⁴ Untuk mengatasi persoalan di atas, sebelum diumpankan ke ADC sinyal analog harus dikuatkan dan difilter terlebih dahulu. Filter ini disebut filter *anti-aliasing*.

Proses pemutaran suara adalah kebalikan dari proses perekaman suara. Pemutaran suara berarti mengubah data digital menjadi sinyal analog, untuk kemudian dikonversi menjadi gelombang akustik. Hasil konversi DAC masih berupa sinyal analog yang 'kotor'. Dari sudut pandang domain frekuensi, keluaran DAC

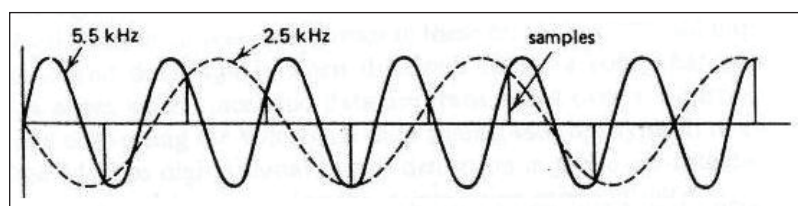
yang seharusnya cuma terdiri atas frekuensi fundamental (*voice*) tapi kenyataannya juga bercampur dengan spektrum bayangan (*image spectra*), $k \cdot f_s - f_t$ dan $k \cdot f_s + f_t$ dengan $k = 1, 2, 3, 4, \dots$; frekuensi sampling (f_s) dan frekuensi fundamental (f_t).⁵ Karena itu, sinyal 'kotor' ini 'dibersihkan' dengan *reconstruction filter* yang berupa *low-pass filter*. Gambar 2 menunjukkan perbandingan keluaran DAC dengan dan tanpa *reconstruction filter*.⁵

SD Card dan Mikrokontroler R8C/13

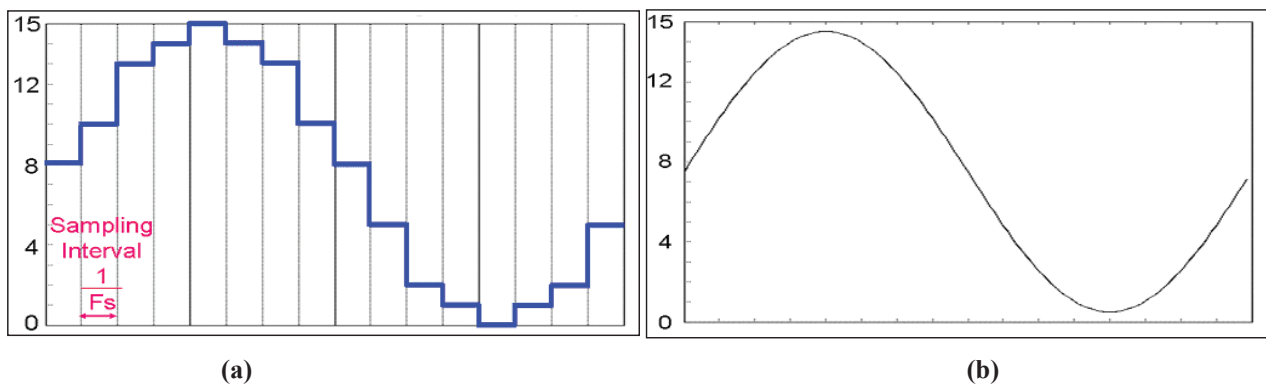
R8C/13 adalah *general-purpose MCU* produksi Renesas yang memiliki 12 *channel* ADC, 4 buah *timer*, 2 buah antarmuka serial, dan fitur-fitur lainnya. R8C/13 bekerja pada *range* tegangan catu yang lebar, yaitu 2,7–5 volt. Komunikasi R8C/13 dengan SD Card berlangsung melalui protokol *Serial Peripheral Interface* (SPI).

Memori SD Card diformat dengan *file system* FAT16 yang secara *default* sudah berisi *Master Boot Record* (MBR) dan *Volume Boot Record* (VBR). Namun, hal ini tidak serta-merta berarti semua data yang direkam dalam SD Card kompatibel dengan *file system* FAT16. Untuk itu, data yang direkam harus memenuhi aturan dan standar *file system* FAT16 itu sendiri. Di sini lah *software* (atau lebih tepatnya *firmware*) mikrokontroler bertanggung jawab penuh untuk menangani abstraksi-abstraksi dalam *file system*.

Data suara yang hendak direkam dalam SD Card adalah berupa *file* WAVE. *File* WAVE (.WAV) adalah format *file* yang umum digunakan pada Windows untuk menyimpan data audio digital tanpa kompresi, seperti *Pulse Code Modulation* (PCM). WAV mendukung audio dengan berbagai resolusi bit dan kecepatan sampling. File WAV terdiri dari tiga bagian besar, yaitu *RIFF chunk descriptor*, *'fmt' sub-chunk*, dan *data sub-chunk*.⁶



Gambar 1. Fenomena *aliasing*



Gambar 2. Sinyal keluaran DAC: (a) tanpa melalui reconstruction filter; (b) melalui reconstruction filter

METODE PENELITIAN

Berdasarkan kebutuhan yang telah dipaparkan pada bagian sebelumnya, maka proses pembuatan alat ini dibagi dalam dua bagian, yaitu *hardware* dan *software*.

Perancangan dan Implementasi *Hardware*

Perancangan *hardware* terdiri dari penguat mikrofon, *anti-aliasing filter*, *reconstruction filter*, DAC, antarmuka SPI R8C/13 dengan SD Card, *keypad*, LCD display, dan RTC (*Real Time Clock*). Gambar 3 menunjukkan diagram blok perancangan *hardware*.

Untuk diperhatikan bahwa pada bagian blok *Control/Keypad*, input dapat berupa *keypad* atau berupa sinyal kontrol dari devais lain. Dalam perancangan ini, yang digunakan adalah *keypad*. Penggunaan *keypad* hanya untuk tujuan uji coba. Jika yang digunakan adalah sinyal kontrol dari devais lain alih-alih menggunakan *keypad*, misalnya alat ini sebagai subsistem dalam sistem yang lebih besar, dibutuhkan penyesuaian konfigurasi *hardware* dan *software*. Hal ini bukanlah suatu masalah karena tersedianya *source-code software* dan rangkaian *hardware* alat. Di pasaran memang sudah banyak dijual alat perekam suara (Sony, Canon, Toshiba, dan lain-lain) dengan kapasitas besar. Akan tetapi, ada dua hal penting yang tidak kita dapatkan dari alat-alat tersebut, yaitu *source code* dan skematik *hardware*. Tanpa kedua hal itu, sangatlah sulit atau bahkan tidak mungkin untuk melakukan *interfacing* dengan devais lain, sebagaimana dibutuhkan dalam desain *embedded-system*.

Penguat Mikrofon dan *Anti-Aliasing Filter*

Tahap pertama dalam proses perekaman adalah mengubah gelombang akustik suara menjadi sinyal analog menggunakan mikrofon. Sinyal analog yang masih lemah ini perlu dikuatkan. Sebagai penguat, digunakan IC LM386. Berdasarkan *data sheet*, apabila pin 1 dan pin 8 dihubungkan dengan kapasitor 10 μ F maka akan didapatkan penguatan sebesar 200 kali.⁷

Sinyal yang telah dikuatkan kemudian dilewatkan pada *low pass filter* (*anti-aliasing*) dengan frekuensi *cut-off*, $f_c = 3400$ Hz. Dengan nilai kapasitor C sebesar 56 nF, maka nilai resistor R adalah sebesar ± 1 k Ω berdasarkan persamaan:

$$R = 1 / (2\pi \cdot f_c \cdot C) \quad (1)$$

Gambar 4 menunjukkan implementasi penguat dan *anti-aliasing filter*.

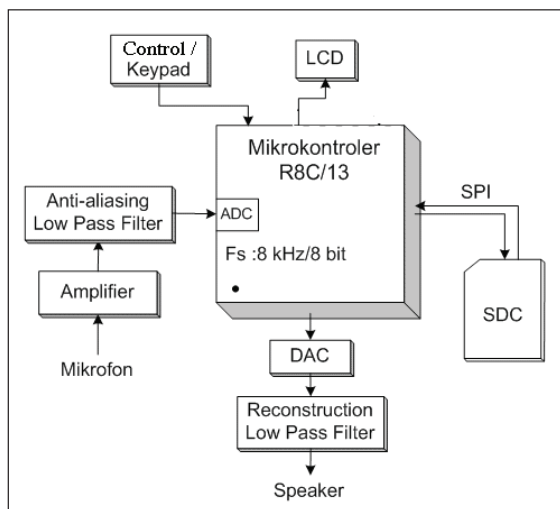
Keypad

Keypad yang dirancang terdiri atas 5 buah tombol yang berfungsi untuk merekam, membaca, lompat ke *file* sebelumnya, lompat ke *file* berikutnya, *pause*, *stop* dan hapus *file*. Implementasi rangkaian *keypad* menggunakan metode pembagi tegangan yang diumpankan ke salah satu *channel* ADC mikrokontroler R8C/13. Gambar 5 menunjukkan rangkaian *keypad*. Nilai-nilai resistor ditentukan sedemikian rupa agar hasil konversi digital dari tiap tombol tidak berdekatan.

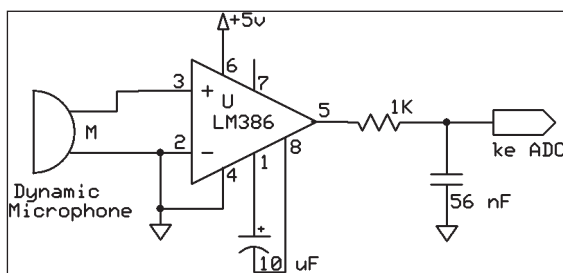
DAC dan *Reconstruction Filter*

Untuk mengubah data digital kembali menjadi sinyal analog digunakan DAC0832. Normalnya, DAC0832 dioperasikan dalam mode *current-switching*. Namun, mode ini kurang cocok untuk desain dengan catu daya tunggal. Oleh karena itu, dalam perancangan ini DAC0832 dioperasikan dalam mode *voltage-switching*.

Keluaran DAC0832 difilter dengan *reconstruction filter*. Jika ditentukan frekuensi *cut-off* $f_c = 3400$ Hz dan nilai kapasitor 10 nF, maka didapatkan nilai resistor R sebesar $\pm 4,7$ k Ω berdasarkan Persamaan (1). Gambar 6 menunjukkan rangkaian DAC dan *reconstruction filter*. Digunakan filter orde tiga untuk mendapatkan kurva respon magnitudo yang lebih tajam. Op-amp LM324 digunakan sebagai *buffer* untuk mengurangi efek pembebanan pada keluaran DAC.



Gambar 3. Diagram blok *hardware*



Gambar 4. Rangkaian penguat dan *anti-aliasing filter*

Antarmuka SPI SD Card dengan R8C/13

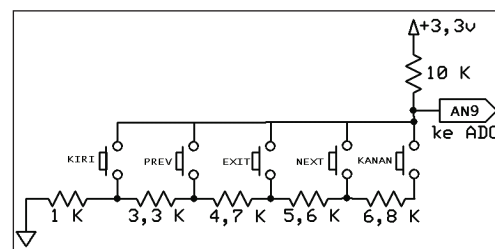
SD Card bekerja dengan catu daya 3,3 volt. Sementara itu, mikrokontroler R8C/13 dapat bekerja pada *range* 2,7–5 volt. Oleh karena itu, keduanya dioperasikan dengan catu daya 3,3 volt. Jalur-jalur SPI SD Card dapat langsung dihubungkan dengan antarmuka serial R8C/13.

Gambar 7 menunjukkan koneksi SD Card dengan R8C/13. R8C/13 bertindak sebagai *master* dan SD Card sebagai *slave*.

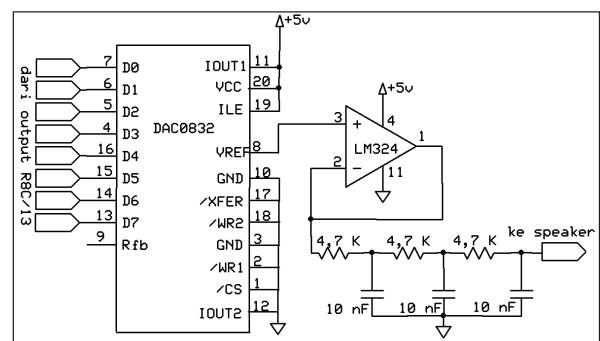
Perancangan dan Implementasi *Software*

Adapun rancangan *software* dapat diabstraksikan dalam beberapa lapisan, seperti ditunjukkan dalam Gambar 8. Lapisan aplikasi menggunakan fungsi-fungsi pada lapisan FAT16. Sementara itu, lapisan FAT16 menggunakan fungsi-fungsi yang ada di bawahnya, yaitu lapisan *hardware*.

Pada lapisan *hardware*, *software* bertugas menangani beberapa hal, yaitu inisialisasi R8C/13 itu sendiri, komunikasi dengan SD Card, mengendalikan LCD dan *keypad*, *sampling* sinyal suara yang masuk ADC, serta mengeluarkan data digital pada DAC dengan frekuensi yang sama



Gambar 5. Rangkaian *keypad*



Gambar 6. Rangkaian DAC dan *reconstruction filter*

dengan saat di-*sampling*. Dengan f_s frekuensi *sampling* ADC sebesar 8 kHz/8 bit dan n 8 bit maka didapatkan *bit rate* 8 kbyte/s berdasarkan persamaan:

$$\text{Bit rate} = f_s \cdot n \quad (2)$$

Jika digunakan SD Card dengan kapasitas 512 MB maka alat ini dapat menyimpan suara selama +/- 65.000 sekon atau setara dengan 18 jam.

Sementara itu, lapisan FAT bertugas menangani *file system* FAT16, seperti analisis MBR, analisis VBR, mencari *cluster* kosong, penamaan *file*, dan sebagainya. Dengan alat ini, rekaman suara akan tersimpan dalam bentuk *file* WAV yang dapat dipindahkan ke PC jika sewaktu-waktu diperlukan. Sebaliknya, *file* WAV dapat di-*copy*-kan dari PC ke SD Card untuk kemudian diputar dengan alat ini. Untuk saat ini, *software* dirancang untuk mengakses *file* yang ada di *root directory* dan belum untuk *file* yang ada di *sub-directory*.

Gambar 9 menunjukkan rancangan *state-diagram* alat. Kotak-kotak dalam gambar tersebut mewakili tampilan LCD. Segera setelah dihidupkan, alat akan memasuki *state* A. Alat akan terus tinggal dalam *state* A sampai adanya pemicu. Misalkan kemudian tombol KIRI ditekan, maka alat akan masuk ke *state* E. Dalam kondisi *state* E dan tombol KIRI ditekan maka alat akan masuk ke *state* G dan memutar (*play*) *file* WAV. Demikianlah seterusnya alat akan pindah dari satu *state* ke *state* lainnya hanya jika ada pemicu.

Pada lapisan aplikasi, fungsi-fungsi lebih bersentuhan langsung dengan pemakai, seperti perintah merekam, memutar, *pause*, *stop*, meng-

hapus, lompat ke *file* rekaman sebelumnya, dan lompat ke *file* rekaman selanjutnya.

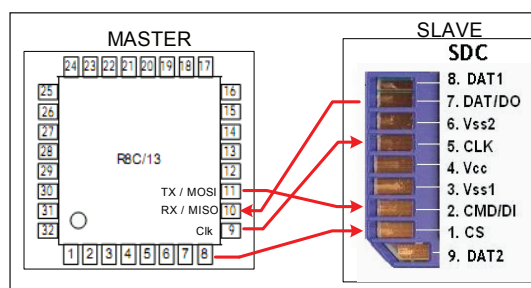
HASIL DAN PEMBAHASAN

Berdasarkan serangkaian pengujian yang telah dilakukan pada *software* dan *hardware*, bisa dikatakan bahwa keduanya telah berjalan sesuai dengan perancangan. Namun untuk mendapatkan performansi yang lebih baik, masih diperlukan penyempurnaan lebih lanjut.

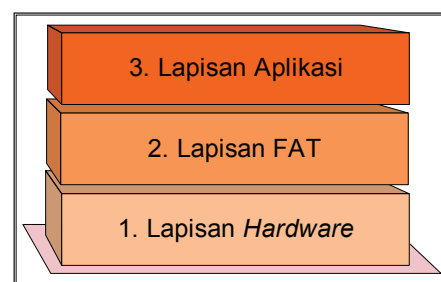
Hasil Pengujian Perekaman Suara

Pengujian hasil perekaman dilakukan dengan membandingkan bentuk gelombang hasil rekaman menggunakan alat ini dan hasil rekaman menggunakan PC. Gambar 10 melukiskan metode pengujian yang dilakukan. Untuk menampilkan bentuk gelombang hasil rekaman digunakan *software* Audacity.

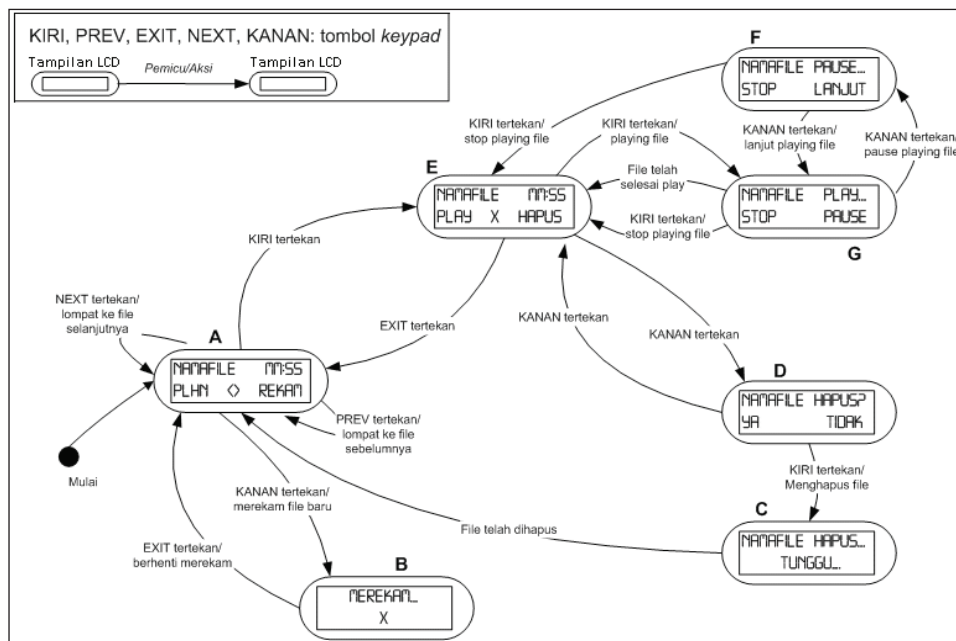
Gambar 11 menunjukkan hasil pengujian perekaman. Bentuk-gelombang bagian atas merupakan hasil rekaman dengan alat ini. Sementara itu, yang bagian bawah merupakan hasil rekaman dengan PC. Berdasarkan pengamatan pada bentuk gelombang tersebut diketahui bahwa hasil rekaman dengan alat mengalami *delay* 0,026 sekon setiap interval 2,048 sekon. Interval 2,048 sekon merupakan durasi suara dalam satu *cluster* memori SD Card. Sebaliknya, *delay* 0,026 sekon merupakan waktu yang dibutuhkan untuk mencari *cluster* selanjutnya dari *linked-list* tabel FAT16. Telinga manusia tentu saja tidak begitu peka untuk mendeteksi terjadinya *delay* selama 0,026 tersebut (Gambar 11b). *Delay* ini dapat



Gambar 7. Koneksi Pin SD Card dengan mikro-kontroler R8C/13



Gambar 8. Rancangan *software*

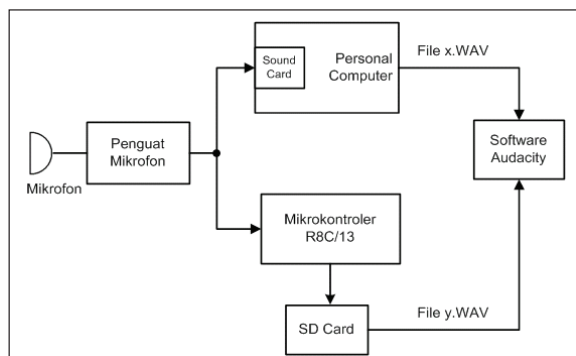


Gambar 9. State-diagram alat

diatasi dengan menggunakan memori RAM mikrokontroler yang lebih besar. Sebagai catatan, mikrokontroler R8C/13 yang digunakan dalam penelitian ini adalah sebesar 1 kbyte.

Hasil Pengujian Pemutaran Suara

Untuk mengetahui performansi pemutaran alat dilakukanlah pengujian sebagai berikut. Sebuah file WAV (misalkan x.WAV) diputarkan menggunakan alat ini. Sinyal analog hasil konversi DAC kemudian direkam menggunakan PC dan didapatkan hasil rekaman berupa file WAV pula (misalkan y.WAV). Setelah itu, file x.WAV dibandingkan dengan file y.WAV menggunakan *software* Audacity. Gambar 12 menunjukkan metode pengujian pemutaran.

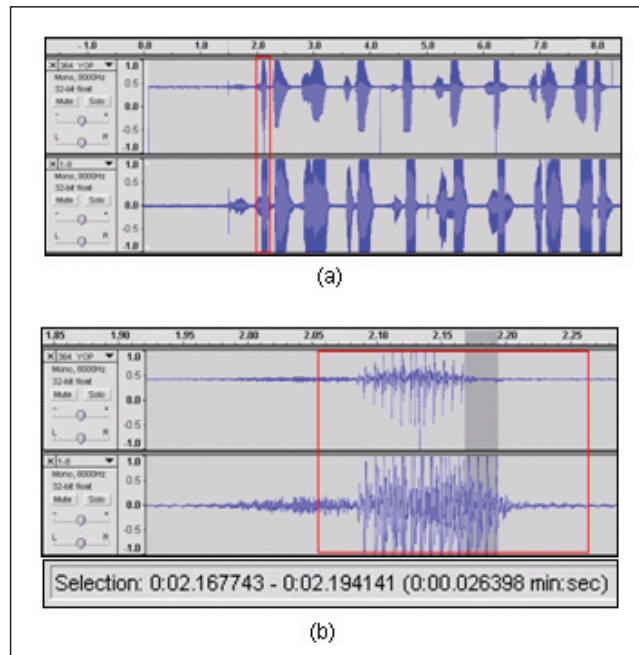


Gambar 10. Metode pengujian perekaman

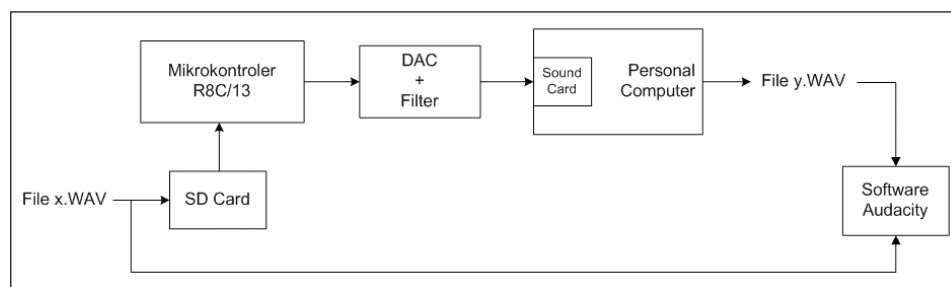
Gambar 13 menunjukkan hasil pengujian pemutaran. Dengan bantuan *software* Audacity diketahui bahwa pada hasil pemutaran terjadi *delay* 0,014 sekon (Gambar 14) dalam setiap interval 2,048 sekon. Seperti pada proses perekaman, interval 2,048 sekon merupakan durasi suara dalam satu *cluster* memori SD Card. Sementara itu, *delay* 0,014 sekon merupakan waktu yang dibutuhkan untuk mencari *cluster* selanjutnya dari *linked-list* tabel FAT16. Telinga manusia tentu tidak begitu peka untuk mendengar adanya *delay* selama 0,014 sekon tersebut. Namun, performansi ini perlu disempurnakan lagi dengan menggunakan memori RAM mikrokontroler yang lebih besar.

Hasil Pengujian Keseluruhan

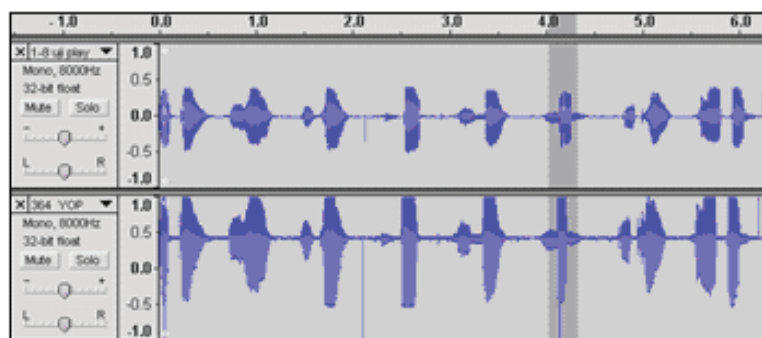
Pengujian ini merupakan pengujian alat secara keseluruhan, baik perangkat keras maupun perangkat lunak. Gambar 15 menunjukkan hasil pengujian alat secara keseluruhan. Dapat dilaporkan bahwa alat telah bekerja sebagaimana yang diharapkan dalam perancangan. Alat dapat menerima perintah melalui *keypad* dan menampilkan informasi melalui LCD serta dapat melakukan fungsi merekam, menghapus, lompat ke *file* sebelumnya, lompat ke *file* selanjutnya, *stop*, *pause*, dan memutar suara dengan baik.



Gambar 11. Delay perekaman: (a) Delay yang terjadi pada saat pencarian cluster berikutnya; (b) Perbesaran dari gambar (a) yang ditandai kotak merah



Gambar 12. Metode pengujian pemutaran

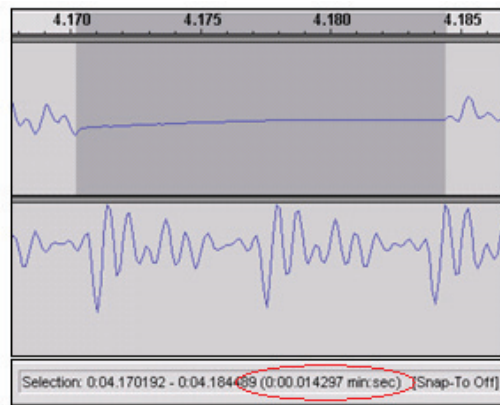


Gambar 13. Perbandingan bentuk gelombang y.WAV(atas) dan x.WAV (bawah)

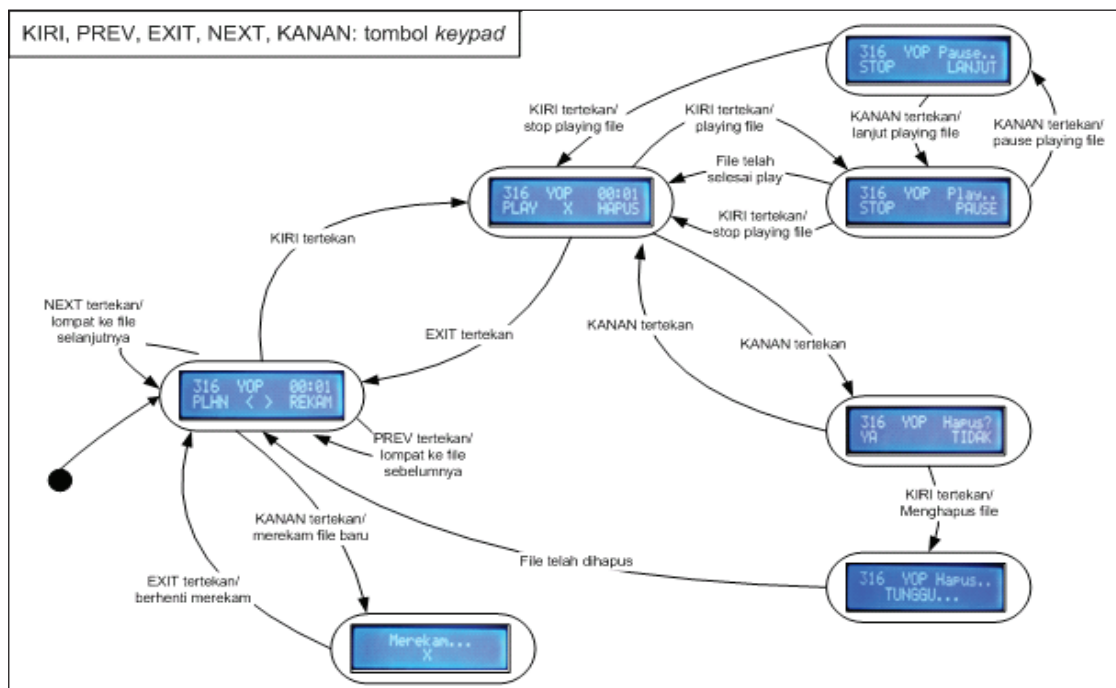
KESIMPULAN DAN SARAN

Alat perekam suara digital ini dapat merekam suara dengan durasi yang panjang. Dengan SD Card 512 MB alat dapat merekam hingga 18 jam. Dengan kemampuan menangani *file system* FAT16, *file* WAV hasil rekaman dapat

dipindahkan dari dan ke PC. Lebih dari itu, yang terpenting yaitu adanya *source-code* dan skematik *hardware* sangat memudahkan untuk melakukan *interfacing* dengan devais lain sebagaimana diperlukan dalam desain *embedded-system*.



Gambar 14. Perbesaran dari Gambar 13



Gambar 15. State-diagram hasil pengujian alat secara keseluruhan

Untuk mengatasi persoalan *delay* yang terjadi dalam proses perekaman dan pemutaran, sebaiknya digunakan memori mikrokontroler yang lebih besar lagi.

DAFTAR PUSTAKA

- ¹SD Association. 2008. SD Technology Overview. (<http://www.sdcard.org>, diakses 9 April 2010).
- ²Nuvoton Technology. 2008. ISD ChipCorder®. (<http://www.nuvoton.com>, diakses 10 Mei 2010).

- ³Eek, J. and Torvmark. 2003. *Wireless Audio Using CC1010*. USA: Chipcon AS.
- ⁴Bellamy, J. 1991. *Digital Telephony*. USA: John Wiley & Sons.
- ⁵Okawara, H. 2008. *Mixed Signal Lecture Series*. Japan: Verigy.
- ⁶Wilson, S. 2003. WAVE PCM sound file format. (<http://www.ccrma.stanford.edu>, diakses 9 April 2010).
- ⁷National Semiconductor. 2000. *LM386 Low Voltage Audio Power Amplifier Datasheet*. USA.