

REKONTRUKSI FILE DATA MAGNETOMETER UNTUK OBSERVASI NEAR REAL TIME AKTIFITAS MEDAN MAGNET

Setyanto. C. P

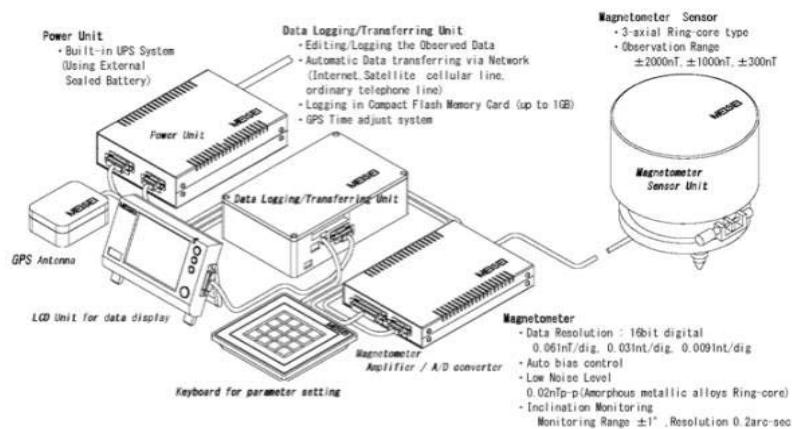
Email : setya_cp@yahoo.com

Abstrak

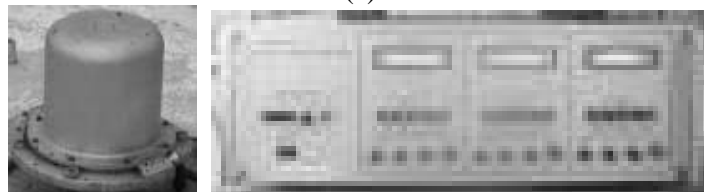
Data hasil pengamatan yang dilakukan oleh magnetometer tersimpan dalam suatu media penyimpanan - *compact flash* dengan format text file. Ukuran file yang dibuat oleh file dengan format text selalu berubah-ubah sehingga susah memperkirakan kapan batas pergantian dari media penyimpanan - *compact flash* tersebut. Rekonstruksi file data magnetometer kedalam format bineri file bertujuan untuk memperkirakan lama pemakaian memori media penyimpanan - *compact flash* pada magnetometer, memperkecil ukuran dari file data yang terbentuk, mempercepat pada saat transfer file. Dalam makalah ini membahas mengenai teknik rekonstruksi file. Output yang dihasilkan dari sistem ini adalah memaksimalkan penggunaan media penyimpanan - *compact flash* dengan memperkecil ukuran file data, pembuatan file setiap lima menit dan harian.

PENDAHULUAN

Magdas dan fluxgate magnetometer merupakan suatu instrument yang digunakan untuk melakukan pengamatan terhadap variasi medan magnet. Kedua tipe tersebut memiliki tipe sensor yang sama yaitu *ring core*. Magdas merekam data absolute geomagnet komponen H, D, Z, dan F sedangkan fluxgate magnetometer merekam data variasi geomagnet komponen H, D, Z. Komponen H menyatakan medan geomagnet arah utara-selatan, D timur-barat dan Z arah vertical. Gambar magnetometer ditunjukkan pada gambar.1.



(a)



(b)

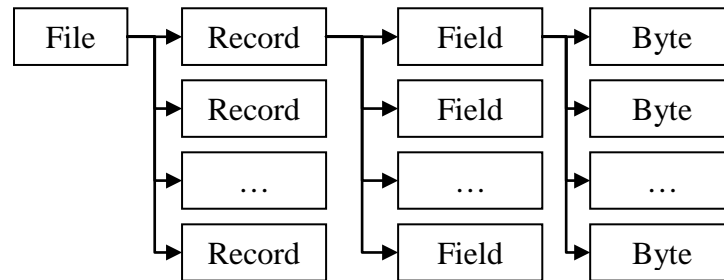
Gambar.1. (a). Magdas dan (b). Fluxgate Magnetometer

Data yang dihasilkan oleh magnetometer tersebut terekam dalam suatu media - *Compact Flash* yang merupakan media penyimpanan yang bersifat *removeable*. Tipe file data yang digunakan pada magnetometer tersebut untuk menyimpan data adalah file teks, dimana ada beberapa kelemahan dari penggunaan tipe file ini diantaranya (1) penulisan datanya berdasarkan karakter dan bersifat sekuensial (2) sifat file ini terbuka dalam artian hampir bisa dibaca oleh semua software editor, (3) sulit memperkirakan ukuran kapasitas file secara pasti. Untuk efisiensi

penggunaan kapasitas *compat flash* serta untuk mendapatkan ukuran kapasitas yang tetap dari file perekaman data magnetometer maka perlu dibuat suatu program yang bertujuan untuk merekontruksi file magnetometer tersebut.

Struktur File

Hampir setiap instrument pengamatan memerlukan suatu *storage* sebagai media untuk menyimpan atau membaca ulang suatu data. Data hasil pengamatan magnetometer disimpan dalam *storage- compat flash* sebagai suatu kesatuan yang biasa disebut file. Suatu file merupakan organisasi dari sejumlah record. Masing-masing record dapat terdiri dari satu atau beberapa *field* dan setiap *field* terdiri dari satu atau beberapa byte data. Satu byte data terdiri dari susunan 8 bit data. Struktur data dari file ditunjukkan pada gambar.2.



Gambar.2. Struktur File

Tipe File

File dapat didefinisikan sebagai objek data yang tersimpan pada media penyimpanan . Secara umum file diklasifikasi kan dalam dua tipe yaitu (1) file teks, file yang pola penyimpanan datanya dalam bentuk karakter - karakter. Data pada file teks disimpan dan dibaca secara sekuensial / baris per baris. File tipe ini biasa dipakai untuk menyimpan data bertipe karakter atau string sedangkan , (2) file biner, merupakan file yang pola penyimpanan terstruktur dalam bentuk biner. Tipe file yang sebelumnya digunakan untuk meyimpan data pada magnetometer adalah tipe teks. File ini terdiri dari bagian *header* dan bagian data yang tersusun atas tiga kolom (H, D, Z), dengan format penulisan sebagai berikut:

AYYMMDDHHNN.STN

- A : kode file ascii
- YY : dua digit yang merepresentasikan tahun
- MM : dua digit yang merepresentasikan bulan
- DD : dua digit yang merepresentasikan tanggal
- HH : dua digit yang merepresentasikan jam
- NN : dua digit yang merepresentasikan menit
- STN : tiga digit yang merepresentasikan kode stasiun pengamatan

contoh : A0905060500.BIK, ukuran file 6.402 byte.

```

BIK      2009  5  6 LT=   9 ST=   1 sec
H=  1.14e-02 D=  1.14e-02 Z=  1.14e-02 nT/LSB
GPS 9999
■
-19248  14495  -1744
-19248  14495  -1760
-19248  14495  -1744
-19248  14511  -1760
-19248  14511  -1760
. . . . .
-19248  14511  -1616
-19248  14511  -1616
-19248  14495  -1616
-19248  14495  -1600
-19248  14495  -1600
    
```

Gambar.3. Isi File Teks

```

00000000: 42 49 4B 20 20 20 20 20|20 32 30 30 39 20 20 35 | BIK      2009 5
00000010: 20 20 36 20 4C 54 3D 20|20 20 39 20 53 54 3D 20 | 6 LT= 9 ST=
00000020: 20 31 20 73 65 63 20 0D|0A 48 3D 20 20 31 2E 31 | 1 sec H= 1.1
00000030: 34 65 2D 30 32 20 44 3D|20 20 31 2E 31 34 65 2D | 4e-02 D= 1.14e-
00000040: 30 32 20 5A 3D 20 20 31|2E 31 34 65 2D 30 32 20 | 02 Z= 1.14e-02
00000050: 6E 54 2F 4C 53 42 20 20|0D 0A 47 50 53 20 39 39 | nT/LSB HGPS 99
00000060: 39 39 0D 0A 1A 20 0A 2D|31 39 32 34 38 20 20 31 | 99 H -19248 1
00000070: 34 34 39 35 20 20 2D 31|37 34 34 0A 2D 31 39 32 | 4495 -1744 -192
00000080: 34 38 20 20 31 34 34 39|35 20 20 2D 31 37 36 30 | 48 14495 -1760
00000090: 0A 2D 31 39 32 34 38 20|20 31 34 34 39 35 20 20 | -19248 14495
000000A0: 2D 31 37 34 34 0A 2D 31|39 32 34 38 20 20 31 34 | -1744 -19248 14
000000B0: 35 31 31 20 20 2D 31 37|36 30 0A 2D 31 39 32 34 | 511 -1760 -1924
000000C0: 38 20 20 31 34 35 31 31|20 20 2D 31 37 36 30 0A | 8 14511 -1760
.....
000018C0: 36 31 36 0A 2D 31 39 32|34 38 20 20 31 34 34 39 | 616 -19248 1449
000018D0: 35 20 20 2D 31 36 31 36|0A 2D 31 39 32 34 38 20 | 5 -1616 -19248
000018E0: 20 31 34 34 39 35 20 20|2D 31 36 30 30 0A 2D 31 | 14495 -1600 -1
000018F0: 39 32 34 38 20 20 31 34|34 39 35 20 20 2D 31 36 | 9248 14495 -16
00001900: 30 30 | 00

```

Gambar.4. Peta Memori Pada File Teks

Penamaan File

Penamaan file dari data magnetometer hasil rekontruksi dikelompokkan kedalam empat buah tipe yaitu;

- File dengan tipe RS merupakan file bineri yang dibuat setiap lima menit dengan sampling data setiap 1 detik. Kapasitas file yang terbentuk sebesar 1908 byte.
- File dengan tipe RM merupakan file bineri yang dibuat setiap lima menit dengan sampling data setiap 1 menit. Kapasitas file yang terbentuk sebesar 138 byte.
- File dengan tipe S merupakan file harian yang dibuat setiap 24 jam dengan sampling data setiap 1 detik. Kapasitas file yang terbentuk sebesar 518508 byte.
- File dengan tipe M merupakan file harian yang dibuat setiap 24 jam dengan sampling data setiap 1 menit. Kapasitas file yang terbentuk sebesar 8748 byte.

Format penamaan file bineri data geomagnet adalah sebagai berikut:

DTYYMMDDHHNN.STN

DT : Tipe Data (RS, RM, S dan M)

YY : Tahun

MM : Bulan

DD : Tanggal

HH : Jam dengan format 24 jam

NN : Menit dengan kelipantan 5 menit

STN : Kode Stasiun Pengamatan dengan penulisan tiga digit. Untuk saat ini terdapat 7 lokasi pengamatan dengan kode sebagai berikut; (1) KTB, untuk stasiun Kototabang (Sumatera Barat), (2) TJS, untuk stasiun Tanjung sari (Jawa Barat), (3) PTN, untuk stasiun Pontianak (Kalimantan Barat), (4) BIK, untuk stasiun Biak (Papua), (5) MND, untuk stasiun Manado (Sulawesi Utara), (6) PRP, untuk stasiun Pare- Pare (Sulawesi Selatan), (7) KPG, untuk stasiun Kupang (Nusa Tenggara Timur).

Contoh penulisan file lima menitan dengan sampling 1 detik untuk data biak tanggal 6 Mei 2009 jam 05:00:00 adalah sebagai berikut : RS0905060500.BIK

Struktur File

Struktur dari file bineri yang dibangun terbagai atas dua bagian utama yaitu bagian header dan bagian data.

Byte No.	0x00	0x01	0x02	0x03	0x04	0x05	0x06	0x07
00000000	kode stasiun							
00000008	tahun (yyyy)					bulan (mm)		
00000010	tanggal (dd)				lokal time			
00000018	lokal time				Sampling			
00000020	Sampling							
00000028	H Sens							
00000030	H Sens					D Sens		
00000038	D Sens							
00000040	D Sens			Z Sens				
00000048	Z Sens							
00000050								
00000058	GPS							
00000060	pemisah							
00000068	Pemisah				H ₀ (L)	H ₀ (H)	D ₀ (L)	D ₀ (H)
00000070	Z ₀ (L)	Z ₀ (H)	H ₁ (L)	H ₁ (H)	D ₁ (L)	D ₁ (H)	Z ₁ (L)	Z ₁ (H)
00000768	Hn-1(L)	Hn-1(H)	Dn-1(L)	Dn-1(H)	Zn-1(L)	Zn-1(H)	Hn(L)	Hn(H)
00000770	Dn(L)	Dn(H)	Zn(L)	Zn(H)				

Gambar.8 Peta Memori File Magnetometer

Implementasi Sistem

Pemrosesan Data

Ada beberapa tahapan dalam proses pembuatan / rekontruksi file magnetometer yaitu:

1. Membaca komponen data (H,D,Z) yang dihasilkan magnetometer

Sebelum melakukan proses penyimpanan file yang perlu dilakukan adalah membaca data dan mempersiapkan nama filenya. Adapun proses baca data adalah sebagai berikut;

```

Procedure BacaData(Var Data1,Data2,Data3: Integer);
Var
  DataRun : Array [0..2] Of Word;
  Sts,Mux,Chanel: Byte;
  LValue,HValue,WValue: Integer;
Begin
  Mux:=3;
  Repeat
    Out32($37a,8 + Mux);
    Out32($378,255);
    Out32($37a,36 + Mux);
    Repeat
      Sts:= Inp32($379);
      Sts:= Sts And 128;
    Until(STS=128);
    LValue:= Inp32($378);
    HValue:= Inp32($379);
    HValue:= HValue And 120;
    HValue:= HValue Shl 5;
    WValue:= (HValue + LValue)*16;
    Chanel:= (Mux-3)*-1;
    DataRun[Chanel]:= WValue XOR 32768;
    Dec(mux);
    Out32($37a,0);
  Until (Mux=0);
  Data1:= DataRun[0];
  Data2:= DataRun[1];
  Data3:= DataRun[2];
End;

```

2. Membuka/mengaktifkan file

Sebelum file dapat diakses (dibaca atau ditulis), mula-mula file harus diaktifkan lebih dulu dengan menghubungkan variabel file dengan file yang sebenarnya. Untuk keperluan ini fungsi yang digunakan yaitu *AssignFile()*. Bentuk deklarasinya :

AssignFile(var FileData; FileName: string);

dimana: FileData merupakan variabel yang menyatakan tipe dari file

File Name merupakan nama dari file yang akan diaktifkan

Setelah variabel file terhubung dengan filenya, maka langkah berikutnya adalah menentukan operasi filenya; membuka dengan membuat file baru *Rewrite()*, atau membukan file yang sudah ada *Reset()*.

Contoh proses:

```
Var
  FileData: File;
Begin
  AssignFile(FileData, 'RS0905060525.BIK');
  Rewrite(FileData);
End
```

3. Melakukan proses penyimpanan data dan file

Ada dua proses dalam melakukan penyimpanan data yaitu (1), penentuan lokasi awal dari alamat memori,dengan bentuk deklarasi

Seek(var F; N: Longint);

dimana;

F : variabel file yang menghubungkan dengan lokasi dari file sebenarnya

N : posisi alamat memori dalam file

serta, (2) proses penyimpanan file itu sendiri, dengan bentuk deklarasi

BlockWrite(var f: File; var Buf; Count: Integer [; var AmtTransferred: Integer]);

dimana:

F : Variabelfile

Buf: variabel buffer yang digunakan untuk menampung data yang akan ditulis kedalam file.

Count: merupakan parameter yang menyatakan banyaknya record yang akan ditulis kedalam file.

4. Menutupfile

Apabila suatu file sudah tidak diproses lagi, file perlu ditutup. Hal ini sangat penting terutama jika melakukan pemrosesan file secara periodik. Untuk menutup file, fungsi yang digunakan adalah *closefile()*. Bentuk deklarasinya :

CloseFile(var F);

dimana adalah F: Variabel file

Contoh proses:

```
Var
  FileData: File;
Begin
  AssignFile(FileData, 'RS0905060525.BIK');
  Rewrite(FileData,1);
  Data[1]:= Data1;
  Data[2]:= Data2;
  Data[3]:= Data3;
  Seek(FileData,108);
  BlockWrite(FileData, Data,6);
  CloseFile(FileData);
End;
```

KESIMPULAN

Sistem rekonstruksi file ini telah berjalan dengan output berupa

1. Empat file biner dengan ukuran file yang lebih kecil dibandingkan dengan file sebelumnya dalam format teks.
2. File biner yang dihasilkan memiliki ukuran file yang tetap untuk setiap file yang dibuatnya.

DAFTAR PUSTAKA

Jeri R. Hanly and Elliot B. Koffman, Addison Wesley, 2002: *Text and Binary File Processing Problem Solving and Program Design in C.*,

Dale, Nell and Lilly, Susan C., 1988, *Pascal plus Data Structures, Algorithms and Advanced Programming*, D.C. Heath and Company, Lexington.

Cucu. E. H, 2006, *Pengembangan Sistem Akuisisi Data Fluxgate Magnetometer*.

Borland Developer Support Staff, 2000, *How to insert an InterBase BLOB in Delphi using LoadFromFile?*, <http://community.borland.com/article/0,1410,25364,00.htm>.

Borland Developer Support Staff, 2000, *How to retrieve an InterBase Blob in Delphi using SaveToFile?*, <http://community.borland.com/article/0,1410,25238,00.htm>.