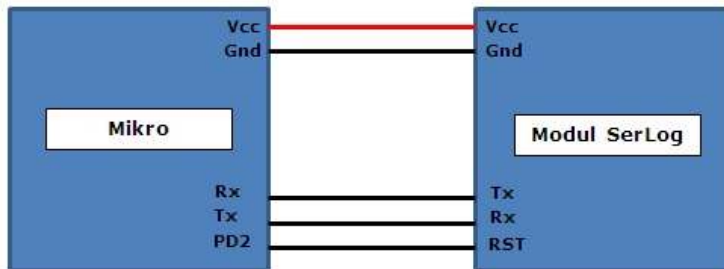


## Modul SerLog - Easy Serial Logger

Modul "**SerLog**" - **Easy Serial Logger**. Modul ini diaplikasikan dalam project "Data Logger". Anda dapat melakukan pencatatan dan pembacaan data melalui memory card. Cara mengaksesnya cukup via serial saja.

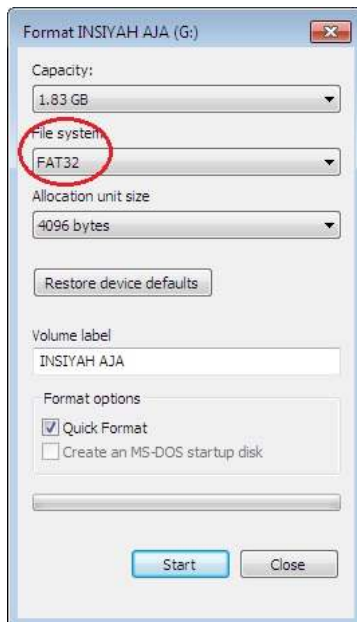
Berikut adalah gambar koneksinya :



Untuk bisa berkomunikasi dengan mikrokontroler, modul ini menggunakan 3 buah pin. Pin Tx-Rx dimana pemasangannya secara cross (silang). Rx mikrokontroler dihubungkan ke Tx SerLog, Tx mikrokontroler dihubungkan ke Rx SerLog. Lalu ditambah lagi 1 pin (bisa dari port mana saja yang kosong) ini digunakan untuk pin reset modul serlog. Kondisi logic "0" pada pin PD2 (pada gambar diatas) akan membuat modul SerLog ter-reset. Sedangkan kondisi logic "1" pada pin PD2 akan membuat modul SerLog berada pada mode "ready".

Untuk bisa menggunakan modul ini dengan baik, perhatikan langkah2 berikut ini :

1. Memory Card (Jenis microSD), di format FAT32, seperti gambar di bawah ini.



2. Setelah di format, langkah berikutnya adalah membuat file config pada memory card nya dengan cara sbb :

Buka program NOTEPAD, lalu ketikkan : 9600,13,1,2,1,1,1 kemudian simpan ke memory cardnya dengan nama **CONFIG.txt** atau seperti pada gambar dibawah ini





Project :  
Version :  
Date : 11/4/2014  
Author :  
Company :  
Comments:

Chip type : ATmega16  
Program type : Application  
AVR Core Clock frequency: 8.000000 MHz  
Memory model : Small  
External RAM size : 0  
Data Stack size : 256

\*\*\*\*\*/

```
#include <mega16.h>
```

```
#include <stdio.h>
```

```
#include <delay.h>
```

```
#define rst_serlog PORTD.2
```

```
// Declare your global variables here
```

```
void reset(void)
```

```
{
```

```
    delay_ms(1000);
```

```
    rst_serlog=0;
```

```
    delay_ms(1000);
```

```
    rst_serlog=1;
```

```
    delay_ms(100);
```

```
}
```

```
void init(void)
```

```
{
```

```
    printf("init\r\n");
```

```
    delay_ms(5000);
```

```
}
```

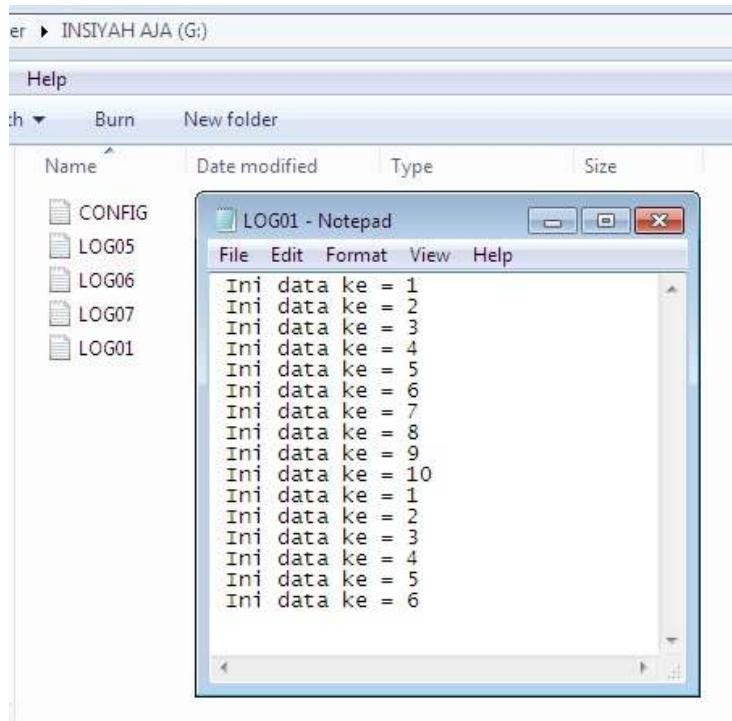
```
void main(void)
{
// Declare your local variables here
int i;
i=0;
PORTD.2=1;
DDRD.2=1;

// USART initialization
// Communication Parameters: 8 Data, 1 Stop, No Parity
// USART Receiver: On
// USART Transmitter: On
// USART Mode: Asynchronous
// USART Baud Rate: 9600
UCSRA=0x00;
UCSRB=0x18;
UCSRC=0x86;
UBRRH=0x00;
UBRRL=0x33;

while (1)
{
// Place your code here
reset();
init();
for (i=1;i<11;i++)
{
printf("writes log01.txt\r\n");
delay_ms(1000);
printf("Ini data ke = %d\r\n",i);
delay_ms(200);
printf("\r\n");
delay_ms(2000);
}
}
}
```

Selanjutnya cek apakah data yang kita tuliskan tadi sudah masuk ke memory card atau belum. Caranya sbb:

Lepas memory cardnya, lalu pasang ke card reader, dan hubungkan ke PC/laptop. Jika tidak ada masalah, kita akan melihat data hasil percobaan tadi seperti ini :



OK, jadi data - data yang kita tulis tadi sudah berhasil masuk ke memory cardnya.

Jika pada aplikasi data logger anda memerlukan prosedur baca data yang sudah disimpan ke memory card tadi, caranya sbb:

#### **Contoh dalam bahasa Bascom AVR**

```
'////////// Baca data  
Print "read " ; File_name ; ".txt"
```

#### **Contoh dalam bahasa Codevision C AVR**

```
////////// Baca data  
printf("read log01.txt\r\n");
```