

Menggunakan Internal EEPROM ATMEGA 128

ATMEGA 128 (AVR) adalah mikrokontroler yang telah dilengkapi EEPROM sebesar 4Kbyte didalam chipnya. EEPROM ini terpisah dari flash memory AVR, dan dapat ditulis, baca per byte. Untuk dapat mengakses EEPROM ini perlu mengatur register-register dan fuse bit. Untuk mencegah kesalahan dalam penulisan dan pembacaan EEPROM diusahakan power supply yang digunakan cukup stabil.

Saat pembacaan EEPROM mikrokontroler berhenti selama 4 cycle clock sebelum mengerjakan perintah berikutnya dan untuk menulis EEPROM mikrokontroler berhenti selama 2 cycle clock sebelum mengerjakan perintah berikutnya.

Register-register yang perlu diatur adalah :

EEPROM address register

Bit	15	14	13	12	11	10	9	8	
	-	-	-	-	EEAR11	EEAR10	EEAR9	EEAR8	EEARH
	EEAR7	EEAR6	EEAR5	EEAR4	EEAR3	EEAR2	EEAR1	EEAR0	EEARL
	7	6	5	4	3	2	1	0	
Read/Write	R	R	R	R	R/W	R/W	R/W	R/W	
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	X	X	X	X	
	X	X	X	X	X	X	X	X	

Register ini berfungsi untuk menentukan alamat pembacaan dan penulisan data pada EEPROM.

1. Bit 15-12 tidak digunakan.
2. Bit 11-0 alamat EEPROM, EEPROM address register ini dapat diakses secara linear dari 0-4096.

EEPROM data register

Bit	7	6	5	4	3	2	1	0	
	MSB							LSB	EEDR
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

Register ini adalah buffer dari EEPROM, saat menulis register ini berisi data yang akan ditulis pada alamat yang diberikan pada register EEARH dan EEARL, saat membaca register ini berisi data yang dibaca.

EEPROM control register

Bit	7	6	5	4	3	2	1	0	
	-	-	-	-	EERIE	EEMWE	EEWE	EERE	EECR
Read/Write	R	R	R	R	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	X	0	

1. Bit 7-4 tidak digunakan
2. EERIE (EEPROM Ready Interrupt Enable). Jika EERIE '1' mengaktifkan EEPROM ready interrupt tetapi terlebih dahulu mengaktifkan bit I dalam register SREG, jika EERIE '0' maka interrupt tidak aktif. Interrupt ini terjadi saat bit EEWE '0'.
3. EEMWE (EEPROM Master Write Enable). Ketika bit ini '1' dan diikuti dengan bit EEWE '1' selama 4 cycle clock akan menulis data ke EEPROM sesuai alamat yang telah diberikan. Jika EEMWE '0' maka bit EEWE tidak akan berpengaruh.
4. EEWE (EEPROM Write Enable) EEWE adalah sinyal write strobe. Saat data dan alamat sudah benar bit EEWE harus di set '1' untuk menulis data ke EEPROM. Jika tidak maka EEPROM tidak akan ditulis. Perlu diperhatikan, tidak boleh terjadi interrupt saat dilakukan penulisan dan pembacaan EEPROM karena data menjadi tidak valid.
5. EERE (EEPROM Read Enable) EERE adalah sinyal read strobe. Saat alamat data EEPROM yang mau dibaca sudah benar, EERE harus diset '1' untuk mengaktifkan pembacaan data EEPROM.

Setelah memahami register-register yang diperlukan untuk menggunakan EEPROM. Dibawah ini merupakan contoh procedure untuk menulis data pada internal EEPROM avr.

```
void EEPROM_write(unsigned int uiAddress, unsigned char ucData)
{
    cli(); // mematikan global interrupt
    while(EECR&(1<<EEWE)); //menunggu sampai operasi sebelumnya selesai
    EEAR=uiAddress; //menyiapkan alamat eeprom yang akan ditulis
    EEDR=ucData; //mengisi register dengan data yang akan diisi
}
```

DELTA ELECTRONIC

www.delta-electronic.com

www.robotindonesia.com

www.deltakits-sby.com

```

EECR|=(1<<EEMWE); //mengaktifkan master write enable
EECR|=(1<<EEWE); //memberi sinyal strobe untuk menulis
sei(); //mengaktifkan kembali global interrupt
}

```

Variable uiAddress adalah alamat data yang akan ditulis, variable ucData adalah data yang ditulis pada eeprom. Contoh procedure dibawah ini adalah untuk membaca data eeprom.

```

unsigned char EEPROM_read(unsigned int uiAddress)
{
    cli(); //menonaktifkan global interrupt
    while(EECR&(1<<EEWE)); //menunggu sampai operasi sebelumnya
    selesai
    EEAR=uiAddress; //menyiapkan alamat eeprom yang dibaca
    EECR|=(1<<EERE); //memberi sinyal strobe untuk membaca
    sei(); //mengaktifkan kembali global interupt
    return EEDR; //mengembalikan data yang dibaca
}

```

Hanya dengan 2 prosedur diatas sudah dapat melakukan penulisan dan pembacaan eeprom pada avr.

Ada satu hal lagi yang penting yaitu fuse bit EESAVE, fuse bit ini akan mengatur apakah pada saat mendownload program ke flash, eeprom akan dihapus isinya atau tidak. Jika fuse bit ini diprogram maka eeprom tidak dihapus pada saat mendownload program ke avr demikian sebaliknya. **Januar, Delta Electronic**