

# Pengantar Modul 5

## Interrupt

### Teori

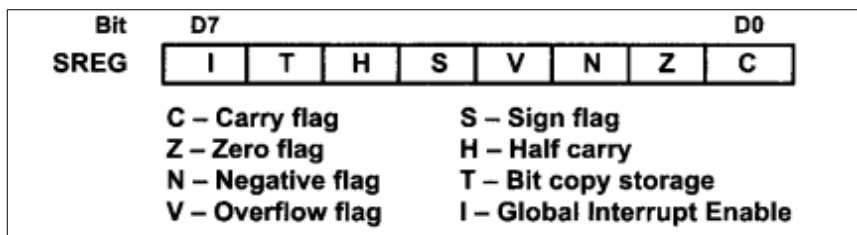
Interrupt merupakan suatu istilah yang digunakan pada mikrokontroller untuk mengintrupsi suatu program yang sedang berjalan. Pada saat interrupt diaktifkan (semisal menggunakan INT0 *external interrupt*), program utama akan seketika dihentikan sementara, dan program interrupt diaktifkan. Mikrokontroller akan menjalankan program INT0 yang berada pada register \$0002 hingga selesai, kemudian berlanjut hingga program pada register \$003A. Jika semua program interrupt telah berakhir program utama akan kembali dijalankan pada kondisi terakhir diaktifkannya interrupt.

### Mengaktifkan Interrupt

Ada beberapa register yang harus diaktifkan untuk menggunakan interrupt, yaitu diantaranya:

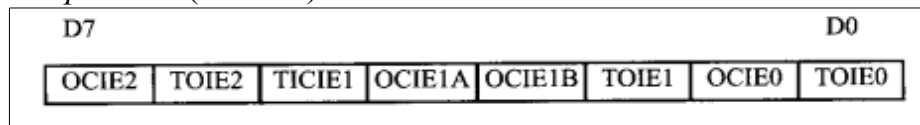
- *Status register (SREG)*

Pada bit D7 merupakan pengaturan *Global Interrupt Enable*, dengan memberikan nilai biner 1 pada bit tersebut, semua jenis *interrupt* siap untuk diaktifkan.



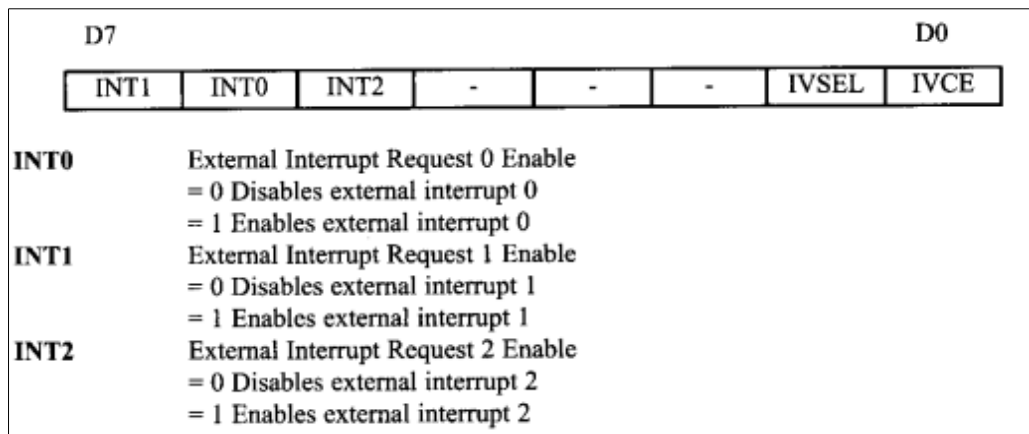
**Gambar 1. SREG**

- *Timer Interrupt Mask (TIMSK)*



**Gambar 2. TIMSK**

- *General Interrupt Control Register (GICR)*



**Gambar 3. GICR**

## Jenis-jenis *Interrupt* pada ATmega128

Untuk melihat jenis *interrupt* apa saja yang ada pada ATmega128, dapat melihat pada datasheet (halaman 59).

Vector No.	Program Address <sup>(2)</sup>	Source	Interrupt Definition
1	\$0000 <sup>(1)</sup>	RESET	External Pin, Power-on Reset, Brown-out Reset, Watchdog Reset, and JTAG AVR Reset
2	\$0002	INT0	External Interrupt Request 0
3	\$0004	INT1	External Interrupt Request 1
4	\$0006	INT2	External Interrupt Request 2
5	\$0008	INT3	External Interrupt Request 3
6	\$000A	INT4	External Interrupt Request 4
7	\$000C	INT5	External Interrupt Request 5
8	\$000E	INT6	External Interrupt Request 6
9	\$0010	INT7	External Interrupt Request 7
10	\$0012	TIMER2 COMP	Timer/Counter2 Compare Match
11	\$0014	TIMER2 OVF	Timer/Counter2 Overflow
12	\$0016	TIMER1 CAPT	Timer/Counter1 Capture Event
13	\$0018	TIMER1 COMPA	Timer/Counter1 Compare Match A
14	\$001A	TIMER1 COMPB	Timer/Counter1 Compare Match B
15	\$001C	TIMER1 OVF	Timer/Counter1 Overflow
16	\$001E	TIMER0 COMP	Timer/Counter0 Compare Match
17	\$0020	TIMER0 OVF	Timer/Counter0 Overflow
18	\$0022	SPI, STC	SPI Serial Transfer Complete
19	\$0024	USART0, RX	USART0, Rx Complete
20	\$0026	USART0, UDRE	USART0 Data Register Empty
21	\$0028	USART0, TX	USART0, Tx Complete
22	\$002A	ADC	ADC Conversion Complete
23	\$002C	EE READY	EEPROM Ready
24	\$002E	ANALOG COMP	Analog Comparator
25	\$0030 <sup>(3)</sup>	TIMER1 COMPC	Timer/Counter1 Compare Match C
26	\$0032 <sup>(3)</sup>	TIMER3 CAPT	Timer/Counter3 Capture Event
27	\$0034 <sup>(3)</sup>	TIMER3 COMPA	Timer/Counter3 Compare Match A
28	\$0036 <sup>(3)</sup>	TIMER3 COMPB	Timer/Counter3 Compare Match B
29	\$0038 <sup>(3)</sup>	TIMER3 COMPC	Timer/Counter3 Compare Match C
30	\$003A <sup>(3)</sup>	TIMER3 OVF	Timer/Counter3 Overflow

**Gambar 4.** *Reset dan interrupt vector ATmega128*

## Program *Interrupt* Dengan Bahasa C

```
#include <mega128.h>
#include <alcd.h>
#include <delay.h>
#include <stdio.h>
#include <math.h>
```

```
// inisialisasi variabel char, int, float , string dls
```

```
interrupt [EXT_INT0] void ext_int0_isr(void)
{ //Masukan program Interrupt INT0 }
```

```

interrupt [EXT_INT1] void ext_int1_isr(void)
{ //Masukan program Interrupt INT1}

void main(void)
{
DDRE = 0xFF; // Untuk menentukan port output dan input
DDRA=0xFF;
PORTA=0x00;
DDRB=0xFF;

// External Interrupt(s) initialization
// INT0: On
// INT0 Mode: Falling Edge
// INT1: On
// INT1 Mode: Falling Edge
EICRA=0x0A; // Mengaktifkan interrupt
EICRB=0x00;
EIMSK=0x03;
EIFR=0x03;

// Alphanumeric LCD initialization
// Connections specified in the
// Project|Configure|C Compiler|Libraries|Alphanumeric LCD menu:
// RS - PORTC Bit 0
// RD - PORTC Bit 2
// EN - PORTC Bit 1
// D4 - PORTC Bit 4
// D5 - PORTC Bit 5
// D6 - PORTC Bit 6
// D7 - PORTC Bit 7
// Characters/line: 16
lcd_init(16); // Mengaktifkan tampilan di LCD

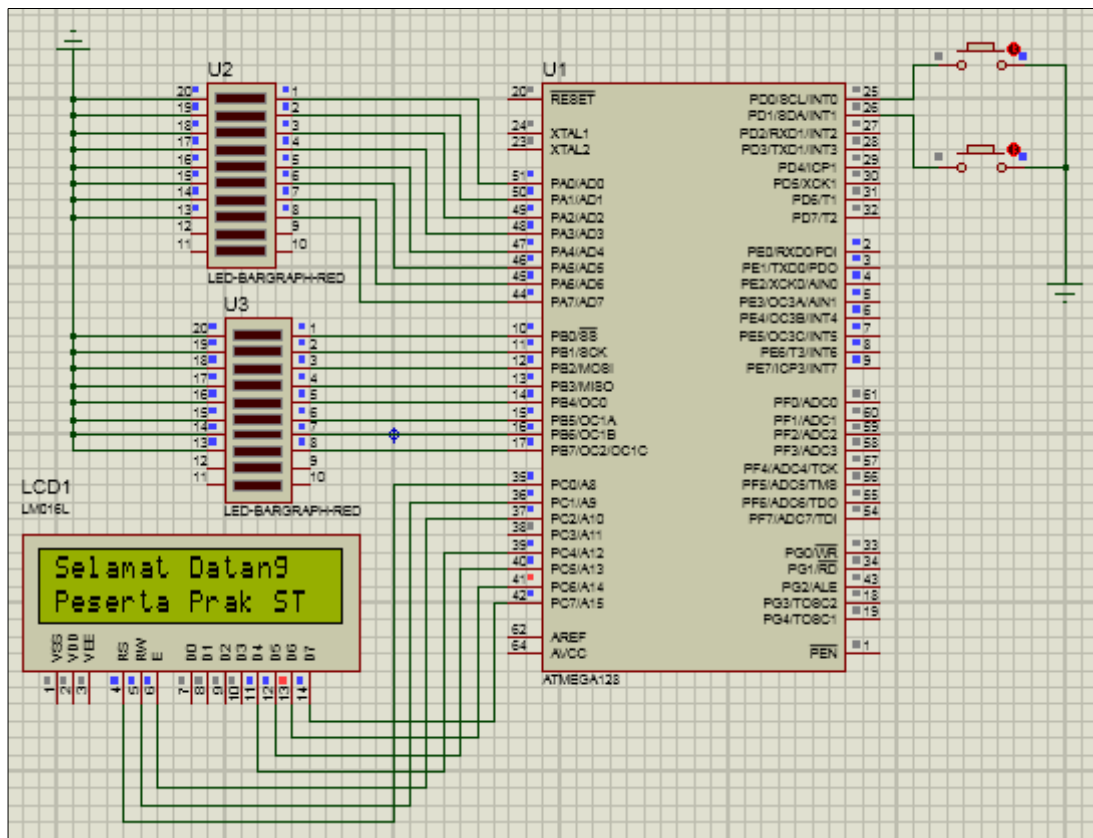
// Global enable interrupts
#pragma asm("sei") // Global interrupt aktif (ini wajib)

while (1)
{ // Masukkan program utama }
}

```

\*NOTE ( Wajib ) :

- Dalam penulisan LP silahkan dibuat contoh kode program interrupt, disimulasikan di proteus dan dijelaskan per bagian dari *code* (misalnya apa yang dimaksud dengan EICRA = 0x0A, EIMSK=0x03 dls)!
- Di bagian teori dasar, tambahkan dan jelaskan jenis-jenis *interrupt* dan fungsinya serta pada register apa yang harus diaktifkan untuk mengaktifkan program *interrupt* !
- Rangkaian Proteus tidak boleh sama dengan yang di modul ini !



Gambar 5 . Contoh Rangkaian untu program *interrupt* (INT0 dan INT1)

## REFERENSI

Mazidi, Muhammad Ali. 2011. The Microcontroller and Embedded System: Using Assembly and C. Pearson Education, inc: New Jersey

\_\_\_\_\_, *8-bit Atmel Microcontroller with 128KBytes in-System Programmable Flash- ATmega128 and ATmega128L*, [pdf], (<http://www.atmel.com/images/doc2467> [pdf] diakses tanggal 19 Oktober 2016)