Mechatronics Engineering

Interrupts

Prof. Ayman A. El-Badawy
Department of Mechatronics Engineering
Faculty of Engineering and Materials Science
German University in Cairo

Pointers and Addresses

- This artificial sequence shows how to declare a pointer and how to use & and *:

```
int x = 1, y = 2, z[10];
int *ip;              /* ip is a pointer to int */
ip = &x;              /* ip now points to x */
y = *ip;              /* y is now 1 */
*ip = 0;              /* x is now 0 */
ip = &z[0];           /* ip now points to z[0] */
```

- The declaration of x, y, and z are what we've seen all along. The declaration of the pointer ip,

```
int *ip;
```

Pointers and Function Arguments

- Since C passes arguments to functions by value, there is no direct way for the called function to alter a variable in the calling function. For instance, a sorting routine might exchange two out-of-order arguments with a function called swap. It is not enough to write

```
swap(a, b);
```

where the swap function is defined as

```
void swap(int x, int y)     /* WRONG */
{
int temp;
temp = x;
x = y;
y = temp;
}
```

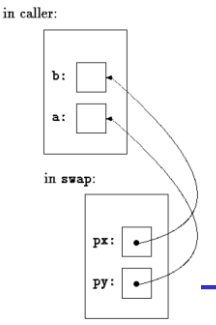
- Because of call by value, `swap` can't affect the arguments a and b in the routine that called it.
- The function above swaps *copies* of a and b.

- The way to obtain the desired effect is for the calling program to pass *pointers* to the values to be changed:

```
swap(&a, &b);
```

- Since the operator & produces the address of a variable, &a is a pointer to a.
- In swap itself, the parameters are declared as pointers, and the operands are accessed indirectly through them.

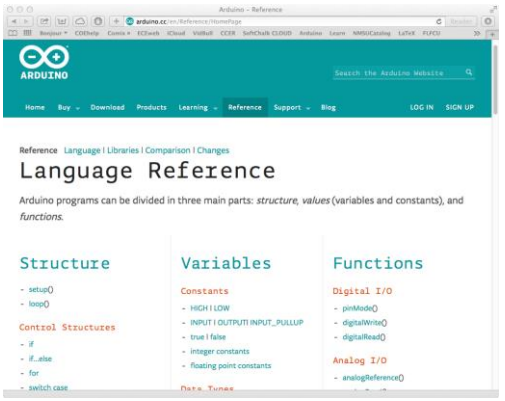
```
void swap(int *px, int *py)  /* interchange *px and *py */
{
   int temp;
   temp = *px;
   *px = *py;
   *py = temp;
}
```



GUC German University in Cairo
Prof. Ayman A. El-Badawy
Department of Mechatronics Engineering
Faculty of Engineering and Material Science

Blink Program (IDE)

```
int ledPin = 13;
void setup ()
{
   pinMode(ledPin , OUTPUT);
}
void loop ()
{
   digitalWrite(ledPin , HIGH);
   delay (1000);
   digitalWrite(ledPin , LOW);
   delay (1000);
}
```



Speed Test: LED Flash

Arduino IDE

```
void setup()
{
   pinMode(13, OUTPUT);
}
void loop()
{
digitalWrite(13, HIGH);
digitalWrite(13, LOW);
}
```

Frequency = 121 kHz

C/C++

```
void setup()
{
   pinMode(13, OUTPUT);
}
void loop()
{
  while(1)
  {
    PORTB |= 0x20; // ON
    PORTB &= ~0x20;// OFF
  }
}
```

Frequency = 2.66 MHz

Blink Program (C)

```
void MyDelay (unsigned long mSecondsApx );
int main()
{
    unsigned char *portDDRB ;
    portDDRB = (unsigned char *) 0x24;
    *portDDRB |= 0x20;
    While (1)
    {
      unsigned char *portB ;
      portB = (unsigned char *)  0x25;
      *portB |= 0x20;
      MyDelay (1000);
      *portB &= 0xDF;
      MyDelay (1000);
    }
    Return 0;
}
void MyDelay (unsigned long mSecondsApx )
{
    volatile unsigned long i;
    unsigned long endTime = 1000 * mSecondsApx ;
    for (i = 0; i < endTime ; i++);
}
```

GUC German University in Cairo
Prof. Ayman A. El-Badawy
Department of Mechatronics Engineering
Faculty of Engineering and Material Science

Blink Program (C)

```
void MyDelay (unsigned long mSecondsApx );
void setup ()
{
    unsigned char *portDDRB ;
    portDDRB = (unsigned char *) 0x24;
    *portDDRB |= 0x20;
}
void loop ()
{
    unsigned char *portB ;
    portB = (unsigned char *)  0x25;
    *portB |= 0x20;
    MyDelay (1000);
    *portB &= 0xDF;
    MyDelay (1000);
}
void MyDelay (unsigned long mSecondsApx )
{
    volatile unsigned long i;
    unsigned long endTime = 1000 * mSecondsApx
;
    for (i = 0; i < endTime ; i++);
}
```

GUC German University in Cairo
Prof. Ayman A. El-Badawy
Department of Mechatronics Engineering
Faculty of Engineering and Material Science

Polling Vs. Interrupt

- Polling
  - ➢Ties down the CPU
- Interrupt
  - ➢Efficient CPU use
  - ➢Has priority
  - ➢Can be masked

```
while (true)
{
  if(PIND.2 == 0)
        //do something;
}
```

```
main( )
{
  Do your common task
}

whenever PIND.2 is 0 then
   do something
```


Interrupt vectors in ATmega328

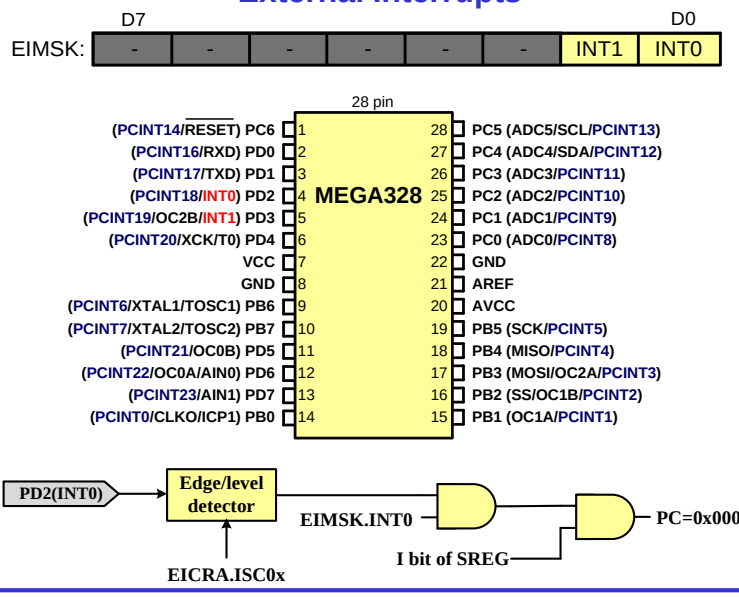
Interrupt	Address (hex)
Reset	0000
External Interrupt Request 0	0002
External Interrupt Request 1	0004
Pin Change Interrupt Request 0	0006
Pin Change Interrupt Request 1	0008
Pin Change Interrupt Request 2	000A
Watchdog Time-out Interrupt	000C
Timer/Counter2 Compare Match A	000E
Timer/Counter2 Compare Match B	0010
Timer/Counter2 Overflow	0012
Timer/Counter1 Capture Event	0014
Timer/Counter1 Compare Match A	0016
Timer/Counter1 Compare Match B	0018
Timer/Counter1 Overflow	001A
Timer/Counter0 Compare Match A	001C
Timer/Counter0 Compare Match B	001E
Timer/Counter0 Overflow	0020
SPI Serial Transfer Complete	0022

Table 3: Interrupt Vector Name for the ATmega32/ATmega16 in WinAVR

Interrupt	Vector Name in WinAVR
External Interrupt request 0	INT0_vect
External Interrupt request 1	INT1_vect
External Interrupt request 2	INT2_vect
Time/Counter2 Compare Match	TIMER2_COMP_vect
Time/Counter2 Overflow	TIMER2_OVF_vect
Time/Counter1 Capture Event	TIMER1_CAPT_vect
Time/Counter1 Compare Match A	TIMER1_COMPA_vect
Time/Counter1 Compare Match B	TIMER1_COMPB_vect
Time/Counter1 Overflow	TIMER1_OVF_vect
Time/Counter0 Compare Match	TIMER0_COMP_vect
Time/Counter0 Overflow	TIMER0_OVF_vect
SPI Transfer complete	SPI_STC_vect
USART, Receive complete	USART0_RX_vect
USART, Data Register Empty	USART0_UDRE_vect
USART, Transmit Complete	USART0_TX_vect
ADC Conversion complete	ADC_vect
EEPROM ready	EE_RDY_vect
Analog Comparator	ANALOG_COMP_vect
Two-wire Serial Interface	TWI_vect
Store Program Memory Ready	SPM_RDY_vect

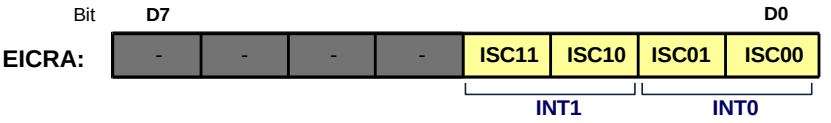
GUC German University in Cairo
Prof. Ayman A. El-Badawy
Department of Mechatronics Engineering
Faculty of Engineering and Material Science

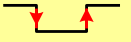
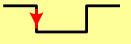
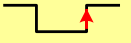
External Interrupts



GUC German University in Cairo
Prof. Ayman A. El-Badawy
Department of Mechatronics Engineering
Faculty of Engineering and Material Science

Edge trigger Vs. Level trigger in external interrupts

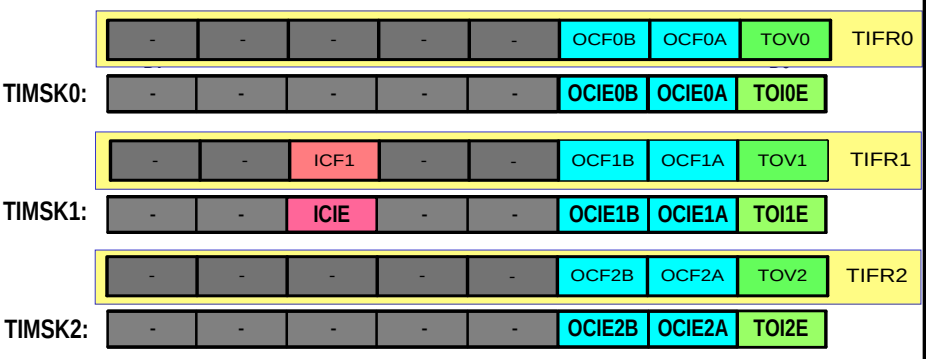


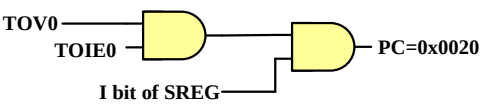
ISCx1	ISCx0	
0	0	
0	1	
1	0	
1	1	

```
EICRA = 0x02;  //INT0 on falling edges
```


GUC German University in Cairo
Prof. Ayman A. El-Badawy
Department of Mechatronics Engineering
Faculty of Engineering and Material Science

Timer Interrupts

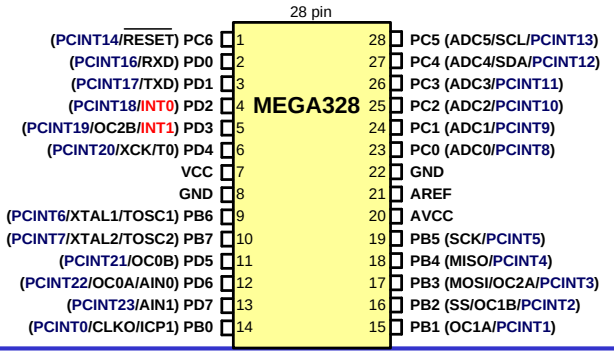




GUC German University in Cairo
Prof. Ayman A. El-Badawy
Department of Mechatronics Engineering
Faculty of Engineering and Material Science

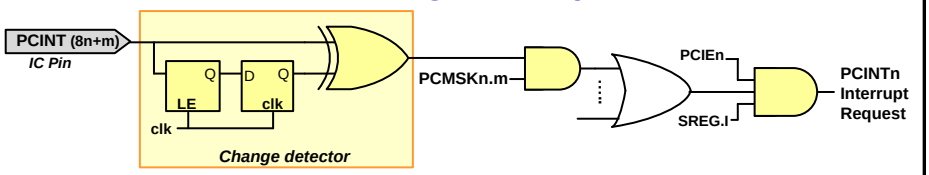
Pin Change Mask Registers

Bit	D7						D0	
PCMSK0: (for PORTB)	PCINT7	PCINT6	PCINT5	PCINT4	PCINT3	PCINT2	PCINT1	PCINT0	0x6B
PCMSK1: (for PORTC)	PCINT15	PCINT14	PCINT13	PCINT12	PCINT11	PCINT10	PCINT9	PCINT8	0x6C
PCMSK2: (for PORTD)	PCINT23	PCINT22	PCINT21	PCINT20	PCINT19	PCINT18	PCINT17	PCINT16	0x6D



Prof. Ayman A. El-Badawy
Department of Mechatronics Engineering
Faculty of Engineering and Material Science

Pin Change Interrupt



Bit	D7						D0	
PCICR:	-	-	-	-	-	PCIE2	PCIE1	PCIE0	0x68

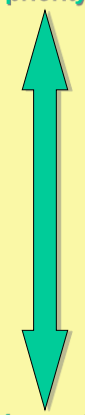
PCIE0: Pin Change Interrupt Enable bit for PORTB
PCIE1: Pin Change Interrupt Enable bit for PORTC (0: disabled, 1: enabled)
PCIE2: Pin Change Interrupt Enable bit for PORTD (0: disabled, 1: enabled)

Bit	D7						D0	
PCMSK0: (for PORTB)	PCINT7	PCINT6	PCINT5	PCINT4	PCINT3	PCINT2	PCINT1	PCINT0	0x6B
PCMSK1: (for PORTC)	PCINT15	PCINT14	PCINT13	PCINT12	PCINT11	PCINT10	PCINT9	PCINT8	0x6C
PCMSK2: (for PORTD)	PCINT23	PCINT22	PCINT21	PCINT20	PCINT19	PCINT18	PCINT17	PCINT16	0x6D

Prof. Ayman A. El-Badawy
Department of Mechatronics Engineering
Faculty of Engineering and Material Science

Interrupt priority

Interrupt	Address (hex)
Reset	0000
External Interrupt Request 0	0002
External Interrupt Request 1	0004
Pin Change Interrupt Request 0	0006
Pin Change Interrupt Request 1	0008
Pin Change Interrupt Request 2	000A
Watchdog Time-out Interrupt	000C
Timer/Counter2 Compare Match A	000E
Timer/Counter2 Compare Match B	0010
Timer/Counter2 Overflow	0012
Timer/Counter1 Capture Event	0014
Timer/Counter1 Compare Match A	0016
Timer/Counter1 Compare Match B	0018
Timer/Counter1 Overflow	001A
Timer/Counter0 Compare Match A	001C
Timer/Counter0 Compare Match B	001E
Timer/Counter0 Overflow	0020
SPI Serial Transfer Complete	0022
USART Rx Complete	0024
USART Data Register Empty	0026
USART Tx Complete	0028
ADC Conversion Complete	002A
EEPROM ready	002C
Analog Comparator	002E



Interrupt inside an interrupt

- The I flag is cleared when the AVR begins to execute an ISR. So, interrupts are disabled.
- The I flag is set when RETI is executed.

GUC German University in Cairo
Prof. Ayman A. El-Badawy
Department of Mechatronics Engineering
Faculty of Engineering and Material Science

C programming

- Using Timer0 generate a square wave on pin PORTB.5, while at the same time transferring data from PORTC to PORTD.

```
#include "avr/io.h"
#include "avr/interrupt.h"
int main ()
{
    DDRB |= (1<<5);     //DDRB.5 = output
    TCNT0 = -32;        //timer value for 2 µs
    TCCR0A = 0x00;
    TCCR0B = 0x01;              //Normal mode, int clk, no prescaler
    TIMSK0 = (1<<TOIE0);        //enable Timer0 overflow interrupt
    sei ();                     //enable interrupts
    DDRC = 0x00;                //make PORTC input
    DDRD = 0xFF;                //make PORTD output
    while (1)           //wait here
        PORTD = PINC;
}
ISR (TIMER0_OVF_vect) //ISR for Timer0 overflow
{
    TCNT0 = -32;
    PORTB ^= 0x20;      //toggle PORTB.5
}
```

```
sei ( );     //set I
cli  ( );    //clear
```

Prof. Ayman A. El-Badawy
Department of Mechatronics Engineering
Faculty of Engineering and Material Science

C programming Example 2

- Using Timer1 and CTC mode write a program that toggles pin PORTB.5 every second, while at the same time transferring data from PORTC to PORTD. Assume XTAL = 16 MHz.

```
#include <avr/io.h>
#include <avr/interrupt.h>
int main () {
        DDRB |= (1<<5);         //make DDRB.5 output

        OCR1A = 15624;
        TCCR1A = 0x00; //CTC mode, internal clk, prescaler=1024
        TCCR1B = 0x0D;
        TIMSK1 = (1<<OCIE1A);  //enable Timer1 compare match A int.
        sei ();                 //enable interrupts

        DDRC = 0x00;            //make PORTC input
        DDRD = 0xFF;            //make PORTD output
        while (1)               //wait here
                PORTD = PINC;
}
ISR (TIMER1_COMPA_vect) {       //ISR for Timer1 compare match A
        PORTB ^= (1<<5);        //toggle PORTB.5
}
```

Prof. Ayman A. El-Badawy
Department of Mechatronics Engineering
Faculty of Engineering and Material Science

C programming Example 3

- Assume that the INT0 pin is connected to a switch that is normally high. Write a program that toggles PORTB.5, whenever INT0 pin goes low.

```
#include <avr/io.h>
#include <avr/interrupt.h>
int main ()
{
        DDRB = 1<<5;            //PB5 as an output
        PORTD = 1<<2;           //pull-up activated
        EICRA = 0x2;            //make INT0 falling edge triggered

        EIMSK = (1<<INT0);      //enable external interrupt 0
        sei ();                 //enable interrupts

        while (1);                      //wait here
}

ISR (INT0_vect)                         //ISR for external interrupt 0
{
        PORTB ^= (1<<5);                //toggle PORTB.5
}
```

GUC German University in Cairo
Prof. Ayman A. El-Badawy
Department of Mechatronics Engineering
Faculty of Engineering and Material Science