```
#include <avr/interrupt.h>
#include <avr/io.h>
// define the global variables that can be used in every function  ==========
volatile unsigned char ADC_result;
volatile unsigned int ADC_result_flag;
void main()
{
        cli();  // disable all of the interrupt ==========================

         // config the external interrupt ======================================
        EIMSK |= (_BV(INT2));  // enable INT2
        EICRA |= (_BV(ISC21) | _BV(ISC20));                  // rising edge interrupt
        // config ADC ==========================================================
        // by default, the ADC input (analog input is set to be ADC0 / PORTF0
        ADCSRA |= _BV(ADEN); // enable ADC
        ADCSRA |= _BV(ADIE); // enable interrupt of ADC
        ADMUX |= _BV(ADLAR) | _BV(REFS0);

        // set the PORTA as output to display the ADC result ==================
        DDRA  = 0xff;
        // sets the Global Enable for all interrupts ==========================
        sei();
        // initialize the ADC, start one conversion at the beginning ==========
        ADCSRA |= _BV(ADSC);
        while (1)
        {
                if (ADC_result_flag)
                {
                        PORTA = ADC_result;
                        ADC_result_flag = 0x00;
                }
        }
}
// sensor 3: 2nd Optical Inductive, Active HIGH starts AD converstion =======
 ISR(INT2_vect)
{
        // when there is a rising edge, we need to do ADC =====================
        ADCSRA |= _BV(ADSC);
}
// the interrupt will be trigured if the ADC is done =======================
ISR(ADC_vect)
{
        ADC_result = ADCH;
        ADC_result_flag = 1;
}
```