```c
/*
 * asynchronous_sw_read_interrupt.c
 *
 * Created: 3/18/2021 6:03:31 PM
 * Author : Judah Ben-Eliezer
 */

#define BAUD_RATE 4800UL                                    //  baud rate.
#define F_CPU 4000000UL                                     //  4MHz clock.

#include <avr/io.h>
#include <util/delay.h>
#include <avr/interrupt.h>

uint8_t USART_sw_read();                                    //  declaration for read function.

int main(void)
{
    PORTB.DIRCLR = PIN1_bm;                                 //  set PB1 as input.
    PORTB.PIN1CTRL |= PORT_ISC_FALLING_gc;                  //  enable interrupt on falling edge.
    sei();                                                  //  enable global interrupts.

    while (1)
    {
        asm volatile ("nop");                               //  nop to avoid optimization deletion.
    }
}

ISR (PORTB_PORT_vect) {
    c = USART_sw_read();                                    //  call USART_sw_read.
    PORTB.INTFLAGS |= PIN1_bm;                              //  clear interupt.
}

uint8_t USART_sw_read() {

    uint8_t d;                                              //  bit time.
    if (BAUD_RATE == 4800UL) {
        d = 48;
    } else if (BAUD_RATE == 9600UL) {
```

```
40          d = 99;
41      } else if (BAUD_RATE == 19200UL) {
42          d = 201;
43      } else return 0x00;
44
45      uint8_t data = 0;
46
47      _delay_us(d/2);
48      if ((PORTB_IN & PIN1_bm) != 0) return 0x00;                    //  ↵
          check for false start.
49      _delay_us(d);                                                  //  ↵
          delay for bit time.
50
51      uint8_t i;
52      for (i = 0; i < 8; ++i) {
53          data >>= data | ((PORTB_IN | PIN1_bm) << 6);               //  ↵
              read little endian input into data.
54          _delay_us(d);                                              //  ↵
              delay for bit time.
55      }
56
57      return data;
58  }
59
60
```