

# The Phenomenal Doorbell

## Bilaga 2. Källkod

---

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

//Variabler
int flag = 0;
int count = 0;
int timeVect[8];
int bellbutton = 0;
int waitbutton = 0;
int comeinbutton = 0;
int time = 0;

void setup() {
    DDRA = 0b00001111;
    DDRB = 0b11111111;
    DDRC = 0b00000000;
    DDRD = 0b11100000;
    disp_initial();
    disp_welcomemessage();
    enableButtonInterrupt();
    enableTimeInterrupt();
    sei();
}

void set_pin(char port, char pin, char state){

    char set = _BV(pin);

    if('A' == port){
        set &= PORTA;
        if(set != 0 && state == 0){ //Ändrar från 1 till 0
            PORTA ^= set;
        } else if(set == 0 && state == 1){ //Ändrar från 0 till 1
            PORTA ^= _BV(pin);
        }
    }
    else if('B' == port){
        set &= PORTB;
        if(set != 0 && state == 0){ //Ändrar från 1 till 0
            PORTB ^= set;
        } else if(set == 0 && state == 1){ //Ändrar från 0 till 1
            PORTB ^= _BV(pin);
        }
    }
    else if('C' == port){
        set &= PORTC;
        if(set != 0 && state == 0){ //Ändrar från 1 till 0
            PORTC ^= set;
        }
    }
}
```

```
        } else if(set == 0 && state == 1){ //Ändrar från 0 till 1
        PORTC ^= _BV(pin);
        }
    }
    else if('D' == port){
        set &= PORTD;
        if(set != 0 && state == 0){ //Ändrar från 1 till 0
            PORTD ^= set;
        } else if(set == 0 && state == 1){ //Ändrar från 0 till 1
            PORTD ^= _BV(pin);
        }
    }
}
//-----//
//----- SKÄRMINSTÄLLNINGAR -----//
//-----//

void write_cmd(char value) //RS är en nolla för att endast behöva läsa in commands
{
    PORTB = value;
    _delay_ms(5);
    set_pin('D', PD5, 0); // RS
    set_pin('D', PD6, 0); // RW
    _delay_ms(5);
    set_pin('D', PD7, 1); // E
    _delay_ms(5);
    set_pin('D', PD7, 0); // E
    _delay_ms(5);
}

void disp_writeCh(char val) // RS är en etta för att kunna läsa in tecken
{
    PORTB = val;
    _delay_ms(1);
    set_pin('D', PD6, 0); //RW
    set_pin('D', PD5, 1); //RS
    set_pin('D',PD7, 1); //Enable
    set_pin('D',PD7, 0); //Enable
}

void disp_nbr(int number) {
    int nbr1 = number/10;
    itoa(nbr1,timeVect,10);
    disp_writeCh(timeVect[0]);
    int nbr2 = number%10;
    nbr1 = itoa(nbr2, timeVect,10);
    disp_writeCh(timeVect[0]);
}

void disp_clear() {
    write_cmd(0b00000001); //Clear display
}

void disp_initial() //Initiala displayinställningar
{
```

```
disp_clear();
_delay_ms(5);
write_cmd(0b00111000); //Function set
_delay_ms(5);
write_cmd(0b00001111); //ON/OFF set
_delay_ms(5);
write_cmd(0b00000010); //Entry mode set
}

//-----//
//----- SKRIVA PA SKÄRMEN -----//
//-----//
void disp_waitmessage(int number) {
    disp_clear();
    disp_writeCh('W');
    disp_writeCh('a');
    disp_writeCh('i');
    disp_writeCh('t');
    disp_writeCh('i');
    disp_writeCh('n');
    disp_writeCh('g');
    disp_writeCh('t');
    disp_writeCh('i');
    disp_writeCh('m');
    disp_writeCh('e');
    disp_writeCh(':');
    disp_writeCh(' ');
    disp_nbr(number); //Visa tiden beroende på knapptryck
}
void disp_welcomemessage() {
    disp_writeCh('W'); //Visa "Welcome" på skärmen
    disp_writeCh('e');
    disp_writeCh('l');
    disp_writeCh('c');
    disp_writeCh('o');
    disp_writeCh('m');
    disp_writeCh('e');
}
void disp_stepinsidemessage() {
    disp_writeCh('S'); //Visa "Step inside" på skärmen
    disp_writeCh('t');
    disp_writeCh('e');
    disp_writeCh('p');
    disp_writeCh(' ');
    disp_writeCh('i');
    disp_writeCh('n');
    disp_writeCh('s');
    disp_writeCh('i');
    disp_writeCh('d');
    disp_writeCh('e');
}
```

```
}  
void disp_oomessage() {  
    disp_writeCh('O'); //Visa "Out of office" på skärmen  
    disp_writeCh('u');  
    disp_writeCh('t');  
    disp_writeCh(' ');  
    disp_writeCh('o');  
    disp_writeCh('f');  
    disp_writeCh(' ');  
    disp_writeCh('o');  
    disp_writeCh('f');  
    disp_writeCh('f');  
    disp_writeCh('i');  
    disp_writeCh('c');  
    disp_writeCh('e');  
  
}  
//-----  
//----- LED -----  
//-----  
void redled_on()  
{  
    set_pin('A', PA0, 1);  
}  
void yellowled_on()  
{  
    set_pin('A', PA1, 1);  
}  
void greenled_on()  
{  
    set_pin('A', PA2, 1);  
}  
void redled_off()  
{  
    set_pin('A', PA0, 0);  
}  
void yellowled_off()  
{  
    set_pin('A', PA1, 0);  
}  
void greenled_off()  
{  
    set_pin('A', PA2, 0);  
}  
int check_button() {  
  
    char val = PIND;  
  
    val = val & 0b00011010;  
    if(val == 8) //Sjuksköterskan trycker på vänta-knappen  
    {  
        return 1;  
    }  
  
    if(val == 16) //Sjuksköterskan trycker på kom in-knappen
```

```
{
    return 2;
}
if (val == 2) //Någon ringer på
{
    return 3;
}
}

//Interrupts
void enableButtonInterrupt(){
    MCUCR = 0b00000011; // sida 67
    SREG = 0x82; //?
    GICR = (1 << INT0);
}

void enableTimeInterrupt(){
    //timer for count
    TCCR1A = 0b00000000;
    TCCR1B = 0b00000101;
    TCNT1 = 65145;

    //timer for buzzer
    TCCR0 = 0b00001001;
    TCNT0 = 0b11111111;
    TIMSK = 0b00000101;
}

ISR(INT0_vect){ //Globalt button-interrupt

    _delay_ms(1); //felsökning med Berra

    int a = check_button(); //KOLLA ALLA KNAPPAR

    if(a == 1){
        waitbutton = 1;
    }

    else if(a == 2){
        comeinbutton = 1;
    }

    else if (a == 3){
        bellbutton = 1;
    }

}

ISR(TIMER0_OVF_vect){ //Buzzer

    if(bellbutton == 1 && count < 20){
        set_pin('A', PA3, 0);
        _delay_ms(1);
        set_pin('A', PA3, 1);
    }
}
```

```
TCNT0 = 0b11111111;

}
ISR(TIMER1_OVF_vect){ //Tidräknare som körs varje tiondels sekund

    if(bellbutton == 1 && count < 100 && count >= 0){
        count++;
    }
    if(waitbutton == 1 && count > 0 && count < 100){
        time++;
        waitbutton = 0;
    }

    if(waitbutton == 1 && count == 0){
        waitbutton = 0;
        time = 0;
    }

    if(comeinbutton == 1 && count == 0){
        comeinbutton = 0;
        time = 0;
    }
    if(comeinbutton == 1 && count > 0 && count < 100){
        count = 0;
        time = 0;
        bellbutton = 0;
        comeinbutton = 0;
        flag = 3;    //green
    }

    if(count == 100 && time > 0){
        count = 0;
        bellbutton = 0;
        waitbutton = 0;
        flag = 1;    //yellow
    }

    if(count == 100 && time == 0){
        count = 0;
        bellbutton = 0;
        flag = 2;    //red
    }

}

TCNT1 = 65145;
}

void actionPerformed(){

    if(flag == 3){ // tända grön led
        greenled_on();
        disp_clear();
        disp_stepinsidemessage();
        _delay_ms(500);
    }
}
```

```
    disp_clear();
    greenled_off();
    disp_welcomemessage();
}

else if (flag == 1){ // tända gul leds, skriva meddelande
    yellowled_on();
    time = time*5;
    disp_waitmessage(time);
    _delay_ms(500);
    disp_clear();
    yellowled_off();
    disp_welcomemessage();
    time = 0;
}

else if(flag == 2){ //tända röd led
    redled_on();
    disp_clear();
    disp_ooomessage();
    _delay_ms(500);
    disp_clear();
    redled_off();
    disp_welcomemessage();
}

flag = 0;
}
```

```
//Huvudprogram
void main(void)
{
    setup();
    while(1)
    {
        actionPerformed();
    }
}
```