

```

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

#define bforward 0b11111111 //Knapp 2
#define bback 0b11111101 //Knapp 0
#define bleft 0b11111010 //Knapp 5
#define bright 0b11111000 //Knapp 7
#define photodiode_on 0b1111100 //Knapp 3
#define photodiode_off 0b11111011 //Knapp 4
#define bstop 0b11111001 //Knapp 6

int right_done = 0; //Om bilen
precis har kört till höger
int left_done = 0; //Om bilen
precis har kört till vänster

void initPorts()
{
    DDRA = 0b11011000; //IR MODE OUT|IR RESET OUT|IR DIOD
    IN|RÖD DIOD OUT x2
    DDRB = 0b00000000; //IN från IR RECEIVER
    DDRC = 0b00000000;
    DDRD = 0b11111111; //OUT till motorerna
    PORTA = 0b00000000;
    PORTB = 0b11110000;
    PINB = 0b00000000;
    PORTD = 0b00000000;
}

void initADC(){
    ADMUX=(1<<REFS0) |(1<<ADLAR); //Sätt referensspänning till Vcc (+5V) och
    //vänsterjustera ADC-registret
    ADCSRA |= (1<<ADEN) |(1<<ADPS0) |(1<<ADPS1); //Aktivera AD-konvertering,
    //sätt //konverteringsklockan
    //till 1/8 av //processorns
}

void initInterrupts()
{
    //Aktivera en timer
    TCNT0 = 0x00;

    TCCR0 = 0x01;

    TIMSK = 0x01;

    sei(); //Aktivera interrupts
}

uint8_t read_ADC(){
    ADMUX |= 0b0000101;
    //Läs från PA5
    ADCSRA |= (1<<ADSC);
    //Starta konvertering A->D
    while(!(ADCSRA &(1<<ADIF))); //Gör ingenting så länge
    konverteringen inte är klar
    return ADCH;
    //Returnera 8 bitar (0-255)
}

```

```

void init_warning()
{
    PORTA |= (1<<PA4);
    //Tänd lampan
    stop();
    //Stoppa motorerna
    _delay_ms(20);
    //Stå still i X sekunder
}

void abort_warning()
{
    PORTA &= ~(1 << PA4);
    //Släck lampan
}

check_light(){
    uint8_t light_level = read_ADC();           //Uppmät ljusnivå
    if(light_level > 250)                       // &&
        is_checking_for_light == 1
        //Om nivån är tillräckligt hög : Initiera varningsprotokoll
        {
            init_warning();
        }
        else                                     //Annars:
        Avbryt varningsprotokoll
        {
            abort_warning();
        }
}

ISR(TIMER0_OVF_vect)
{
    if(PORTA &= (1<<PA3)){
        check_light();                           //Kontrollera om bilen
        är vid en mina, annars...
    }
    else                                         //Annars:
    Avbryt varningsprotokoll
    {
        abort_warning();
    }

    if(PINB == bforward){                       //Kör framåt (knapp 2)
        left_done = 0;
        right_done = 0;
        forward();
    }
    else if(PINB == bleft){                     //Kör vänster (knapp 5)
        right_done = 0;
        if(left_done==0)
            left();
        else stop();
    }
    else if(PINB == bright){                    //Kör höger (knapp 7)
        left_done = 0;
        if(right_done==0)
            right();
    }
}

```

```

        else stop();
    }
    else if(PINB == bback){           //Backa (knapp 0)
        left_done = 0;
        right_done = 0;
        back();
    }
    else if(PINB == bstop){          //Stanna (knapp 6)
        left_done = 0;
        right_done = 0;
        stop();
    }
    else if(PINB == photodiod_on){   //Sätt på
diod (knapp 3)
        PORTA |= (1<<PA3);
    }
    else if(PINB == photodiod_off){ //Stäng av
diod (knapp 4)
        PORTA &= ~(1 << PA3);
    }
}
}

void forward()
{
    PORTD = 0b10100000;
}

void back()
{
    PORTD = 0b01010000;
}

void stop()
{
    PORTD = 0b00000000;
    _delay_ms(500);
}

void right()
{
    PORTD = 0b00110000;
    _delay_ms(500);
    stop();
    right_done=1;
}

void left()
{
    PORTD = 0b11000000;
    _delay_ms(500);
    stop();
    left_done=1;
}

void main(void)
{
    initPorts();
    initInterrupts();
    initADC();
    PORTD = 0;
}

```

```
while(1)
{
}
}
```