

Klasse antrieb.cpp zum Erzeugen von PWM und Ansteuern der H-Brücken :

```
#include "antrieb.h"
#include <avr/io.h>

antrieb::antrieb()
{
    DDRB = DDRB | 112;    //Pin 4,5,6 an Port B Ausgang für Motorbrücken
    DDRH = DDRH | 32;    //Pin 5 an Port H Ausgang für Motorbrücken

    PORTB = PORTB | 112; //alle Motorpins = 1 (aus)
    PORTH = PORTH | 32;

    DDRG = DDRG | 32;    // PG5 (Pin4) ist das OC0B-Bit, Ausgang B für die PWM von Counter 0
    DDRH = DDRH | 64;    // PH6 (Pin9) ist OC2B, Ausgang B von Zähler 2
    PORTG = PORTG | 32;  // PB7 mit 1 initialisieren (Motor aus)
    PORTH = PORTH | 64;

    TCCR0A = TCCR0A | 51; // set OC0B on Compare, Fast PWM
    TCCR0B = 0; // Prescaler = 0 -> pwm stop

    TCCR2A = TCCR2A | 51; // set OC0B on Compare, Fast PWM
    TCCR2B = 0; // Prescaler = 0 -> pwm stop
}

void antrieb::set_pwm()
{
    OCR0B = n_links;    //Compare Register setzen
    OCR2B = n_rechts;
    TCCR0B = 3;        //Prescaler = 64, pwm an
    TCCR2B = 4;        //Prescaler = 64, pwm an
}

void antrieb::motoren_aus()
{
    PORTB = PORTB | 112; //alle Motorpins = 1 (aus)
    PORTH = PORTH | 32;
}

void antrieb::rechts_vor()
{
    PORTH = PORTH & 223; //vorwärts : Pin 5 von Port H auf Null
    PORTB = PORTB | 16; //nicht ! rückwärts : Pin4 von port B auf eins
}

void antrieb::rechts_rueck()
```

```

{
    PORTH = PORTH | 32;    //nicht ! vorwärts : Pin 5 von Port H auf eins
    PORTB = PORTB & 239;  //rückwärts : Pin 4 Port B auf Null
}

void antrieb::links_vor()
{
    PORTB = PORTB & 191;  //vorwärts : Pin6 von Port B auf Null
    PORTB = PORTB | 32;   //nicht ! rückwärts : Pin5 von port B auf eins
}

void antrieb::links_rueck()
{
    PORTB = PORTB | 64;   //nicht ! vorwärts : Pin 5 von Port H auf eins
    PORTB = PORTB & 223;  //rückwärts : Pin 6 Port B auf Null
}

```

Definitionsdatei der Klasse antrieb.cpp :

```

class antrieb
{
public:

    unsigned char n_links, n_rechts;

    antrieb();
    void set_pwm();
    void motoren_aus();
    void links_vor();
    void rechts_vor();
    void links_rueck();
    void rechts_rueck();

};

```