```cpp
#include "constants.h"
#include <Arduino.h>

//@brief Firmware Version
const char constants::version[] = "v1.5.0";

//@brief The path of the config file
const char constants::sdcardIO::ConfigFilePath[] = "/config.txt";


//@brief The Interval of the blinking in ms
uint16_t constants::pins::led::Interval = 500;

//@brief The Interval of the blinking in ms
uint16_t constants::pins::led::BottomLEDsIntervalOn = 500;

//@brief The Interval of the blinking in ms
uint16_t constants::pins::led::BottomLEDsIntervalOff = 500;

//@brief The Interval of the blinking in ms
uint16_t constants::pins::led::BottomLEDsIntervalBetween = 750;

//@brief Pin of the big Top LEDs, has to be a PWM Pin
uint8_t constants::pins::led::Top = 13;

//@brief Pin of one half of the body LEDs, has to be a PWM Pin
uint8_t constants::pins::led::Led1 = 15;

//@brief Pin of the other half of the body LEDs, has to be a PWM Pin
uint8_t constants::pins::led::Led2 = 2;


uint8_t constants::pins::motor::FrontLeft_Dir = 4;
uint8_t constants::pins::motor::FrontLeft_Speed = 16;
uint8_t constants::pins::motor::FrontRight_Dir = 17;
uint8_t constants::pins::motor::FrontRight_Speed = 5;
uint8_t constants::pins::motor::BackLeft_Dir = 21;
uint8_t constants::pins::motor::BackLeft_Speed = 3;
uint8_t constants::pins::motor::BackRight_Dir = 33;
uint8_t constants::pins::motor::BackRight_Speed = 22;

uint8_t constants::pins::motor::ServoLowerPin = 12;
uint8_t constants::pins::motor::ServoUpperPin = 14;
uint8_t constants::pins::motor::ServoExtenderPin = 27;

uint8_t constants::pins::motor::ServoExtenderLimitSwitchLower_Pin = 39;
uint8_t constants::pins::motor::ServoExtenderLimitSwitchUpper_Pin = 36;

//@brief The Buzzer Pin
uint8_t constants::pins::buzzer::buzzer_pin = 5;
uint8_t constants::pins::buzzer::pwm_channel = 7;

uint8_t constants::pins::uss::Trigger = 25;
```

```
54  uint8_t constants::pins::uss::Front_Sens = 36;
55  uint8_t constants::pins::uss::Back_Sens = 39;
56  uint8_t constants::pins::uss::Left_Sens = 34;
57  uint8_t constants::pins::uss::Right_Sens = 35;
58
59  uint8_t constants::uss::FrontBack_Min_Distance = 15;
60  uint8_t constants::uss::Side_Min_Distance = 20;
61
62
63  std::string constants::controller_vars::BT_ADDR = "8C:7C:B5:5F:3B:64";
64
65  //@brief Set the value of a setting based on the name
66  //@param key The name of the setting
67  //@param value The value the setting should be set to
68  //@return 0 if setting was successful
69  //@return 1 if setting name was not found
70  bool constants::setValue(std::string key, String value)
71  {
72      uint8_t uint8_t_value = atoi(value.c_str());
73
74      String tmp_value = value;
75      uint8_t ip_0, ip_1, ip_2, ip_3;
76      ip_0 = atoi(tmp_value.substring(0, tmp_value.indexOf(".")).c_str
          ());
77      tmp_value = tmp_value.substring(tmp_value.indexOf(".") + 1);
78      ip_1 = atoi(tmp_value.substring(0, tmp_value.indexOf(".")).c_str
          ());
79      tmp_value = tmp_value.substring(tmp_value.indexOf(".") + 1);
80      ip_2 = atoi(tmp_value.substring(0, tmp_value.indexOf(".")).c_str
          ());
81      tmp_value = tmp_value.substring(tmp_value.indexOf(".") + 1);
82      ip_3 = atoi(tmp_value.substring(0, tmp_value.indexOf(".")).c_str
          ());
83
84      IPAddress ipaddress_value = IPAddress(ip_0, ip_1, ip_2, ip_3);
85
86      if     (key == "pins::led::Interval")
          { constants::pins::led::Interval = uint8_t_value; }
87      else if(key == "pins::led::Top")
          { constants::pins::led::Top = uint8_t_value; }
88      else if(key == "pins::led::Led1")
          { constants::pins::led::Led1 = uint8_t_value; }
89      else if(key == "pins::led::Led2")
          { constants::pins::led::Led2 = uint8_t_value; }
90      else if(key == "pins::motor::FrontLeft_Dir")
          { constants::pins::motor::FrontLeft_Dir = uint8_t_value; }
91      else if(key == "pins::motor::FrontLeft_Speed")
          { constants::pins::motor::FrontLeft_Speed = uint8_t_value; }
92      else if(key == "pins::motor::FrontRight_Dir")
          { constants::pins::motor::FrontRight_Dir = uint8_t_value; }
93      else if(key == "pins::motor::FrontRight_Speed")
          { constants::pins::motor::FrontRight_Speed = uint8_t_value; }
94      else if(key == "pins::motor::BackLeft_Dir")
```

```
          { constants::pins::motor::BackLeft_Dir = uint8_t_value; }
 95     else if(key == "pins::motor::BackLeft_Speed")
          { constants::pins::motor::BackLeft_Speed = uint8_t_value; }
 96     else if(key == "pins::motor::BackRight_Dir")
          { constants::pins::motor::BackRight_Dir = uint8_t_value; }
 97     else if(key == "pins::motor::BackRight_Speed")
          { constants::pins::motor::BackRight_Speed = uint8_t_value; }
 98     else { return false; }
 99     return true;
100 }
```