

Actuators - Actuators Value Treatment

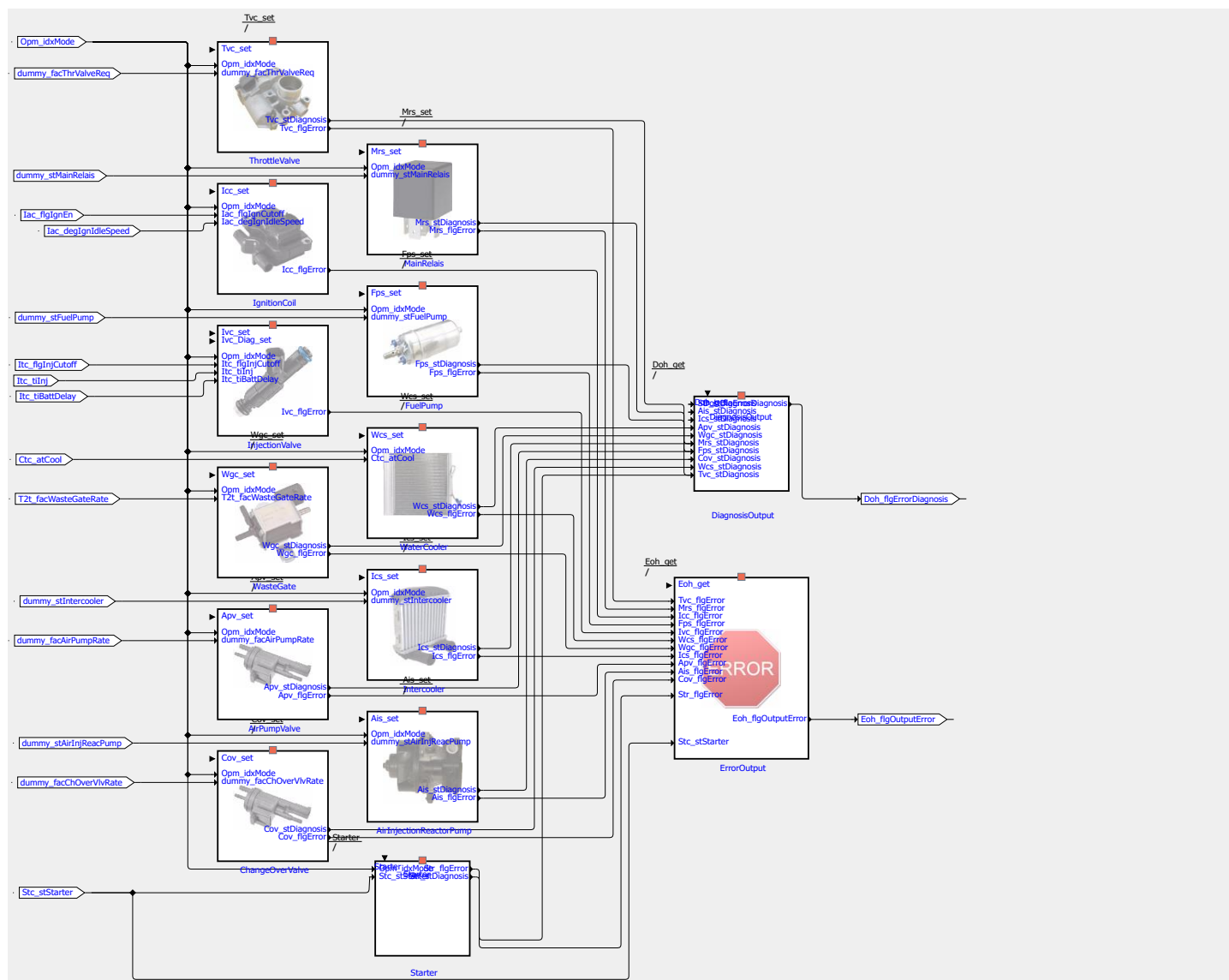
1 [Actuators Value Treatment] Actuators Value Treatment

1.1 [Overview]

Figure 1: [Actuators Function Overview]



Figure 2: Komponentenübersicht der Aktuatorik der Motorsteuerung [Actuators.Main.Actuators]



1.2 Komponentenmodule der Aktuatorik [Komponentenmodule der Aktuatorik]

In den Modulen für die Aktuatorik werden physikalische Größen in Daten gewandelt, die durch die Basissoftware-Schnittstellenklassen als elektrische Signale an die Pins der FlexECU angelegt werden.

Es wird überwacht, ob der Funktionsaufruf der Basissoftware-Schnittstellenklasse erfolgreich ist. Die Fehlerflags der einzelnen Komponentenmodule der Aktuatorik werden gesammelt und als Nachricht an das Funktionsmodul Operationsmodus weitergegeben.

Die Funktionen jedes Moduls werden gemäß der Tasklist für Bypass-Funktionen aufgerufen. Informationen zu den jeweils verwendeten Implementierungsdatentypen und Kennlinien und -feldwerten befinden sich in Anhang I.

Abbildung 5.4 zeigt die Hierarchieebene der Aktuatorik. Der Anschlussbelegung der Komponenten an die FlexECU wird in Anhang B gezeigt.

1.2.1 Drosselklappe [Drosselklappe]

Das Modul Drosselklappe (ThrottleValve) regelt die Position der Drosselklappe. Dazu wird in der Hierarchieebene ThrottleValvePosition die aktuelle Position ermittelt und anschließend, je nach Operationsmodus, die einzuregelnde Position. Dabei wird aus Sicherheitsgründen der aus den Funktionen kommende Stellwert **dummy_facThrValveReq** begrenzt, um zu verhindern, dass eine nicht erreichbare Position eingeregelt werden soll, was zu einer Beschädigung führen kann, da der Stellwert für den Motor bis zum Maximum erhöht wird.

Der untere Anschlag der Drosselklappe liegt bei rund 4° (Faktor -0,12), der obere bei weit über 90° (Faktor 1,19) und die Ruhelage oder auch Nulldrehmomentposition bei rund 14° (Faktor 0,0). Dabei ist zu beachten, dass jede Drosselklappe

fertigungsbedingt leicht unterschiedlich ist. Die obere Grenze der Stellwertlimitierung `Tvc_facThrValveLimitMax` von 1,0 entspricht einem Öffnungswinkel von 90° , also ganz geöffnet, die untere Grenze `Tvc_facThrValveLimitMin` von -0,09 einem von rund 7° .

Mit setzen des Flags `Tvc_flgThrValveReqOverride` kann die einzuregelnde Position manuell mittels der Variablen `Tvc_facThrValveReqMan` eingestellt werden.

In der Hierarchie `ThrottleValveControl` erfolgt die Positionsregelung. In die Regelung gehen Soll- und Istwert der Position ein und das benötigte Tastverhältnis zur Ansteuerung der H-Brücke (Stellwert) wird berechnet. Die Ausgabe an die Aktuatorik erfolgt anschließend in `ThrottleValveOutput`.

Zur Absicherung gegen Fehler findet in den einzelnen Teilen eine Fehler- und Plausibilitätsprüfung statt. Sobald ein Fehlverhalten auftritt, wird das Fehlerflag `Tvc_flgError` auf true gesetzt und als Nachricht weitergereicht. Die gesamte Auswertung erfolgt durch die Funktion `Tvc_set` im 10ms-Raster.

Positionserfassung der Drosselklappe

Die Positionserfassung erfolgt über zwei gegenläufige Potentiometer. Es werden beide, an den Widerständen der Potentiometer abfallenden, Spannungen (`Tvc_uSens1` und `Tvc_uSens2`) erfasst und über experimentell bestimmte Kennlinien (`Tvc_uSens2facThrValve1_CUR` und `Tvc_uSens2facThrValve2_CUR`) in Positionen (`Tvc_facThrValve1` und `Tvc_facThrValve2`) umgerechnet. Die Position `Tvc_facThrValve` ergibt sich aus dem Mittelwert der beiden Einzelpositionen.

Zur Absicherung gegen Fehler werden die Statusinformationen der Klassen zum Auslesen der analogen Spannungswerte (`Egc_stlA1` und `Egc_stlA2`) herangezogen, eine Plausibilitätsprüfung auf maximal und minimal mögliche Spannungen (`Egc_uSensMinValid` und `Egc_uSensMaxValid`) durchgeführt und auf eine Abweichung der beiden Positionswerte von kleiner 5% geprüft. Sobald ein Fehlverhalten auftritt, wird das Fehlerflag `Tvc_flgPosError` auf true gesetzt.

Positionsregelung der Drosselklappe

Die Positionsregelung erfolgt mit einem PID-Regler. Eingang ist die Regeldifferenz zwischen Soll- und Istwert, Ausgang das benötigte Tastverhältnis.

Die Auslegung des Reglers erfolgt nach den Erfahrungsregeln zur Einstellung von PIDReglern nach Chien, Hrones, Reswick für Führungsverhalten.

Gründe dafür sind:

- Unkenntnis des Blockschaltbilds
- Unkenntnis der Übertragungsfunktion
- Unkenntnis der Parameter
- Nichtlinearitäten, wie Reibung und Stellgrößenbegrenzung
- Regelstrecke mit Ausgleich und guter Dämpfung

Abbildung 5.5 zeigt den Sprung des Tastverhältnisses (rot) und die Sprungantwort der Regelstrecke (grün) und die daraus abgeleitete Berechnung der Reglerparameter. Abbildung 5.6 zeigt den Sprung der Soll-Position (rot) von Null auf Eins und die Sprungantwort der Positionsregelung (rot). Damit wird die, aus Erfahrungswerten bestimmte, Einregelzeit von rund 100ms erreicht. Die Abtastzeit von 10ms ist ebenfalls auf Erfahrungswerte zurückzuführen.

Problematisch ist die Regelung um die Ruhelage, da ein relativ hohes Tastverhältnis benötigt wird um die Klappe zu bewegen, was durch das Moment der verbauten Torsionsfeder zustande kommt, der integrierende Anteil des PID-Reglers jedoch relativ viel Zeit benötigt um sich von positiven auf negative Werte abzubauen und umgekehrt. Zur Lösung dieser Problematik wird zusätzlich eine Vorsteuerung verwendet, die additiv wirkt, d. h. bei negativen Sollwerten wird von vornherein bereits ein negatives Tastverhältnis `Tvc_facDutycyclePilotControlNeg` und bei positiven Stellwerten ein positives Tastverhältnis `Tvc_facDutycyclePilotControlPos` vorgegeben. Dadurch wird auch bei kleineren Verstärkungsfaktoren eine zügiges einregeln erreicht. Die Werte von 2000 und -2000 werden aus Erfahrungen heraus festgelegt. Die Vorsteuerung, die das System noch schneller macht, wird bei der Reglerauslegung nicht berücksichtigt und zu Null gesetzt.

Durch das Flag `Tvc_flgThrValveDutycycleOverride` kann auf eine manuelle Vorgabe des Tastverhältnisse `Tvc_facThrValveDutycycleMan` umgeschaltet werden. Falls in den vorangegangenen Berechnungen ein Fehler aufgetreten ist, wird der Regler zurückgesetzt.

Signalausgabe der Drosselklappe

Bei der Ausgabe muss beachtet werden, dass bei Tastverhältnis von Null die H-Brücke ausgeschaltet wird. Daher findet eine Prüfung statt, ob es zwischen -1 und 1 liegt und gegebenenfalls reagiert. Ebenfalls ausgeschaltet wird die H-Brücke im Fehlerfall oder wenn der Operationsmodus nicht Normal ist. Im ausgeschalteten Zustand stellt sich die Ruhe- bzw. Notlaufelage ein.

Figure 3: Reglerauslegung für die Drosselklappe nach Chien, Hrones, Reswick [actuators_png_1]

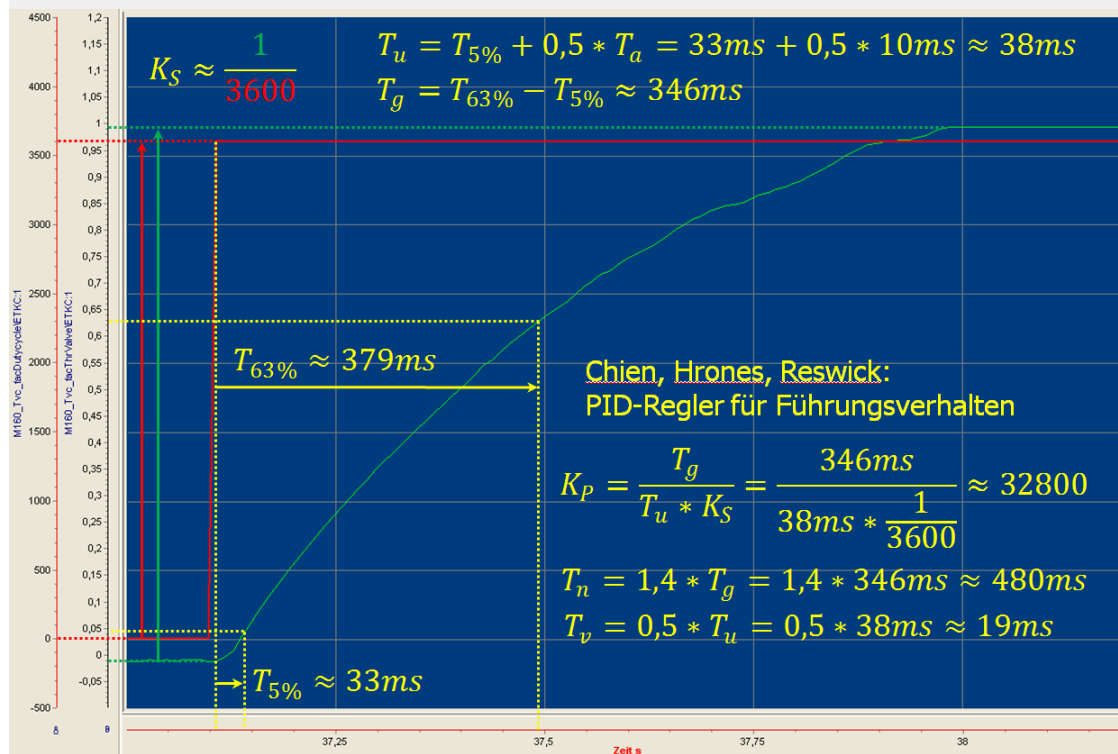
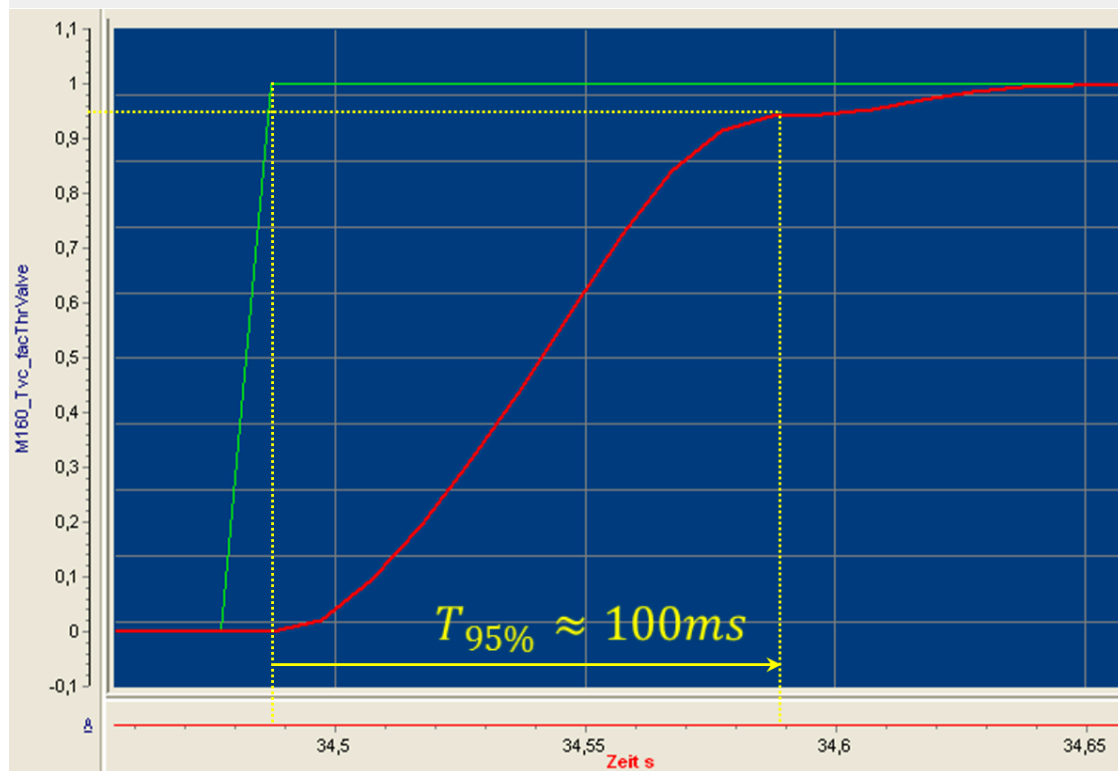


Figure 4: Sprungantwort der Drosselklappe mit Regler [actuators_png_2]



1.2.2 Wastegate [Wastegate]

Das Modul Wastegate (WasteGate) steuert das Wastegate über ein PWM-Signal an. Dabei ist das Wastegate bei einem Tastverhältnis von Null voll geöffnet und bei einem Tastverhältnis von Eins komplett geschlossen. Im normalen Operationsmodus kann durch Setzen des Flags [Wgc_flgWastegateRateOverride](#) das Tastverhältnis manuell mittels der Variablen [Wgc_facWasteGateRateMan](#) eingestellt werden.

Zur Absicherung gegen Fehler werden die Statusinformationen der Klasse zum Setzen des PWM-Signals herangezogen. Sobald ein Fehlverhalten auftritt, wird das Fehlerflag [Wgc_flgError](#) auf true gesetzt und als Nachricht weitergereicht.

Die gesamte Auswertung erfolgt durch die Funktion [Wgc_set](#) im Synchron-Raster.

1.2.3 Einspritzventile [Einspritzventile]

Mit dem Modul Einspritzventile (InjectionValve) werden die Einspritzventile über die Basissoftware angesteuert. Zur Vereinfachung der Motorsteuerung wird auf eine zylinderindividuelle Einspritzzeit verzichtet und bei allen Zylindern das Ventil gleich lange offengelassen, d. h., jeder Wert im Feld [lvc_tilnjEffCylInd_Arr](#) wird mit [lvc_tilnj](#) beschrieben. Durch Setzen des Flags [lvc_flgInjTimeOverride](#) kann die Einspritzzeit manuell mittels der Variablen [lvc_tilnjMan](#) eingestellt werden.

Weiterhin wird die batteriespannungsabhängige Verzugszeit übergeben, die durch Setzen des Flags [lvc_flgBattDelayOverride](#) manuell mittels der Variablen [lvc_tiBattDelayMan](#) überschrieben werden kann.

Aus Sicherheitsgründen wird geprüft, ob der normale Operationsmodus ausgeführt wird und gegebenenfalls das Muster 11111112 zur Einspritzabschaltung aller Zylinder gesetzt. Sonst wird je nach Nachricht [lvc_flgInjCutoff](#) eine Abschaltung durchgeführt. Durch Setzen des Flags [lvc_flgInjCutoffOverride](#) kann die Einspritzung manuell durch [lvc_flgInjCutoffMan](#) an- und abgeschaltet werden.

Zur Absicherung gegen Fehler werden die Statusinformationen der Klasse zum Setzen der Werte für die Saugrohreinjection herangezogen. Sobald ein Fehlverhalten auftritt, wird das Fehlerflag [lvc_flgError](#) auf true gesetzt und als Nachricht weitergereicht.

Die Funktion [lvc_set](#) wird im Synchron-Raster ausgeführt. Der Vollständigkeit halber wird die Funktion [lvc_Diag_set](#) im 100ms-Raster ausgeführt, wobei die Diagnoseinformationen im Folgenden nicht weiter verwendet werden.

Hint:

Die aktuellste Version von EHOOKS befindet sich noch im Entwicklungsstadium und es ist aufgrund eines Initialisierungsfehlers bei der Kompilierung nicht möglich, Parameterfelder zu verwenden. Daher wird bei der Implementierung des Feldes mit den Einspritzendwinkeln [weea](#) der etwas umständlichere Weg gewählt, das Variablenfeld [lvc_deglInjEnd](#) mit Parametern zu beschreiben und dann zu übergeben.

Während der Arbeit wurden diverse Fehler im Saugrohreinjection-Modul der Basissoftware festgestellt. Beispielsweise ist unklar wann und wie der Modus von Simultanstart auf normale Operation umgeschaltet wird und welche Auswirkungen einige Parameter auf das Verhalten haben.

1.2.4 Zündspulen [Zündspulen]

Mit dem Modul Zündspulen (IgnitionCoil) werden die Zündspulen über die Basissoftware angesteuert. Zur Vereinfachung der Motorsteuerung wird auf einen zylinderindividuellen Zündwinkel verzichtet und auf alle Zylindern zugleich gezündet, d. h., jeder Wert im Feld [lcc_deglgn_Arr](#) wird mit [lcc_deglgn](#) beschrieben. Durch setzen des Flags [lcc_flgIgnAngleOverride](#) kann der Zündwinkel manuell mittels der Variablen [lcc_deglgnMan](#) eingestellt werden.

Aus Sicherheitsgründen wird geprüft, ob der normale Operationsmodus ausgeführt wird und gegebenenfalls eine Aussetzung der Zündung veranlasst. Ansonsten wird je nach Nachricht [lcc_flgIgnCutoff](#) eine Abschaltung durchgeführt. Außerdem kann durch setzen des Flags [lcc_flgIgnCutoffOverride](#) die Zündung manuell durch [lcc_flgIgnCutoffMan](#) an und abgeschaltet werden. Dies geschieht jeweils bei einer fallenden Signalfanke durch die Funktion [fadeout](#). Sobald kein Abschaltewunsch mehr vorliegt, findet eine Wiedereinsetzung mit einer steigenden Signalfanke durch die Funktion [fadeln](#) statt. In der Variablen [lcc_stlgnFadeOutReq](#) kann der aktuelle Zustand der Abschaltung verfolgt werden.

Zur Absicherung gegen Fehler werden die Statusinformationen der Klasse zum Setzen der Werte für die Zündung herangezogen. Sobald ein Fehlverhalten auftritt, wird das Fehlerflag [lcc_flgError](#) auf true gesetzt und als Nachricht weitergereicht.

Die Funktion `Icc_set` wird im Synchron-Raster ausgeführt.

1.2.5 Wasserkühler [Wasserkühler]

Mit dem Modul Wasserkühler (WaterCooler) wird der Lüfter des Wasserkühlers, je nach Kühlmitteltemperatur `Ctc_atCool`, über einen Hystereseschalter, an- und ausgeschaltet. Durch setzen des Flags `Wcs_flgWaterCoolerOverride` kann der Lüfter manuell durch `Wcs_flgWaterCoolerMan` geschaltet werden.

Da es sich um einen „Low-Side-Switch“ handelt, wird das Signal `Wcs_stWaterCooler` invertiert, bevor es ausgegeben wird. Zur Absicherung gegen Fehler werden die Statusinformationen der Klasse zum Setzen der Digitalausgänge herangezogen. Sobald ein Fehlverhalten auftritt, wird das Fehlerflag `Wcs_flgError` auf true gesetzt und als Nachricht weitergereicht.

Die Funktion `Wcs_set` wird im 100ms-Raster ausgeführt.

1.2.6 Hauptrelais [Hauptrelais]

Mit dem Modul Hauptrelais (MainRelais) wird das Hauptrelais des Steuergeräts geschaltet. In der Regel ist das Hauptrelais während des Betriebs des Steuergeräts immer eingeschaltet und wird nur zum Abschalten verwendet. Das Steuergerät kann sich damit selbst Ausschalten, nachdem beispielsweise ein Nachlauf zum Speichern von Werten im EEPROM1 erfolgt ist. Durch setzen des Flags `Mrs_flgMainRelaisOverride` kann das Steuergerät manuell durch `Ivc_flgMainRelaisMan` (an- und) abgeschaltet werden.

Da es sich um einen „Low-Side-Switch“ handelt, wird das Signal `Mrs_stMainRelais` invertiert, bevor es ausgegeben wird. Zur Absicherung gegen Fehler werden die Statusinformationen der Klasse zum Setzen der Digitalausgänge herangezogen. Sobald ein Fehlverhalten auftritt, wird das Fehlerflag `Mrs_flgError` auf true gesetzt und als Nachricht weitergereicht.

Die Funktion `Mrs_set` wird im 100ms-Raster ausgeführt.

Hint:

Zum Zeitpunkt der Arbeit ist das Modul zum Nachlauf (Afterrun & Shutdown) in der Basissoftware noch nicht vollständig implementiert. Das Steuergerät schaltet sich immer ca. 5sek nach Abschalten von Klemme 15 aus und ein Schalten des Hauptrelais mit diesem Modul bewirkt nichts. Es wird für spätere Zwecke implementiert.

1.2.7 Kraftstoffpumpe [Kraftstoffpumpe]

Das Modul Kraftstoffpumpe (FuelPump) ist, mit angepasster Benennung, analog zum Modul Hauptrelais aufgebaut und wird daher nicht extra vertieft.

1.2.8 Ladeluftkühler [Ladeluftkühler]

Das Modul Ladeluftkühler (Intercooler) ist, mit angepasster Benennung, analog zum Modul Hauptrelais aufgebaut und wird daher nicht extra vertieft.

1.2.9 Sekundärluftpumpe [Sekundärluftpumpe]

Das Modul Sekundärluftpumpe (AirInjectionReactorPump) ist, mit angepasster Benennung, analog zum Modul Hauptrelais aufgebaut und wird daher nicht extra vertieft.

1.2.10 Umschaltventil Sekundärluft [Umschaltventil Sekundärluft]

Das Modul Umschaltventil Sekundärluft (AirPumpValve) steuert ein Ventil über ein PWM-Signal an. Dabei ist das Ventil bei einem Tastverhältnis von Null voll geöffnet und bei einem Tastverhältnis von Eins komplett geschlossen. Im normalen Operationsmodus kann durch setzen des Flags `Apv_flgAirPumpRateOverride` das Tastverhältnis manuell mittels der Variablen `Apv_facAirPumpRateMan` eingestellt werden.

Zur Absicherung gegen Fehler werden die Statusinformationen der Klasse zum Setzen des PWM-Signals herangezogen. Sobald ein Fehlverhalten auftritt, wird das Fehlerflag `Apv_flgError` auf true gesetzt und als Nachricht weitergereicht.

Die gesamte Auswertung erfolgt durch die Funktion `Apv_set` im 100ms-Raster.

Hint:

Im Verlauf der Arbeit konnte die genaue Aufgabe des Ventils nicht vollständig geklärt werden. Eine Vermutung ist, dass es sich um ein Unterdruckventil handelt, d. h., mit dem PWM-Signal wird festgelegt wie viel Druck auf das eigentliche Ventil gegeben wird, um damit die Öffnung festzulegen.

1.2.11 Umschaltventil Druckgeber [Umschaltventil Druckgeber]

Das Modul Umschaltventil Druckgeber (ChangeOverValve) ist, mit angepasster Benennung, analog zum Modul Umschaltventil Sekundärluft aufgebaut und wird daher nicht extra vertieft. Es handelt sich dabei um dasselbe Ventil.

Hint:

Im Verlauf der Arbeit konnte die genaue Aufgabe des Ventils nicht vollständig geklärt werden. Eine Vermutung ist, dass damit gesteuert wird, woher der Druck für das Umschaltventil Sekundärluft kommt. Im Falle eines Unterdrucks kann der Druck dem Saugrohr entnommen werden, bei vorherrschendem Ladedruck wird dann mit diesem auf eine andere Quelle vor dem Saugrohr umgeschaltet.