

# Sensors - Sensor Value Treatment

## 1 [Sensor Value Treatment]

### 1.1 [Overview]

Figure 1: [Sensors Function Overview]

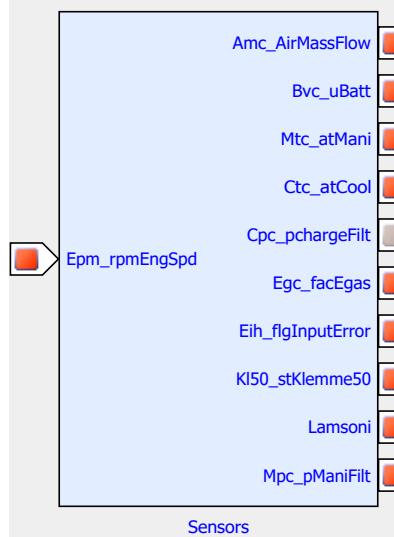
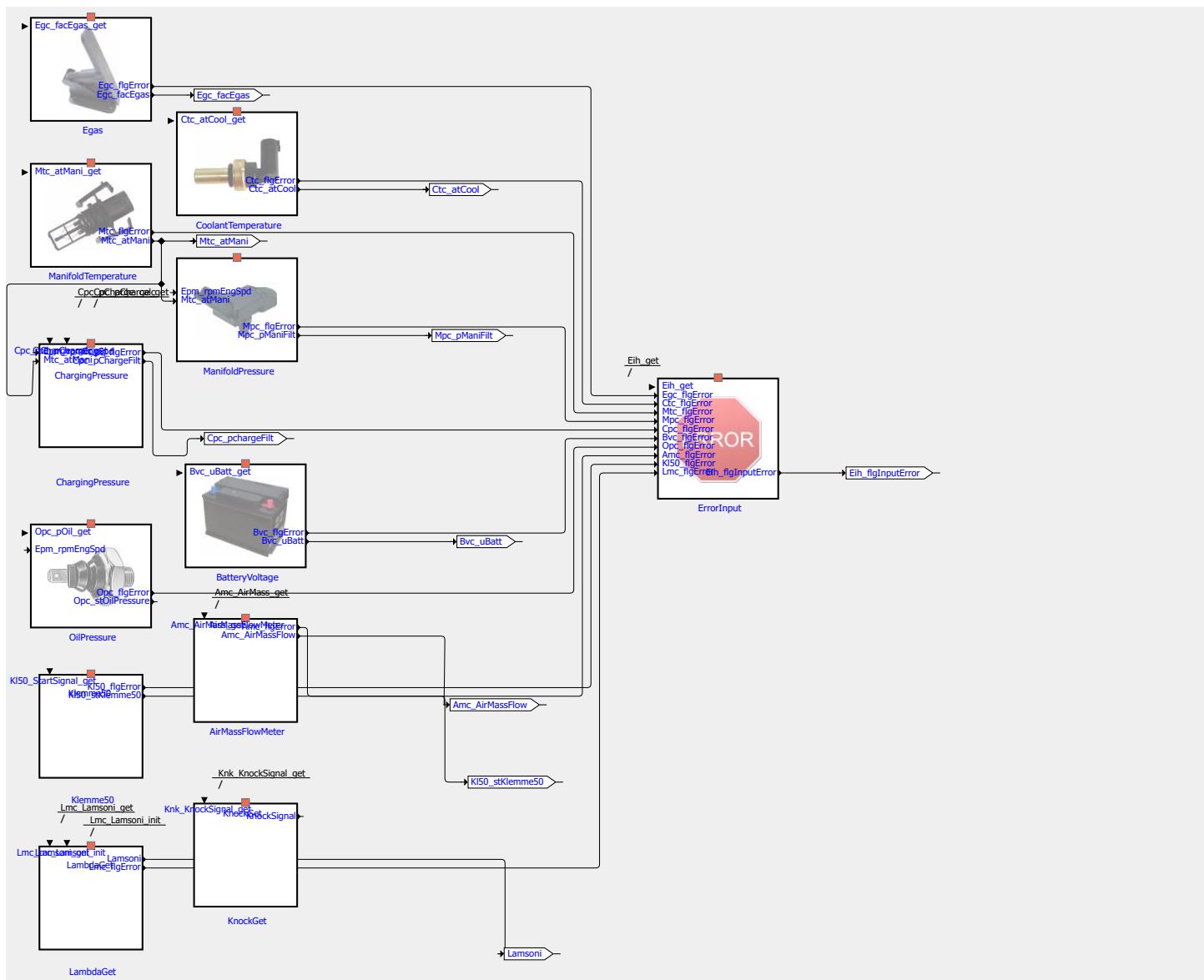


Figure 2: Komponentenübersicht der Sensorik der Motorsteuerung [Sensors.Main.Sensors]



## 1.2 Komponentenmodule der Sensorik [Komponentenmodule der Sensorik]

In den Modulen für die Sensorik werden Rohdaten über die Basissoftware-Schnittstellenklassen aus Sensoren ausgelesen und durch Rechnung oder über Kennlinien in physikalische Größen gewandelt und als Nachrichten für alle anderen Module zur Verfügung gestellt.

Es werden alle Signale auf Plausibilität geprüft und im Fehlerfall ein Fehlerflag gesetzt. Außerdem wird überwacht, ob der Funktionsaufruf der Basissoftware-Schnittstellenklasse erfolgreich ist. Die Fehlerflags der einzelnen Komponentenmodule der Sensorik werden gesammelt und als Nachricht an das Funktionsmodul Operationsmodus weitergegeben.

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

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

### 1.2.1 Egas-Steller [Fahrpedal]

Das Modul Egas-Steller (Egas) erfasst die Position des Egas-Stellers. Es werden beide, an den Widerständen der Potentiometer abfallenden, Spannungen erfasst und über experimentell bestimmte Kennlinien in Positionen umgerechnet. Die Position ergibt sich aus dem Mittelwert der beiden Einzelpositionen. Es ist möglich, über Egc\_flgEgasOverride auf eine manuelle Positioneingabe durch Egc\_facEgasMan umzuschalten. Mit dem Flag Egc\_facEg

`asFilt` kann das Signal zusätzlich durch einen Tiefpass gefiltert werden, um Sprünge und große Änderungsraten zu vermeiden, bevor es als Nachricht weitergeben wird.

Zur Absicherung gegen Fehler werden die Statusinformationen der Klassen zum Auslesen der analogen Spannungswerte (`Egc_stIA1` und `Egc_stIA2`) 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, und die Position nicht durch eine manuelle Eingabe gesetzt ist, wird das Fehlerflag `Egc_flgError` auf true gesetzt und als Nachricht weitergereicht.

Die gesamte Auswertung erfolgt durch die Funktion `Egc_facEgas_get` im 10ms-Raster.

### 1.2.2 Kühlmitteltemperatur [Kühlmitteltemperatur]

Das Modul Kühlmitteltemperatur (CoolantTemperature) erfasst die Kühlmitteltemperatur am Auslass des Motorblocks. Bei dem Sensor handelt es sich um ein NTC-Element, dass zusammen mit dem internen Pull-up-Widerstand des Steuergerätepins einen Spannungsteiler mit 5V -Versorgung bildet. Die Auswertung berechnet aus der am Sensor abfallenden Spannung den Widerstand `Ctc_rSens` nach Gleichung 5.1 und daraus über eine hinterlegte Kennlinie die Temperatur `Ctc_atCool`.

Figure 3: [sensors.png\_1]

$$R_{Sensor} = \frac{U_{Sensor} \times R_{Pull-Up}}{U_{Supply} - U_{Sensor}}$$

Um eine gute Auflösung im hohen Temperaturbereich zu erhalten, wird ein Pin mit einem Pull-up-Widerstand von 1000 gewählt.

Zur Absicherung gegen Fehler werden die Statusinformation der Klasse zum Auslesen des analogen Spannungswerts `Ctc_stIA` herangezogen, eine Plausibilitätsprüfung auf minimal und maximal mögliche Spannungen (`Ctc_uSensMinValid` und `Ctc_uSensMaxValid`) durchgeführt und geprüft, ob die ausgelesene Spannung `Ctc_uSens` kleiner ist als die angegebene Referenzspannung der Versorgung `Ctc_uSupply`. Mit der letzten Prüfung wird außerdem eine Division durch 0 verhindert und gegebenenfalls die weitere Auswertung nicht durchgeführt. Sobald ein Fehlverhalten auftritt, wird das Fehlerflag `Ctc_flgError` auf true gesetzt und als Nachricht weitergereicht.

Die gesamte Auswertung erfolgt durch die Funktion `Ctc_atCool_get` im 1000ms-Raster.

### 1.2.3 Ansauglufttemperatur [Ansauglufttemperatur]

Das Modul Ansauglufttemperatur (ManifoldTemperature) erfasst die Temperatur der angesaugten Luft im Saugrohr. Auch bei diesem Sensor handelt es sich um ein NTCElement, das analog zum Kühlmitteltemperatursensor funktioniert.

Um eine gute Auflösung im niedrigen Temperaturbereich zu erhalten, wird ein Pin mit einem Pull-up-Widerstand von 2150 gewählt. Die gesamte Auswertung erfolgt durch die Funktion `Ctc_atCool_get` im 20ms-Raster.

### 1.2.4 Saugrohrdruck [Saugrohrdruck]

Das Modul Saugrohrdruck (ManifoldPressure) erfasst den Druck im Saugrohr. Der Sensor wird mit einer 5V -Spannung versorgt und liefert einen analogen Spannungswert `Mpc_uSens`, der über eine lineare Kennlinie `Mpc_uSens2pManiBase_CUR` in den Druck umgerechnet wird. Zusätzlich wird über die Kennlinie `Mpc_atManiTmp2facPresCorr_CUR` ein Korrekturfaktor, je nach Ansauglufttemperatur, multiplikativ in die Berechnung miteinbezogen.

Zur Absicherung gegen Fehler werden die Statusinformation der Klasse zum Auslesen des analogen Spannungswerts `Mpc_stIA` herangezogen und eine Plausibilitätsprüfung auf minimal und maximal mögliche Spannungen (`Mpc_uSensMinValid` und `Mpc_uSensMaxValid`) durchgeführt. Sobald ein Fehlverhalten auftritt, wird das Fehlerflag `Mpc_flgError` auf true gesetzt und als Nachricht weitergereicht.

Aufgrund von Pulsationseffekten im Saugrohr durch die Absaugung in die einzelnen Zylinder wird ein gemittelter Saugrohrdruck zur weiteren Verarbeitung verwendet. Dazu wird, solange kein Fehler vorliegt und der Motor sich dreht (Drehzahl größer 30 1 min), im 1ms-Raster in der Funktion `Mpc_pMani_get` der Saugrohrdruck erfasst und in einem Akkumulator aufsummiert, während gleichzeitig die Anzahl erfasster Werte in einem Zähler mitgezählt wird. Zur Darstellung der Pulsationseffekte wird der ausgelesene Druck ohne Mittelwertsbildung als Nachricht `Mpc_pMani` zur Verfügung gestellt.

Im Synchron-Raster werden dann in der Funktion `Mpc_pMani_calc` der Mittelwert durch Division der aufsummierten Druckwerte durch die Anzahl gebildet, in `Mpc_pManiFilt` zur Verfügung gestellt und sowohl Akkumulator als auch Zähler zurückgesetzt. Der Synchron- Interrupt wird für jeden Zylinder alle zwei Umdrehungen, also einmal pro Arbeitsspiel ausgelöst, d.h., bei jeder Mittelwertsberechnung des Saugrohrdrucks wird sichergestellt, dass sich die Kurbelwelle um genau den gleichen Winkel gedreht hat seit der vorangegangenen Berechnung, nämlich 240°KW. Dadurch ist sichergestellt, dass bei jeder Auswertung die gleichen Konditionen vorliegen. Zusätzlich wird durch Prüfung des Zählers auf Null eine Division durch Null verhindert.

Das Aufsummieren des Akkumulators und Inkrementieren des Zählers wird als atomarer Rechenschritt ausgeführt (Kennzeichnung Beginn durch ^, Ende durch v), um zu verhindern, dass der Synchron-Interrupt dazwischen aufgerufen wird. Gleiches gilt für das Rücksetzen der Zähler, auch wenn hier die Notwendigkeit nicht gegeben ist, da der Synchron-Interrupt präemptiv ausgeführt wird (s. auch Abschnitt 2.4).

### 1.2.5 Batteriespannung [Batteriespannung]

Das Modul Batteriespannung (BatteryVoltage) erfasst die Batteriespannung über einen Spannungsteiler. Dabei wird der Bereich der Batteriespannung, durch entsprechende Wahl der Widerstände des Spannungsteilers, von rund 0 - 21V auf den Messbereich eines Analogeingangs des Steuergeräts von 0 - 5V abgebildet.

Der ausgelesene Wert `Bvc_uSens` wird nach Gleichung 5.2 in die Batteriespannung zurückgerechnet und in `Bvc_uBatt` als Nachricht zur Verfügung gestellt.

Figure 4: [sensors.png\_2]

$$U_{Batt} = \frac{R_{High} + R_{Low}}{R_{Low} \times U_{Sensor}}$$

Zur Absicherung gegen Fehler werden die Statusinformation der Klasse zum Auslesen des analogen Spannungswerts (`Bvc_stIA` herangezogen und eine Plausibilitätsprüfung auf maximal und minimal sinnvolle Spannungen (`Bvc_uBattMin` und `Bvc_uBattMax`) durchgeführt. Sobald ein Fehlverhalten auftritt, wird das Fehlerflag `Bvc_flgError` auf true gesetzt und als Nachricht weitergereicht.

Die gesamte Auswertung erfolgt durch die Funktion `Bvc_uBatt_get` im 10ms-Raster.

**Hint:**

Das Komponentenmodul Batteriespannung ist notwendig, da in der aktuellen Version der Basissoftware die Batteriespannung noch nicht richtig ausgelesen und in der Variablen `BattU_u` zur Verfügung gestellt wird. Sie beinhaltet immer 0V. Sobald das behoben ist, wird das Modul nicht weiter benötigt.

### 1.2.6 Öldruck [Öldruck]

Das Modul Öldruck (OilPressure) stellt fest, ob bei drehendem Motor der Öldruck aufgebaut wird. Der Sensor ist ein Schalter, der bei fehlendem Öldruck den Anschluss auf Masse zieht. Die Funktion `Opc_pOil_get` wird im 10ms-Raster ausgeführt und gibt den Zustand `Opc_stSens` des Sensors zurück. Sobald die Statusinformation der Klasse zum Auslesen des Digitalwerts negativ ist oder bei einer Drehzahl größer `Opc_rpmEngMinOilPressure` kein Öldruck vorhanden ist, wird das Fehlerflag `Opc_flgError` auf true gesetzt und als Nachricht weitergereicht.

Zusätzlich steht der aktuelle Zustand in der Nachricht `Opc_stOilPressure` zur Verfügung.